

# An Intelligent Systems Approach for Supporting Privacy Awareness in Agile Software Development

Guntur Budi Herwanto<sup>1,2</sup>

<sup>1</sup>Faculty of Computer Science, University of Vienna

<sup>2</sup>Department of Computer Science and Electronics, Universitas Gadjah Mada

## Abstract

Privacy by design principles is an established standard guiding the design and development of privacy-aware systems. Privacy engineering acts as a role to close the gap between the privacy policy and the realization of the system or technology that will be developed. Many privacy engineering methodologies depend heavily on a waterfall-style approach that can be very time-consuming and is not tailored to the speed of agile process, which the majority of the industry is currently taking. In this research, we aim to address those challenges by an intelligent system approach in the form of a natural language processing and recommendation system. As a scientific basis, we use experimental design research to evaluate our intelligent systems that will be integrated in privacy requirements and design context. With this research, we intend to contribute to the advancement of privacy engineering in an agile environment by providing a system that allows better integration of privacy protection with currently used development processes, such as Scrum.

## Keywords

privacy engineering, privacy requirement, agile software development, intelligent system

## 1. Introduction

Tailoring privacy aspects into the software development process has become a key concern and challenge for the industry. Privacy engineering has emerged as a research framework that focuses on adapting privacy into organizational and technical measures [1]. Privacy engineering is integrated into the software development life cycle (SDLC), including the requirements and design phase. The requirements phase can therefore be referred to as privacy requirement engineering. Many privacy requirements engineering methodologies depend heavily on a waterfall-style approach that can be time-consuming and not tailored to the agile speed that much of the industry is currently taking. Researchers have studied these challenges and clearly state the conflicting nature of agile software development (ASD) and privacy engineering [2].

The agile turn makes modeling privacy threats or designing a privacy-aware system become more challenging [3]. According to the findings of a study on legal compliance within agile teams, the teams did not know how to identify privacy principles in user requirements [4]. They


---

In: J. Fischbach, N. Condori-Fernández, J. Doerr, M. Ruiz, J.-P. Steghöfer, L. Pasquale, A. Zisman, R. Guizzardi, J. Horkoff, A. Perini, A. Susi, M. Daneva, A. Herrmann, K. Schneider, P. Mennig, F. Dalpiaz, D. Dell'Anna, S. Kopczyńska, L. Montgomery, A. G. Darby, and P. Sawyer (eds.): *Joint Proceedings of REFSQ-2022 Workshops, Doctoral Symposium, and Poster & Tools Track*, Birmingham, UK, 21-03-2022, published at <http://ceur-ws.org>

✉ [gunturbudi@ugm.ac.id](mailto:gunturbudi@ugm.ac.id) (G. B. Herwanto)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

believe it is challenging to incorporate privacy considerations into the development process [4]. To overcome this, researchers are developing the method that can suit to the agile process [5], or proposing the lean process on privacy engineering. [6]. Tool support is also proposed as a way to capture the privacy requirement in agile situation [7].

Intelligent software engineering (ISE) has long been used in ASD [8]. Perkusich [8] refers ISE as "*the application of intelligent techniques to software engineering*". In addition, they defined intelligent technique as "*a technique that explores data (from digital artifacts or domain experts) for knowledge discovery, reasoning, learning, planning, natural language processing, perception or supporting decision-making*". The implementation of the ISE has the purpose of helping better manage and even accelerate the agile process, including software requirement and design [8]. Advances in intelligent techniques, such as natural language processing (NLP), have accelerated the adoption of intelligent techniques in requirements engineering [9, 10].

The potential of ISE, however, has not been realized to assist privacy engineering in ASD. The empirical research on agile teams found that they prefer to be assisted with techniques and tools to perform privacy requirement elicitation [4]. They also get some difficulties in capturing privacy aspects in textual user stories [4]. There is a clear opportunity to bridge the gap between the advancement of IS techniques such as NLP with privacy requirement engineering [11]. We hypothesize that IS-enabled tools will be able to overcome these challenges. To the best of our knowledge, there is still a lack of research that sees the implementation of IS in privacy engineering.

