Privacy-compliant software reuse: A framework for considering privacy compliance in software reuse scenarios

Jenny Guber [1]

[1] *Department of Information Systems, University of Haifa, Haifa, Israel*

Abstract

In recent years, privacy-compliant software development has become an important topic, especially with the emergence of the EU General Data Protection Regulation (GDPR). Existing practices of software development challenge privacy compliance by increasingly promoting reuse, adaptation and integration of existing software artifacts from organizational or open-source repositories. Methods and approaches have been introduced to accelerate and improve development through reuse on the one hand and to mitigate threats related to data privacy on the other hand. However, the operationalization of this body of knowledge for developing systems that intensively reuse software artifacts is understudied.

Moreover, ontologies, taxonomies and frameworks developed to conceptualize, organize and model privacy requirements focus on forward engineering activities (software design and development), and are less oriented for application in existing software projects and artifacts that are considered for reuse and integration.

The aim of this research is to create a framework aimed to investigate, explore and guide privacy-compliant software reuse, especially in open-source environments. To this end, we will follow a design science approach whose main artifact will be a privacy compliance assessment method. The method will be developed in three steps: (1) systematically reviewing and analyzing the state-of-the-art in privacy-compliant software reuse; (2) empirically studying open-source repositories (in particular, GitHub) for privacy discussions, including ontology-based machine learning method for privacy discussions identification; and (3) developing and evaluating a privacy assessment method, for supporting reuse decisions, utilizing the existing models and frameworks.

Keywords

Privacy; Software Reuse; Compliance; Software Development; Open-Source; GDPR

1. Introduction and motivation

1.1. Privacy regulations, strategies and technologies

Privacy is a fundamental human right. In the digital world, we are dependent on trustworthy functioning of information and communication technologies on one hand and experience a growing power of imbalance between data processing entities and the individuals whose data is at stake on the other hand [1]. To protect the individuals' right for privacy, several regulations and standards have been established, with the General Data Protection Regulation (GDPR) [2] being the most studied one. Published in 2016 and enforced in Europe since May 2018, the GDPR has introduced several meaningful changes [3], expanding the scope of data protection and the definition of personal data.

The evolution of information technologies and the growing concern for privacy and data protection yielded the development of Privacy Enhancing Technologies (PETs), for assessing and mitigating

ER2023: Companion Proceedings of the 42nd International Conference on Conceptual Modeling: ER Forum, 7th SCME, Project Exhibitions, Posters and Demos, and Doctoral Consortium, November 06-09, 2023, Lisbon, Portugal

jguber@campus.haifa.ac.il (J. Guber)

0000-0002-2585-6601 (J. Guber)

privacy risks. PETs protect the individual's privacy using technical means, such as encryption, anonymous communication protocols, attribute-based access and private database querying [4].

To utilize the full benefit for privacy and data protection, PETs need to be introduced into the initial stages of system development, i.e., design, rather than added in late development stages [1], [5]. The Privacy-by-Design (PbD)[2] approach has been developed [6] with the aim to ensure data protection of individuals by integrating privacy considerations from the outset of the development of products, services, business practices, and infrastructures. PbD can be supported through eight privacy design strategies [7]: minimize, hide, separate, aggregate, inform, control, enforce and demonstrate.

While the PbD approach is already incorporated into the industry standards and practices and the importance of PETs is recognized, those techniques and approaches focus mainly on forward engineering activities, and privacy compliance of existing artifacts is less studied.

1.2. Privacy compliance in software engineering

Even before the GDPR came into force, it has been imperative for organizations to consider privacy compliance at the initial stages of the software development [8]. However, software reuse raises additional challenges to privacy compliance. While the existing privacy compliance approaches mainly reside in the forward engineering activities, i.e., designing and building privacy compliant software components [9], assessing compliance of existing artifacts and integrating them in an existing system in a privacy compliant manner are still challenging. These require reverse engineering of privacy requirements and analysis of applied privacy strategies and PETs. The effort required to adapt privacy-compliant software reuse methods to different (and evolving) privacy regulations should be further investigated. In addition, objectively measuring the level of privacy compliance may assist in improving reuse decisions in general and selecting the most appropriate artifacts for reuse in particular.

