Privacy and Security documents for Agile Software Engineering: An experiment of LGPD Inventory adoption

Juliana Saraiva
Informatics Center
Federal University of Pernambuco
Recife, Brazil
julianajags@dcx.ufpb.br

Sérgio Soares
Informatics Center
Federal University of Pernambuco
Recife, Brazil
scbs@cin.ufpe.br

***Abstract*— Context: Society 5.0 depends on intelligent technologies that capture and monitor data in real-time. Software development needs to guarantee the privacy and protection of personal data, according to regulations such as the GDPR in Europe and the LGPD in Brazil. Objective: The main goal of this study is to check the possibility of Personal Data Inventory (PDI) adoption for User Stories and BDD Scenarios creation, to verify its relevance regarding software understanding and documentation. Method: An experiment was performed with 34 undergraduate students from higher education institutions in Brazil. They had to create User Stories and BDD Scenarios from two documents: (1) PDI and (2) a detailed system description. Results: After assessing 184 software engineering artifacts, findings indicated there were no statistically significant differences in the quantity and quality of Stories and Scenarios generated by the two groups. Conclusion: The PDI document can be effectively employed in the Requirements Engineering field as a valuable tool for comprehending and specifying software, thereby ensuring compliance with data protection and privacy regulations. The fusion of Information Security and Software Engineering has the potential to enhance the development of products and services that align better with privacy requirements.**

Keywords*— *Personal Data Inventory, Agile Requirements Specification, Privacy Laws, LGPD

I. Lay Abstract

Today's society relies heavily on intelligent technologies that capture and monitor real-time data. In the realm of software development, it is crucial to ensure the protection of personal data, in adherence to regulations like the GDPR in Europe and the LGPD in Brazil. These regulations necessitate privacy considerations throughout the entire software lifecycle, starting from its inception. Companies offering products and services are required to maintain a comprehensive Personal Data Inventory (PDI), which entails mapping the lifecycle of personal data. This study evaluates the impact of utilizing the PDI in the creation of User Stories and Behavior-Driven Development (BDD) Scenarios. The aim is to assess the PDI relevance in enhancing the understanding and documentation of software. The evaluation involved 34 undergraduate students from institutions of higher education. The results demonstrate no statistically significant difference in the quantity and quality of the artifacts generated by two groups: (1) those with access solely to the PDI, and (2) those with access to the detailed system description. Consequently, it can be concluded that the PDI holds the potential for adoption in Requirements Engineering. These findings bear significance, particularly since numerous Brazilian companies are either in the early stages or yet to commence the LGPD adoption process. The PDI can serve as a tool in facilitating the comprehension and specification of software, thereby ensuring compliance with data protection regulations. Furthermore, the integration of Information Security and Software Engineering can greatly benefit the development of software that align more effectively with privacy requirements.

II. Introduction

Given the prevalence of intelligent technologies that capture and monitor data in real-time, today's society necessitates software development to ensure the privacy and protection of personal data. It is important to acknowledge that the increasing use of IoT devices in people's lives and within organizations has significantly amplified the collection, sharing, and processing of personal data. Considering that this data is part of the subjects' identity and self-determination, regulations on Privacy and Protection of Personal Data have been proposed around the world, as is the GDPR (General Data Protection Regulation) case in Europe [1] and the LGPD (General Data Protection Law – *Lei Geral de Proteção de Dados*) in Brazil [2].

In response to this data-driven landscape, various regulations on Privacy and Protection of Personal Data have been proposed globally [3]. These regulatory frameworks emphasize that privacy considerations should be embedded in software products and solutions from their inception, integrating privacy by design and privacy by default principles throughout the software's lifecycle [4]. Consequently, these obligations directly impact the Requirements Engineering Process, requiring a thorough understanding and documentation of the software's features and limitations from a privacy perspective [5] [6].

Furthermore, these regulations also need the creation of a Personal Data Inventory (PDI), which serves as a document mapping the journey of personal data from its collection to its final processing [7]. The PDI provides insights into how personal data is collected and flows within an organization.

In Brazil, the ANPD (National Data Protection Authority – *Autoridade Nacional de Proteção de Dados*) is responsible for safeguarding personal data and ensuring compliance with the LGPD [1]. As part of its responsibilities, the ANPD has issued normative instructions over the years to address matters related to personal data protection and privacy. One such instruction is the Guide for constructing the PDI, which outlines the information that should be included in the inventory, such as data processing processes, processed data, data categories, legal bases for data use, and information security measures adopted to protect personal data [7].