## **2. Problem Statement**

The privacy engineering aspect of this research is focused on the requirement and design phases. Several methodologies exist to elicit privacy requirements, including threat modeling and privacy impact assessment. However, as indicated in the introduction, agile teams continue to face challenges with incorporating this into the development process [4]. Therefore, we intend to build an intelligent system (IS) that is able to assist agile teams in privacy requirements and design activities. We begin our research by investigating how IS might be used to help the requirement and design phases of agile privacy engineering. This includes identifying the current practice of privacy requirements and design engineering, as well as the kind of intelligent techniques suitable for supporting it.

Obtaining privacy requirements typically requires following several steps, such as identifying assets, identifying personal data, modeling the system, analyzing data flow, identifying threats, and eliciting requirements according to specific privacy principles. The amount of work required for some of these processes is considered a challenge by some software teams [2]. Once the privacy requirements are defined, it can be difficult for software teams to choose from a number of design patterns or privacy enhancement technologies (PET) in order to meet the requirements. An empirical evaluation of our IS technique must be performed to demonstrate the methodology's rigor. The intelligent technique that we will implement must meet specific criteria, such as recall [12], so that it can be used to support agile teams.

### 3. Relevance

The effort to integrate privacy in the software development process has been studied by providing frameworks and methodologies. One of the most popular research to treat privacy is the risk management approach. However, the practice of risk management is still conducted in a traditional plan-driven way [13]. The approach to address privacy in the ASD context has been made in a limited amount of research [14, 7]. PRIPARE project has provided a handbook for applying privacy by design to ASD [14], while OASIS projects focus on the documentation [15]. PRIPARE project proposes incremental privacy and security, sprint zero, and privacy security sprints, while OASIS project mentions applying proactive and iterative, which derived the first principle of the original privacy by design.

Privacy is considered to be closely linked to security and risk management in software systems [16]. The research by Dam et al. [17] tries to automatically inspect code vulnerabilities to reduce the risk of security infringement. They use a deep learning model called Long Short Term Memory (LSTM) to learn the semantic and syntactic representation of the code that can lead to vulnerabilities. The use of the deep learning model has been shown to outperform the previous approach for some intelligent system tasks.

The potential for automating the privacy process has been studied in several research [18, 11]. Study by Zimmermann [18] identifies the potential of automation in a privacy engineering context, especially the privacy impact assessment process. They argued that automatic selection of privacy patterns for a given set of specifications is advantageous to the privacy engineer. Aberkane [11] explicitly mentions the potential use of the Natural Language Processing model to support the privacy requirement engineering. Utilizing the reusable elements [19] from design patterns, techniques, methods or tools can help achieve this goal. Design patterns claimed to play an important role on enabling adaptation to privacy compliance [20]. However, software practitioners still had difficulties connecting the requirement of regulation with a suitable technical measure such as design pattern [21]. Recommendation systems ought to be the solution to the array of choices in the design pattern. Semi-automation approach through an online questionnaire or wizard has been studied and implemented by Colesky et al. [22] to get a suitable privacy design pattern based on the knowledge base [23]. Nevertheless, an integrated solution that streamlines assets, data flow, and privacy requirements while providing an appropriate privacy design pattern can be beneficial.

Incorporating intelligent systems to ASD has been studied comprehensively in a systematic literature review conducted by Perkusich [8]. Most of these techniques are applied to support the management of ASD, such as improving the estimation of effort and delay risk prediction [24]. Intelligent systems also impact the other areas of ASD, such as software requirements, software design, software quality, and testing with a lesser amount of study. Regarding security in ASD, only one study uses fuzzy logic to combine security activities with ASD [25]. In a more recent research, Villazimar [26] uses NLP to match a text feature from a user story with security properties and security requirements. However, a specific study on applying the intelligent system to privacy in ASD is not mentioned in the study [8].

## 4. Research Method

This research's main objective is to provide an intelligent system to help the agile team integrate privacy aspects into their software system. Tool support would be necessary to allow an easy adaptation for agile teams. Therefore, we took the design science research framework, which aims to create new and innovative artifacts [27]. Along with the design science research, we aim to use the experimental design [28] method to measure our objectives. We plan to answer the first problem in several cycles, which targets the requirement and design phase in privacy engineering. In each cycle, an evaluation will be conducted to ensure the rigor of the proposed artifact.