1.3. Privacy models, ontologies and taxonomies

Over the years, several ontologies and taxonomies have been proposed to conceptualize, organize and model privacy requirements. Two of the recent examples are the ontology by Gharib et al. [10] and the taxonomy by Sangaroonsilp et al. [11]. Conducting a systematic literature review, Gharib et al. identified 55 concepts and relations, grouped into four main categories: organizational concepts (including agentive, intentional and informational entities, as well as entities' interactions), risk, treatment, and privacy [requirement] concepts. Sangaroonsilp et al. developed a taxonomy that provides a comprehensive set of privacy requirements based on four well-established personal data protection regulations and privacy frameworks, including GDPR, ISO/IEC 29100, Thailand Personal Data Protection Act (PDPA) and Asia-Pacific Economic Cooperation (APEC). The taxonomy includes seven categories (User Participation, Notice, User Desirability, Data Processing, Breach, Complaint/Request and Security), their sub-categories and the specific privacy requirements that may belong to multiple categories and sub-categories.

In addition, to facilitate and realize the Data Protection by Design approach, a few projects have been initiated by individuals and by the EU to create frameworks for privacy compliance. One example is the privacy threat analysis framework LINDDUN [12], which stands for Linkability, Identifiability, Non-repudiation, Deniability, Disclosure of information, Unawareness and Non-compliance – privacy threat types that negate common widely accepted privacy properties. Additionally, PDP4E project suggested methods and tools for GDPR compliance through privacy and data protection engineering by implementing those methods into ongoing systematic engineering practices, and DEFeND project (Data Governance for Supporting GDPR) delivered a platform for assisting organizations achieve compliance with legal and privacy requirements, focusing on the GDPR implementation [13]. However, those frameworks concentrate on incorporating compliance from the beginning of the software development lifecycle. Facilitating privacy compliance while performing software reuse remains insufficient.

[2] Sometimes referred to as "Data Protection by Design".

1.4. Paper Structure

Privacy has gained an increasing interest in the last two decades and a variety of regulations, strategies and technologies have been proposed to address its different aspects. While performing forward engineering activities in software development in a manner compliant to privacy regulations has already been explored by different studies but achieving privacy compliant software reuse is still understudied. This calls for developing a method that aims to assess and evaluate privacy level of the components and to integrate them into the system in a compliant way.

In addition, while modeling by taxonomies, ontologies and frameworks has been underway, those mostly target privacy requirements and concentrate on forward engineering activities, and do not handle assessing privacy levels of existing software artifacts to support reuse.

The rest of this paper is organized as follows. Section 2 presents the research objectives and the research questions. Section 3 presents the related work. Section 4 describes the research methodology and the expected contributions, while Section 5 details the progress achieved so far.

2. Research objectives and questions

2.1. Research objectives

Our main working hypothesis is that software reuse challenges privacy compliance in general and PbD in particular. Two major gaps identified in literature serve as the motivation for our research.

The first gap regards analyzing, mining, monitoring and tracing privacy requirements. Despite the large corpus of privacy regulations and compliance methods, developers face challenges to operationalize them [14]. This is significantly noticeable when the development includes reuse of open-source artifacts. Our first objective is to analyze and mine existing artifacts for privacy characteristics through analyzing their meta-data and discussions.

The second gap regards adaptation and integration of artifacts in complex projects that comprise of dependent software artifacts [15]. Complex projects may comprise of artifacts originating from different sources (i.e., proprietary software, third-party software, open-source repositories), and having different levels of privacy. Our second objective is to analyze, simulate and devise the "aggregate" level of privacy of the entire system and check whether it satisfies the overall privacy requirements, thus performing privacy compliance assessment of the artifacts and the integrated software.

2.2. Research Questions

To fulfil the above objectives, we consider the following research questions (RQ):

1. **RQ1**. What is the current state-of-the-art in privacy-compliant software reuse?
- RQ1.1 What are the regulations considered in privacy-compliant software reuse?
- RQ1.2 What are the business and technological domains in which privacy-compliant software reuse is researched?
- RQ1.3 What are the utilized reuse approaches and how do they relate to the reuse landscape?
- RQ1.4 What are the privacy strategies implemented in the context of privacy-compliant software reuse?
- RQ1.5 How is privacy-compliant software reuse evaluated?
- RQ1.6 What are the main open challenges for performing privacy-compliant software reuse?

2. **RQ2**. How are privacy issues discussed and dealt with in open-source environments?
- RQ2.1 How can privacy issues be identified in open-source environments?
- RQ2.2 To what extent can the identification be improved based on privacy ontologies?
- RQ2.3 To what extent do automatically analyzed sentiments of privacy-related issues correlate with privacy compliance of those projects?
- RQ2.4 What are the unique privacy-related characteristics for software projects with a high reuse potential?

3. **RQ3**. How can the privacy compliance of a software artifact be assessed to facilitate and support reuse?
- RQ3.1 How can the privacy compliance of existing software artifacts be assessed?
- RQ3.2 What is the aggregate privacy compliance level when integrating software artifacts?

3. Related work

This section briefly reviews the literature relevant for answering our research questions.