The PDI construction is mandatory for Brazilian companies offering products and services for commercial

978-1-6654-5223-6/23/$31.00 ©2023 IEEE

purposes, including software companies. It is worth noting that this requirement is not exclusive to Brazil, as similar privacy laws worldwide, such as the GDPR, also incorporate the need for data inventories. However, since the PDI is typically developed by information security analysts rather than software engineers, it is crucial to investigate whether the information contained within it can aid in understanding and specifying the system during the Requirements Engineering process.

This study focuses on evaluating the PDI applicability in Agile Methodologies, which are widely adopted in software development processes [8]. Among the artifacts commonly generated during the software understanding phases, User Stories and BDD Scenarios are frequently used to document system requirements [9] [10]. Hence, the objective of this work is to assess whether the PDI can contribute to the creation of User Stories and BDD Scenarios.

In Brazil, despite the LGPD being published in 2018 and becoming effective in 2020, studies indicate that most companies have yet to commence or are in the early stages of adapting to the law [11]. As a result, many software development companies have not yet implemented the PDI or are still in the process of constructing it.

To address the challenge of finding fully compliant software companies with the LGPD, this study conducted an experiment involving undergraduate students enrolled in Software Engineering classes at a federal teaching institute in Brazil. The research aimed to analyze the impact of using the PDI in the process of creating User Stories and BDD Scenarios, evaluating whether this Information Security artifact can enhance the understanding and documentation of software.

Although the study did not take place within companies, it was conducted by an information security analyst who holds EXIN LGPD certification and possesses experience in LGPD adaptation processes in companies of various sizes. Furthermore, the PDI utilized in this research was constructed based on a simulated real-world case, leveraging the extensive knowledge of students and teachers involved in the Academic Management System.

This paper comprises seven sections, including this introduction and the Lay Abstract. Section III presents related works, followed by Section IV, which outlines the methodology employed in this study. Section V presents a discussion of the findings, while Section VI addresses potential threats to the study's validity. Finally, Section VII presents conclusions and possibilities for future work.

III. Related works

The intersection of Software Engineering and Information Security has gained significant attention in recent years due to the increasing legal requirements for technology companies to implement security measures to protect the privacy of personal data. One notable tool in this context is the PCM Tool: Privacy Requirements Specification in Agile Software Development, which was introduced by Peixoto et al. [12]. The authors highlighted the challenges faced by software developers in specifying privacy requirements and, in response, proposed this tool as a guide to facilitate the specification of privacy requirements in the context of agile software development.

Building on this research, Peixoto et al. examined the level of comprehension among software developers concerning privacy [13]. They analyzed personal, behavioral, and environmental factors that could impact the decision-making process when modeling privacy requirements. By conducting semi-structured interviews with 30 engineers, the study identified 9 personal, 5 behavioral, and 7 external factors that were deemed significant in defining the comprehension and specification of privacy requirements.

Melo Filho et al. conducted a literature review on Scrum elements that can aid organizations in implementing the LGPD [14]. The authors contend that adopting Scrum offers increased agility and adaptability to legal requirements. In addition to the LGPD, they examined the significance of other Brazilian laws, such as the Internet Civil Mark (*Marco Civil da Internet)*. Consequently, the authors put forth a Scrum Methodology as an effective and versatile approach for implementing and raising awareness about the LGPD within organizations.

In a similar vein, Cardoso et al. conducted a study with the objective of providing a practical guide for Brazilian companies to comply with the LGPD [15]. They introduced a methodology adopting the Data Protection Management System for process management, implementation, and governance of privacy and data protection. The study concluded that the Scrum framework exhibited robustness in terms of tools and techniques that streamline the management and implementation of each of the five phases of SGPD. Thus, the SGPD methodology serves as a valuable best practice guide to accomplish the LGPD implementation project goal.

Furthermore, in the same year, Camêlo et al. put forth the G-Priv: A Guide to Support the Specification of Privacy Requirements in Compliance with the LGPD [16]. The authors highlighted the challenges associated with extracting and operationalizing privacy requirements for personal data. To address this issue, they conducted a survey involving 18 professionals and subsequently proposed a catalog of privacy standards along with the G-Priv guide. These resources aim to facilitate the specification of privacy requirements in alignment with the LGPD.