We propose an Intelligent System (IS) concept to support decision-making on privacy requirements by utilizing the reusable knowledge from privacy framework, privacy patterns, and legal requirements. Privacy as one of the requirements in software development already has an abundance of reusable knowledge [19]. Tools that can assist requirement analyst in managing the requirement is needed. Connecting IS with reusable privacy knowledge is the focus of the proposed approach. The IS is in the form of (1) Personal data identification from user stories, (2) Automatic Data Flow Diagram Generation, (3) Privacy requirement recommendation systems, and (4) Privacy design pattern recommendation system.

The success of the proposed approach will be measured through experimental design. The system performance for automatic privacy entities detection, automatic data flow diagram generation, privacy requirement recommendation, and design recommendation system will be evaluated using precision, recall, and F-Measure. We will perform a controlled experiment on the requirement and design phase to compare the speed, leanness, and learning when intelligent systems are incorporated to build privacy-aware software systems.

## 5. Towards An Intelligence Systems Approach

Our solution centered around the user stories as the primary input. Thus, we mainly use the Natural Language Processing (NLP) methods to identify the needs of privacy requirements. The final output is a set of privacy requirements and design recommendations to support privacy integration into software under development. We aim to reduce the manual effort on the privacy engineering process by assisting it with some automatic approaches.

The first module detects the personal data attributes in the user story using Named Entity Recognition (NER). This becomes the basis for the generation of Privacy-Aware Data Flow Diagram (DFD). The technical measure or solution for privacy requirements can be derived from mapping the DFD elements with the privacy-enhancing technologies (PET) and privacy patterns. LINDDUN [29] has provided the mapping from privacy requirement to the suggested technical solution in the form of PET. In terms of privacy patterns, a well-documented catalog is already established in [privacypattern.org](http://privacypattern.org). Our work will combine the previously known knowledge base to recommend a suitable technical solution. Our approach tries to minimize the presents of data privacy experts or prior knowledge about privacy. The recommendation system will use a content-based recommendation system based on the similarity between the requirement and the description of PET and the Privacy Pattern. In addition, we will also use

the knowledge-based recommendation system in the form ontology-based recommendation system and constraint-based recommendation system.

## 6. Novelty

Integrating privacy into agile software development is still an open problem. According to Aberkane, the use of NLP can become a potential solution for automating some processes in GDPR compliance [11]. We aim to solve this gap by proposing IS approaches such as NLP and recommendation systems targeting privacy engineering activities, which are requirement generation and threat modeling [3, 30]. Additionally, we hope to streamline the privacy requirement into a usable design pattern through the use of a knowledge base. We build the IS artifact using a design science framework. Lastly, we intend to conduct a novel evaluation that assesses our IS adaptation using experimental design [28].

## 7. Progress

The first phase of our research is to obtain knowledge bases (KB) to support the information systems artifacts that we will build. We intend to use publicly accessible knowledge, including frameworks, instruments, methods, dictionaries, and data sets. We divide the initial phase into four iterations.

The first iteration is dedicated to developing the knowledge base for the NER module. We identify personal data by referencing dictionaries offered by Data Privacy Vocabulary teams. Additionally, we are developing our annotation data set to recognize data subjects and process entities in user stories. Based on this knowledge base, we create our first information system artifact powered by a state-of-the-art machine learning model for NER. This article was published in ESPRE 2021 [31]. The second iteration will utilize the NER module to create a Privacy-Aware Data Flow Diagram (DFD). We use publicly available tools to generate the relationship between entities in a collection of user stories. We conducted a preliminary assessment and will publish our research preview in the REFSQ 2022. In the future, we intend to perform additional evaluations. The third and fourth iterations will develop an information system model to support privacy requirements and design recommendations. These third and fourth iterations are a work in progress.

Our second phase of research will determine the influence of our privacy engineering approach on ASD, particularly in terms of agility degree. Since we believe that this type of empirical evaluation is new to agile privacy engineering, we would like to leverage the knowledge and insights of the doctoral symposium community regarding this evaluation.

## 8. Conclusions

The purpose of this proposal is to establish a research agenda for incorporating privacy into agile software development methodologies through the use of an intelligent system. We offer a novel perspective on the ISE model's privacy requirements and design phases. Our primary objective is to create an artifact that enables agile teams to rapidly meet desirable privacy and

design recommendations. We also aim to streamline the requirement into the correct privacy design pattern. We hope this research will have an impact to provide a solution to overcome the challenge of integrating privacy into the agile software development process.