State-of-the-art in privacy compliant software development (RQ1): A few SLR works were performed in the last decade on security aspects in software development. Among them, the works in [16]–[18] discuss security in cyber-physical systems, electronic health records and software development lifecycle in general. Differently from these works, our study concentrates on software reuse which requires assessment and integration of already existing artifacts, evaluating their level of privacy compliance and preserving the level of privacy when being adapted and integrated.

Several systematic reviews on software requirements reuse [19], and on non-functional requirements [20], [21] that inherently include privacy requirements were also performed. However, these works do not concentrate on privacy compliance aspects.

The systematic mapping study in [22] deals with privacy-by-design approaches in software engineering. This work maps the goals of privacy-by-design to software engineering activities. We further aim to analyze the challenges of privacy compliance in software reuse scenarios.

Discussions on privacy aspects in open-source environments (RQ2): Discussions in open-source environments have been researched for different purposes, such as understanding the social potential of those environments, devising on popularity of the different projects and analyzing their maintenance level and required contributions [23]. In the context of privacy and security, the issues usually provide an important source of information on the project. The work in [24] identified several end-user human-centric issues discussed on GitHub: Inclusiveness, Privacy & Security, Compatibility, Location & Language, Preference, Satisfaction, Emotional Aspects, and Accessibility. Some works further analyzed the sentiments of comments and discussions in social networks. The work in [25], for example, found that applying emotion mining to developer issue reports can be useful to identify and monitor the mood of the development team, and thus predict and resolve potential threats to the team well-being, as well as discover factors that enhance team productivity. The work in [26] created a domain-specific tool for sentiment analysis of the issues documented in software development, improving the accuracy of the analysis by enriching the classifier with domain-specific lexicon. The work in [27] reports on a positive correlation between favorable sentiments and improved practices in the context of software engineering and development. We intend to advance this body of research by analyzing the relations of the analyzed sentiments to privacy compliance, and to explore whether meta-data of OSS projects (in particular, issues) can predict privacy compliance.

One of the challenges in performing empirical research on privacy issues in OSS is a lack of previously annotated datasets on the subject, impeding the precision and correctness of privacy issues identification and categorization. An approach to overcome this challenge is utilizing ontology. The researchers in [28] conducted experiments on text classification with various classifiers, both prior to and subsequent to utilizing a disease ontology. Notably, this integration led to enhancement in the outcomes. Moreover, the adoption of a domain-specific ontology has demonstrated improvement in the accuracy of text classification in situations where a sufficiently large and well-labeled training corpus is not at hand [29]. Our research seeks to extrapolate these approaches to the realm of privacy, employing Gharib et al.'s ontology [10].

Privacy compliance assessment (RQ3): Privacy is a multi-faceted concept [30]–[32] that may refer to social, physical, informational and psychological domains. The methods for assessing privacy levels that are described in literature focus mostly on security aspects [33], [34], such as the Common Vulnerability Scoring System (CVSS)[3], an open scoring system of security vulnerabilities and threats. Previous works analyze privacy and security breaches for a specific domain (smart homes) [35], or a specific type of applications (contact tracing) [36]. However, checking an overall privacy compliance

[3] https://www.first.org/cvss/

of software is challenging. Few works focus on the analysis of the software code itself for implementation of privacy requirements [37], [38]. According to [39], a traditional static code analysis-based vulnerability discovery is insufficient for compliance checking of regulatory requirements. A model for automated privacy compliance checking of applications in the cloud is introduced in [40]. Despite the above studies, to the best of our knowledge, there is no end-to-end approach for assessing and measuring privacy compliance level of software artifacts, and we aim to advance this line of research by creating a privacy compliance assessment method.

4. The research methodology and expected research contributions

Figure 1 depicts the main research activities and their outcomes.

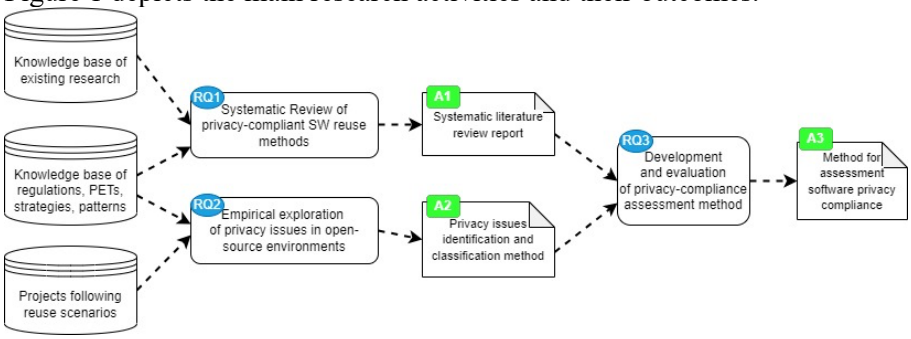

Figure 1: Main research activities and their outcomes

4.1. Systematic literature review of privacy-compliant software reuse

This activity aims to address RQ1 by systematically reviewing and analyzing the existing state-of-the-art privacy-compliant software development, focusing on reuse scenarios. It follows the guidelines for conducting SLRs by Kitchenham [41], complemented by the guidelines on snowball sampling by Wohlin [42] and the guidelines for study selection in PRISMA2020 statement [43].

The SLR search query includes the following concepts with a logical condition of AND between them:

- **Concept 1** (the examined aspect): privacy OR "data protection"
- **Concept 2** (the examined process): reuse OR reusing OR reusab* OR cloning OR clone OR config*
- **Concept 3** (the examined object): software OR product OR system
- **Concept 4** (the examined phase): requirement* OR analys* OR analis* OR analyz* OR analiz* OR domain* OR model* OR design*.

After retrieving the potential papers to be included in the SLR, inclusion and exclusion are applied. The inclusion criteria are: (1) the paper should introduce a technique/method/tool for privacy-compliant software reuse and (2) it should be published in a peer-reviewed journal, conference or workshop. The exclusion criteria are: (1) the paper should not be too short (3 pages or less), (2) the paper is not a variant of another paper in the corpus, (3) the paper is written in English, (4) the paper is not a primary study, (5) the paper is not directly related to software reuse or data privacy, and (5) the full text of the paper is not accessible. We resulted with a corpus of 61 papers which was analyzed to address questions RQ1.x (see Section 5.1 for more details).

4.2. Empirical exploration of privacy issues in open-source environments

To address RQ2, we apply an empirical approach on a sample of open-source projects, with the purpose of identifying privacy aspects in highly reusable projects as compared to those with a lesser reuse potential. Those aspects may appear in different fields (e.g., issues, comments, commits and pull requests). We aim to explore whether machine learning methods can automatically identify privacy-related issues, and whether this method can be enhanced utilizing existing knowledge representations.

We assume that, if the automatically computed outcomes can be mapped to human-generated knowledge base, then large datasets of issues can be automatically analyzed and decisions, e.g., regarding using or reusing the related projects, can be made automatically or semi-automatically, based on privacy requirements, as reflected in issue discussions.

As for devising the reuse potential of the projects, some meta-data available in open-source repositories refer to popularity of projects or users, e.g., the numbers of active forks, stars and followers [44], [45]. The authors of [46] have already demonstrated a relation between popularity of a project to its quality and reuse potential. We plan to analyze the popularity characteristics of a sample of OSS projects, to differentiate between potentially highly reusable and "regular" software projects, and to explore the relations between the privacy characteristics of those two kinds of projects.

While initial results have already been achieved and submitted to a conference (see Section 5.2), we intend to extend the datasets and the explored elements beyond issues. In addition, we intend to devise whether the sentiment of the privacy discussions (positive/neutral/negative) and the subjective privacy compliance level of the project are correlated. We plan to conduct sentiment analysis of the identified privacy issues, and independently, to conduct a survey with OSS project owners or privacy experts to annotate the projects in our dataset to [privacy-]compliant and non-compliant. To this end, we plan to base our survey on the privacy design strategies presented in [7]. The results of those two steps will be compared for correlations using statistical techniques.

4.3. Development of the privacy compliance assessment method

This activity will follow the design science approach [47] and will be based on the results of the SLR (addressing RQ1) and the empirical exploration (addressing RQ2). We will develop techniques for privacy compliance assessments of software artifacts and for presenting those assessments in supporting reuse decisions.

An initial input for the privacy compliance assessment method will be the comparison of the privacy requirements for the regulation compliance of the software artifact and the privacy strategies [7] and/or PETs [48] already implemented in the artifact. Since the discovery of the already implemented privacy strategies and PETs requires detailed technical documentation of the software project and is not always available, our method will incorporate the existing documentation together with the community discussions on and sentiment analysis of the software artifact.

This part of the research has not begun and there are still a few challenges to overcome:

The first challenge refers to the definition of the privacy compliance level of software artifacts. Whilst this level may be either qualitative or quantitative in one or more dimensions, we plan on creating a multi-dimensional qualitative scoring method. The method will include, as a first step, a scoring scale for applying PETs; as a second step mining of relevant privacy strategies will be performed for the specific software artifact, based on the known privacy requirements; next, extraction of already applied PETs will be performed based on the project documentation and meta-data; and finally, the score for the software artifact privacy compliance level will be devised, based on the previous steps and additional data such as sentiment analysis.