## 9. Acknowledgments

The author acknowledge the scholarship granted by the Indonesia Endowment Fund for Education (IEFE/LPDP), Ministry of Finance, Republic of Indonesia and the support received from the University of Vienna, Faculty of Computer Science.

## References

- [1] S. Gürses, J. M. Del Alamo, Privacy Engineering: Shaping an Emerging Field of Research and Practice, *IEEE Security and Privacy* 14 (2016) 40–46. doi:10.1109/MSP.2016.37.
- [2] B. Kostova, S. Gürses, C. Troncoso, Privacy engineering meets software engineering, on the challenges of engineering privacy by design (2020). URL: <http://arxiv.org/abs/2007.08613>.
- [3] R. Galvez, S. Gurses, The Odyssey: Modeling Privacy Threats in a Brave New World, *Proceedings - 3rd IEEE European Symposium on Security and Privacy Workshops, EURO S and PW 2018* (2018) 87–94. doi:10.1109/EuroSPW.2018.00018.
- [4] E. D. Canedo, A. Toffano Seidel Calazans, A. J. Cerqueira, P. H. Teixeira Costa, E. T. Seidel Masson, Agile teams' perception in privacy requirements elicitation: Lgpd's compliance in brazil, in: *2021 IEEE 29th International Requirements Engineering Conference (RE)*, 2021, pp. 58–69. doi:10.1109/RE51729.2021.00013.
- [5] D. Le, M. Inria, I. K. Trilateral, J. María, PRIPARE Privacy- and Security-by-Design Methodology Handbook (2015). URL: <http://pripareproject.eu/>.
- [6] J. Zibuschka, C. Zimmermann, Lean privacy by design, *SICHERHEIT 2020* (2020).
- [7] M. M. Peixoto, Privacy requirements engineering in agile software development: A specification method, *CEUR Workshop Proceedings* 2584 (2020).
- [8] M. Perkusich, L. Chaves e Silva, A. Costa, F. Ramos, R. Saraiva, A. Freire, E. Dilenzo, E. Dantas, D. Santos, K. Gorgônio, H. Almeida, A. Perkusich, Intelligent software engineering in the context of agile software development: A systematic literature review, *Information and Software Technology* 119 (2020) 106241. URL: <https://doi.org/10.1016/j.infsof.2019.106241>. doi:10.1016/j.infsof.2019.106241.
- [9] L. Zhao, W. Alhoshan, A. Ferrari, K. J. Letsholo, M. A. Ajagbe, E. V. Chioasca, R. T. Batista-Navarro, Natural Language Processing for Requirements Engineering, *ACM Computing Surveys* 54 (2021) 1–41. doi:10.1145/3444689.
- [10] I. K. Raharjana, D. Siahaan, C. Fatichah, User Stories and Natural Language Processing: A Systematic Literature Review, *IEEE Access* 9 (2021) 53811–53826.
- [11] A.-J. Aberkane, G. Poels, S. V. Broucke, Exploring automated gdpr-compliance in requirements engineering: A systematic mapping study, *IEEE Access* 9 (2021) 66542–66559. doi:10.1109/ACCESS.2021.3076921.
- [12] D. M. Berry, Evaluation of Tools for Hairy Requirements and Software Engineering Tasks,