An additional challenge is the level at which the privacy compliance level will be calculated – e.g., a project, a component or a function. Currently, our plans refer to whole projects, since most of the data on which we base our analysis is managed on the project level. However, as the privacy compliance level of the different parts of a project may differ, we will have to check the impact of such a score on lower levels of granularity (e.g., components and functions). In addition, many of the reuse scenarios deal with reuse on lower levels of granularity by integrating third-party components, libraries and/or functions into existing projects to resolve existing issues and to create additional functionality. The challenge is to assess the impact of those integrations on the overall privacy compliance level of the software project.

Finally, the suggested method has to be evaluated for usability and generality, and the access to evaluation objects may be challenging [49]. One of the main ways of evaluating method usability and usefulness is with experts. To overcome this challenge, we plan to conduct expert evaluation of the method itself, or its outcomes, with professionals and/or advanced students.

4.4. Expected contributions

The contributions of the work are valuable both for research and practice.

From a research point of view, the research artifacts will enrich the knowledge base with:

A1. A systematic review (RQ1) that will examine the current state-of-the-art and present the results of a comparative analysis of privacy-compliant software reuse methods. Special attention will be given to the reuse approaches and privacy strategies of these methods. The SLR will also identify the contemporary challenges for performing privacy-compliant software reuse and future directions.

A2. An ontology-guided machine-learning method (RQ2) for identifying and classifying privacy-related discussions in the form of issues and additional meta-data items in open-source repositories. The method will also devise the potential privacy compliance of projects based on the sentiment of the privacy discussions in the OSS and opinions of experts involved in the projects, for projects with different reuse potential.

A3. A method for assessing the privacy compliance level of software projects (RQ3) that will support the reuse lifecycle and improve privacy compliance of those projects by making sure the additional software artifacts integrated into the projects do not decrease the current compliance level.

From a practical point of view, we plan on supporting privacy-compliant software reuse by applying the method in A3 on third-party and open-source artifacts and assessing their privacy compliance level to select the most appropriate artifacts for reuse. This will increase the level of privacy compliance in complex software systems that make extensive reuse of software artifacts.

5. Progress and preliminary results

We so far concentrated on the two first research questions, forming the basis for RQ3. Below is a summary of the achievements.

5.1. The current state of the systematic literature review

This part was completed, and its outcomes were sent as a paper to the IST (Information & Software Technology) journal. It is currently under minor revision. We found that the reviewed 61 studies vary in terms of business domains (e.g., healthcare, smart objects and finance) and technological domains (e.g., IoT, mobile, cloud and microservices). Most of the studies do not refer to a specific regulation and if so – to GDPR. Their common purpose is to support benign reuse, most notably through patterns, components & libraries and model-driven engineering, but malicious reuse is also researched to a lesser extent. A strong emphasis is put on integrating privacy strategies whose goal is building trust and transparency (in particular, inform and demonstrate), while other strategies are studied to a limited extent in software reuse context. Evaluation is commonly performed through analytical, observational and experimental approaches.

We further found that the assessment and operationalization of privacy compliance practices for existing software artifacts is still challenging. The challenges encompass improving trustworthiness of reused artifacts, ensuring privacy compliance in distributed architectures, bridging the gap between legal regulations and software requirements, enhancing privacy analysis and vulnerability detection, supporting late application of privacy strategies, and developing objective assessments for privacy-compliant software reuse.

5.2. The current state of the empirical exploration of privacy issues

So far, we studied to what extent machine learning outcomes can be mapped to privacy-related knowledge representations for identifying and categorizing privacy-related issues in open-source environments. We explored a dataset of 2,556 issues from open-source projects. 1,374 of them (about 54%) are issue reports from Jira associated with two large-scale, popular and well-maintained software projects, Chrome and Moodle, used in [11] and annotated as privacy-related. The other 1,182 are issues from six diverse projects in GitHub annotated by [24] as non-privacy related.

First, we preprocessed the dataset by means of text cleaning, removing irrelevant parts (e.g., html tags, numbers, punctuation marks and stop words) and performing tokenization and lemmatization [50]. We extracted words from URLs that appear in the issues and left them for the analyses, assuming that some may be meaningful and relevant for classification. Then, we used YAKE! keyword extractor [51] for extracting potential key-terms from the preprocessed text. We applied Reduced Error Pruning (REP) tree [52] and Support Vector Machine (SVM) [53] classifiers. We chose these classifiers because they differ in their underlying principles, learning approach, and decision-making processes. The metrics of the two classifiers were compared to validate that the results are similar in terms of correctly classified items, precision, recall and F-measure, indicating that similar patterns or relationships in the data have been learned. As indicated by the relatively high values of F-measure for both classifiers (86.6% for REP tree and 87.1% for SVM), we can conclude that the classifiers have the potential for distinguishing between privacy and non-privacy issues. Due to the simple, intuitive and visual format of REP tree outcomes, we used them to identify key-terms for classification of the issues into one of two classes – privacy and non-privacy related. Lastly, we succeeded in manually mapping the identified key terms to the main concepts of the ontology in [10]. The root term, privacy, appeared in only 20% of privacy-related issues, so the relatively high accuracy of the classifier cannot be attributed only to this term. The terms user, datum and setting can be considered organizational; tool policy and tool data privacy relate to policies or guidelines for data privacy handling; and incognito window, incognito mode and [third] party cookie – to privacy-related risks.

Acknowledgements

I am grateful to my advisor Professor Iris Reinhartz-Berger for supervising my research, for countless inspirational discussions and for being available for me almost around the clock. I really appreciate this enormous dedication.

References

[1] G. Danezis *et al.*, *Privacy and data protection by design - from policy to engineering*, no. December. 2015. doi: 10.2824/38623.

[2] "EU Regulation 2016/679 of the European Parliament and of the Council," *Official Journal of the European Union*, 2016. https://gdpr.eu/ (accessed Feb. 17, 2022).

[3] Y. S. Martin and A. Kung, "Methods and tools for GDPR compliance through privacy and data protection engineering," *Proc. - 3rd IEEE Eur. Symp. Secur. Priv. Work. EURO S PW 2018*, pp. 108–111, 2018, doi: 10.1109/EuroSPW.2018.00021.

[4] J. J. Borking and C. D. Raab, "Laws, PETs and other technologies for privacy protection," *J. Information, Law Technol.*, 2001.

[5] S. D. Ringmann, H. Langweg, and M. Waldvogel, "Requirements for legally compliant software based on the GDPR," vol. 11230 LNCS. 2018. doi: 10.1007/978-3-030-02671-4_15.

[6] A. Cavoukian, "Privacy by design: The 7 foundational principles," *Priv. by Des. Canada*, vol. 3, no. 2, pp. 247–251, 2010.

[7] J.-H. Hoepman, "Privacy design strategies," in *SEC 2014, IFIP AICT 428*, 2014, pp. 446–459.

[8] A. J. Aberkane, G. Poels, and S. Vanden Broucke, "Exploring automated GDPR-compliance in requirements engineering: A systematic mapping study," *IEEE Access*, vol. 9, pp. 66542–66559, 2021, doi: 10.1109/ACCESS.2021.3076921.

[9] H. van Rossum *et al.*, "Privacy-enhancing technologies: The path to anonymity," vol. I, no. Volume I, pp. 1–60, 1995.

[10] M. Gharib, P. Giorgini, and J. Mylopoulos, "Towards an ontology for privacy requirements via a systematic literature review," *ER*, 2017, doi: 10.1007/s13740-020-00116-5.

[11] P. Sangaroonsilp, H. K. Dam, M. Choetkiertikul, C. Ragkhitwetsagul, and A. Ghose, "A taxonomy for mining and classifying privacy requirements in issue reports," *Inf. Softw. Technol.*, vol. 157, 2023, doi: 10.1016/j.infsof.2023.107162.

[12] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, and W. Joosen, "A privacy threat analysis framework: Supporting the elicitation and fulfillment of privacy requirements," *Requir. Eng.*,