- in: 2017 IEEE 25th International Requirements Engineering Conference Workshops (REW), 2017, pp. 284–291. doi:10.1109/REW.2017.25.
- [13] M. Marinho, J. Noll, I. Richardson, S. Beecham, Plan-driven approaches are alive and kicking in agile global software development, CoRR abs/1906.08895 (2019). URL: <http://arxiv.org/abs/1906.08895>. arXiv:1906.08895.
- [14] N. Notario, A. Crespo, Y. S. Martin, J. M. Del Alamo, D. L. Metayer, T. Antignac, A. Kung, I. Kroener, D. Wright, PRIPARE: Integrating privacy best practices into a privacy engineering methodology, Proceedings - 2015 IEEE Security and Privacy Workshops, SPW 2015 (2015) 151–158. doi:10.1109/SPW.2015.22.
- [15] A. Cavoukian, D. Jutla, F. Carter, J. Sabo, F. Dawson, J. Fox, T. Finneran, S. Fieten, Privacy by design documentation for software engineers version 1.0, PbD-SE) Burlington, MA: Organization for the Advancement of Structured Information Standards (OASIS), work in progress (2014).
- [16] S. Joyee De, D. Le Métayer, PRIAM: A privacy risk analysis methodology, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 9963 LNCS (2016) 221–229. doi:10.1007/978-3-319-47072-6\_15.
- [17] H. K. Dam, T. Tran, T. T. M. Pham, S. W. Ng, J. Grundy, A. Ghose, Automatic feature learning for predicting vulnerable software components, IEEE Transactions on Software Engineering (2018) 1–1.
- [18] C. Zimmermann, Automation potentials in privacy engineering, in: H. Roßnagel, C. H. Schunck, S. Mödersheim, D. Hühnlein (Eds.), Open Identity Summit 2020, Gesellschaft für Informatik e.V., Bonn, 2020, pp. 121–132. doi:10.18420/ois2020\_10.
- [19] J. C. Caiza, Y. S. Martín, D. S. Guamán, J. M. Del Alamo, J. C. Yelmo, Reusable Elements for the Systematic Design of Privacy-Friendly Information Systems: A Mapping Study, IEEE Access 7 (2019) 66512–66535. doi:10.1109/ACCESS.2019.2918003.
- [20] M. Colesky, J. C. Caiza, A system of privacy patterns for informing users: Creating a pattern system, ACM International Conference Proceeding Series (2018). doi:10.1145/3282308.3282325.
- [21] I. Hadar, T. Hasson, O. Ayalon, E. Toch, M. Birnhack, S. Sherman, A. Balissa, Privacy by designers: software developers’ privacy mindset, Empirical Software Engineering 23 (2018) 259–289. doi:10.1007/s10664-017-9517-1.
- [22] M. Colesky, K. Demetzou, L. Fritsch, S. Herold, Helping Software Architects Familiarize with the General Data Protection Regulation, Proceedings - 2019 IEEE International Conference on Software Architecture - Companion, ICSA-C 2019 (2019) 226–229. doi:10.1109/ICSA-C.2019.00046.
- [23] N. Doty, M. Gupta, J. Zych, [privacypatterns.org](http://privacypatterns.org) - Privacy Patterns, 2015. URL: <http://privacypatterns.org/>.
- [24] H. K. Dam, Artificial intelligence for software engineering, XRDS: Crossroads, The ACM Magazine for Students 25 (2019) 34–37. doi:10.1145/3313117.
- [25] S. K. Bansal, A. Jolly, An encyclopedic approach for realization of security activities with agile methodologies, Proceedings of the 5th International Conference on Confluence 2014: The Next Generation Information Technology Summit (2014) 767–772. doi:10.1109/CONFLUENCE.2014.6949242.

- [26] H. Villamizar, M. Kalinowski, A. Garcia, D. Mendez, An efficient approach for reviewing security-related aspects in agile requirements specifications of web applications, *Requirements Engineering* 25 (2020) 439–468. URL: <https://doi.org/10.1007/s00766-020-00338-w>. doi:10.1007/s00766-020-00338-w.
- [27] A. R. Hevner, S. T. March, J. Park, S. Ram, Design science in information systems research, *MIS quarterly* (2004) 75–105.
- [28] S. L. Pfleeger, Experimental design and analysis in software engineering, *Annals of Software Engineering* 1 (1995) 219–253.
- [29] M. Deng, K. Wuyts, R. Scandariato, B. Preneel, W. Joosen, A privacy threat analysis framework: supporting the elicitation and fulfillment of privacy requirements, *Requirements Engineering* 16 (2011) 3–32. URL: <https://doi.org/10.1007/s00766-010-0115-7>. doi:10.1007/s00766-010-0115-7.
- [30] D. Soares Cruzes, M. Gilje Jaatun, K. Bernsmed, I. A. Tondel, Challenges and experiences with applying microsoft threat modeling in agile development projects, *Proceedings - 25th Australasian Software Engineering Conference, ASWEC 2018* (2018) 111–120. doi:10.1109/ASWEC.2018.00023.
- [31] G. B. Herwanto, G. Quirchmayr, A. M. Tjoa, A named entity recognition based approach for privacy requirements engineering, in: *2021 IEEE 29th International Requirements Engineering Conference Workshops (REW)*, 2021, pp. 406–411. doi:10.1109/REW53955.2021.00072.