vol. 16, no. 1, pp. 3–32, 2011, doi: 10.1007/s00766-010-0115-7.
[13] R. M. de Carvalho *et al.*, "Protecting citizens' personal data and privacy: Joint effort from GDPR EU cluster research projects," *SN Comput. Sci.*, vol. 1, no. 4, pp. 1–16, 2020, doi: 10.1007/s42979-020-00218-8.
[14] B. Kostova, S. Gürses, and C. Troncoso, "Privacy engineering meets software engineering. On the challenges of engineering privacy by design," 2020, [Online]. Available: http://arxiv.org/abs/2007.08613
[15] M. Lungu, R. Robbes, and M. Lanza, "Recovering inter-project dependencies in software ecosystems," *ASE'10 - Proc. IEEE/ACM Int. Conf. Autom. Softw. Eng.*, pp. 309–312, 2010, doi: 10.1145/1858996.1859058.
[16] N. M. Mohammed, M. Niazi, M. Alshayeb, and S. Mahmood, "Exploring software security approaches in software development lifecycle: A systematic mapping study," *Comput. Stand. Interfaces*, vol. 50, no. October 2016, pp. 107–115, 2017, doi: 10.1016/j.csi.2016.10.001.
[17] P. H. Nguyen, S. Ali, and T. Yue, "Model-based security engineering for cyber-physical systems : A systematic mapping study," *Inf. Softw. Technol.*, vol. 83, pp. 116–135, 2017, doi: 10.1016/j.infsof.2016.11.004.
[18] J. Luis Fernandez-Aleman, I. Carrion Senor, P. A. Oliver Lozoya, and A. Toval, "Security and privacy in electronic health records: A systematic literature review," *J. Biomed. Inform.*, vol. 46, no. 3, pp. 541–562, 2013, doi: 10.1016/j.jbi.2012.12.003.
[19] M. Irshad, K. Petersen, and S. Poulding, "A systematic literature review of software requirements reuse approaches," *Information and Software Technology*, vol. 93. Elsevier B.V., pp. 223–245, Jan. 01, 2018. doi: 10.1016/j.infsof.2017.09.009.
[20] M. Glinz, "On non-functional requirements," *Proc. - 15th IEEE Int. Requir. Eng. Conf. RE 2007*, 2007, doi: 10.1109/RE.2007.45.
[21] N. Afreen, A. Khatoon, and M. Sadiq, "A taxonomy of software's non-functional requirements," in *Proceedings of the Second International Conference on Computer and Communication Technologies*, 2016. doi: 10.1007/978-81-322-2517-1.
[22] M. E. Morales-Trujillo, E. O. Matla-Cruz, G. A. García-Mireles, and M. Piattini, "Privacy by design in software engineering: A systematic mapping study," *Av. en Ing. Softw. a Niv. Iberoam. CIbSE 2018*, vol. 22, no. 1, pp. 107–120, 2018.
[23] E. Kalliamvakou, G. Gousios, K. Blincoe, L. Singer, D. M. German, and D. Damian, "An in-depth study of the promises and perils of mining GitHub," *Empir. Softw. Eng.*, vol. 21, no. 5, pp. 2035–2071, 2016, doi: 10.1007/s10664-015-9393-5.
[24] H. Khalajzadeh, M. Shahin, H. O. Obie, and J. Grundy, *How are diverse end-user human-centric issues discussed on GitHub?* Association for Computing Machinery, 2022.
[25] A. Murgia and B. Adams, "Do developers feel emotions ? An exploratory analysis of emotions in software artifacts," in *MSR 2014*, 2014, pp. 262–271. doi: 10.1145/2597073.2597086.
[26] J. Ding, H. Sun, X. Wang, and X. Liu, "Entity-level sentiment analysis of issue comments," in *SEmotion' 18:IEEE/ACM 3rd International Workshop on Emotion Awareness in Software Engineering*, 2018, pp. 7–13. doi: 10.1145/3194932.3194935.
[27] R. S. C. Junior and G. D. F. Carneiro, "Impact of developers sentiments on practices and artifacts in open source software projects : A systematic literature review," in *Proceedings of the 22nd International Conference on Enterprise Information Systems (ICEIS 2020)*, 2020, vol. 2, pp. 978–989. doi: 10.5220/0009313200310042.
[28] S. Malik and S. Jain, "Semantic ontology-based approach to enhance text classification," in *ISIC 2021*, 2021.
[29] N. Sanchez-pi, L. Martí, A. Cristina, and B. Garcia, "Improving ontology-based text classification : An occupational health and security application," *J. Appl. Log.*, vol. 17, pp. 48–58, 2016, doi: 10.1016/j.jal.2015.09.008.
[30] H. Leino-kilpi *et al.*, "Privacy : a review of the literature," *Int. J. Nurs. Stud.*, vol. 38, 2001.
[31] V. Demertzi, S. Demertzis, and K. Demertzis, "An overview of privacy dimensions on Industrial Internet of Things (IIoT)," *arXiv Prepr. arXiv2301.06172.*, pp. 1–17, 2023.
[32] A. Martínez-ballesté, P. A. Pérez-martínez, and A. Solanas, "The pursuit of citizens' privacy : A privacy-aware smart city is possible," *IEEE Commun. Mag.*, no. June, pp. 136–141, 2013, doi: 10.1109/MCOM.2013.6525606.

[33] P. Mell, K. Scarfone, and S. Romanosky, “Common vulnerability scoring system,” *IEEE Secur. Priv.*, vol. 4, no. 6, pp. 85–89, 2006, doi: 10.1109/MSP.2006.145.
[34] P. Mell, “The generation of software security scoring systems leveraging human expert opinion,” *2022 IEEE 29th Annu. Softw. Technol. Conf.*, pp. 116–124, 2022, doi: 10.1109/STC55697.2022.00023.
[35] J. S. Edu, J. M. Such, and G. Suarez-tangil, “Smart home personal assistants : A security and privacy review,” *ACM Comput. Surv.*, vol. 53, no. 6, 2020, doi: 10.1145/3412383.
[36] L. Krehling and A. Essex, “A security and privacy scoring system for contact tracing apps,” *J. Cybersecurity Priv.*, vol. 1, pp. 597–614, 2021.
[37] S. Zimmeck *et al.*, “Automated analysis of privacy requirements for mobile apps,” in *The 2016 AAAI Fall Symposium Series: Privacy and Language Technologies Technical Report FS-16-04*, 2016, vol. 3066, no. 132, pp. 286–296.
[38] M. Tahaei, A. Frik, and K. Vaniea, “Privacy champions in sofware teams : Understanding their motivations , strategies , and challenges,” in *CHI Conference on Human Factors in Computing Systems (CHI ’21)*, 2021. doi: 10.1145/3411764.3445768.
[39] M. Farhadi, H. Haddad, and H. Shahriar, “Compliance checking of open source EHR applications for HIPAA and ONC security and privacy requirements,” in *2019 IEEE 43rd Annual Computer Software and Applications Conference (COMPSAC)*, 2019, vol. 1, pp. 704–713. doi: 10.1109/COMPSAC.2019.00106.
[40] M. Farhadi, G. Pierre, and D. Miorandi, “Towards automated privacy compliance checking of applications in Cloud and Fog environments,” *2021 8th Int. Conf. Futur. Internet Things Cloud*, pp. 11–18, 2021, doi: 10.1109/FiCloud49777.2021.00010.
[41] B. Kitchenham, O. Pearl Brereton, D. Budgen, M. Turner, J. Bailey, and S. Linkman, “Systematic literature reviews in software engineering - A systematic literature review,” *Inf. Softw. Technol.*, vol. 51, no. 1, pp. 7–15, 2009, doi: 10.1016/j.infsof.2008.09.009.
[42] C. Wohlin, “Guidelines for snowballing in systematic literature studies and a replication in software engineering,” *ACM Int. Conf. Proceeding Ser.*, 2014, doi: 10.1145/2601248.2601268.
[43] M. J. Page *et al.*, “The PRISMA 2020 statement: An updated guideline for reporting systematic reviews,” *BMJ*, vol. 372, 2021, doi: 10.1136/bmj.n71.
[44] M. E. Paschali, A. Ampatzoglou, S. Bibi, A. Chatzigeorgiou, and I. Stamelos, “Reusability of open source software across domains: A case study,” *J. Syst. Softw.*, vol. 134, pp. 211–227, 2017, doi: 10.1016/j.jss.2017.09.009.
[45] M. D. Papamichail, T. Diamantopoulos, and A. L. Symeonidis, “Measuring the reusability of software components using static analysis metrics and reuse rate information,” *J. Syst. Softw.*, vol. 158, p. 110423, 2019, doi: 10.1016/j.jss.2019.110423.
[46] F. Kunz and Z. A. Mann, “Finding risk patterns in cloud system models,” *IEEE Int. Conf. Cloud Comput. CLOUD*, vol. 2019-July, no. Vm, pp. 251–255, 2019, doi: 10.1109/CLOUD.2019.00051.
[47] A. R. Hevner, S. T. March, J. Park, and S. Ram, “Design science in information systems research,” *MIS Q.*, vol. 28, no. 1, pp. 75–105, 2004, doi: https://doi.org/10.2307/25148625.
[48] J. Heurix, P. Zimmermann, T. Neubauer, and S. Fenz, “A taxonomy for privacy enhancing technologies,” *Comput. Secur.*, vol. 53, pp. 1–17, 2015, doi: 10.1016/j.cose.2015.05.002.
[49] F. Dervin and C. Dyer, *Constructing methodology for qualitative research*. 2016. doi: 10.1057/978-1-137-59943-8.
[50] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python*. O’Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472, 2009.
[51] R. Campos, V. Mangaravite, A. Pasquali, A. Jorge, C. Nunes, and A. Jatowt, “YAKE! Keyword extraction from single documents using multiple local features,” *Inf. Sci. (Ny).*, vol. 509, pp. 257–289, 2020, doi: 10.1016/j.ins.2019.09.013.
[52] J. R. Quinlan, “Simplifying decision trees,” *Int. J. Hum. Comput. Stud.*, vol. 27, pp. 221–234, 1987, doi: 10.1006/ijhc.1987.0321.
[53] C. Cortes and V. Vapnik, “Support-vector networks,” *Mach. Learn.*, vol. 20, pp. 273–297, 1995.