It is evident that solutions, techniques, and tools employed in the software development process can contribute to ensuring the protection and privacy of personal data. However, there is still a need for studies that investigate whether the artifacts generated by Information Security analysts can be effectively utilized in the software development process. Hence, the objective of this research is to assess the feasibility of utilizing the Personal Data Inventory (PDI) as a tool to support the Requirements Engineering process, with a specific focus on creating User Stories and BDD Scenarios.

It is worth noting that while the mentioned studies primarily address the LGPD, which is the Brazilian regulation governing the privacy of personal data, the global trend of data protection follows a similar path. Moreover, the LGPD itself draws inspiration from the GDPR, which is enforced in Europe. Thus, the significance of these studies, including ours, underscores the importance of interdisciplinary communication and collaboration between the fields of Software Engineering and Information Security in safeguarding the privacy and protection of personal data, as mandated by laws and regulations worldwide.

IV. METHODOLOGY

To assess the feasibility of utilizing the Personal Data Inventory (PDI) in constructing User Stories and BDD Scenarios, the research followed the methodological steps depicted in Figure 1, which will be elaborated upon in the subsequent sections. The study involved undergraduate students from two federal higher education institutions located in the Northeast region of Brazil. A total of 34 students from various disciplines, including Computer Science, Data Science and Artificial Intelligence, and Technology in Systems Analysis and Development, participated in the survey.

It is important to note that despite the LGPD being in effect since 2020 in Brazil, most software companies are still in the early stages of adapting to the law [11]. Consequently, conducting studies with the PDI of these companies directly poses challenges. This motivated the decision to focus on aspiring software engineers, specifically undergraduate students enrolled in Software Engineering courses, as they are in the process of acquiring relevant training.

A. Research Questions and Indicators

To investigate the Personal Data Inventory (PDI) potential in facilitating the creation of Software Engineering artifacts, the following Research Questions (RQ) were formulated:

- **RQ01:** Can User Stories be effectively constructed using the LGPD's PDI?
- **RQ02**: Can BDD Scenarios be effectively constructed using the LGPD's PDI?
- **RQ03:** Is there a difference between the number of Software Engineering artifacts created using the LGPD's PDI compared to those created using a system description in natural language?
- **RQ04:** Is there a difference between the quality of Software Engineering artifacts created using the LGPD's PDI compared to those created using a natural language system description?

Table 1 presents a summary of the metrics and indicators employed in this study. The first column indicates the collected metric, while the second column outlines the purpose of collecting each metric. The third column provides the guiding question for generating potential reference values for the metrics. The selection of metrics was determined using the GQM approach [17], which served as the basis for defining the metrics.

- POUSC – Possibility of User Stories Creation
- POBSC – Possibility of BDD Scenarios Creation
- #US - Number of User Stories
- #BS – Number of BDD Scenarios
- USCt – User Stories Correctness
- BSCt – BDD Scenarios Quality Correctness

For the POUEC and POBSC metrics, we verified if all students produced at least 1 User Story (US) and 1 BDD Scenario (BS) for each suggested functionality. Considering that User Stories intend to capture the user's requirements, they should encompass the primary service that the functionality intends to deliver. The US quality was assessed by evaluating the main task to be executed accuracy by functionality.

TABLE I. STUDY METRICS AND INDICATORS – GQM MODEL

Metric	GQM Approach		
	Goal	*Question*	*Value*
POUSC	Check if it is possible to create User Stories adopting PDI.	Is it possible to create a US?	YES/NO
POBSC	Check if it is possible to create BDD Scenarios adopting PDI.	Is it possible to create a BS?	YES/NO
#US	Check the number of US created.	How many US were created?	Number
#BS	Check the number of BS created.	How many BS were created?	Number
USCt	Assess US Correctness.	What is the % of incorrected US created?	% errors
BSCt	Assess BS Correctness.	What is the % of incorrected BS created?	% errors

The Correctness was determined based on the error percentage in the development of US and BS. These artifacts were deemed incorrect when they deviated from the structural clauses provided at the study outset or when they were not written in accordance with the specifications outlined in the System Description or the process details outlined in the Inventory. In such cases, US and BS were deemed invalid as Software Engineering artifacts for statistical analysis. The evaluation of these artifacts Correctness was conducted by researchers with extensive experience in teaching Software Engineering disciplines over several years.

Through the metrics shown in Table 1, it was possible to create indicators that help answer the research questions. The first indicator corresponds to the Possibility of creating US and BS through the PDI. The POUSC and POBSC metrics compose this indicator. The Number of US and BS is another indicator (No_{Art}) composed of the metrics #US and #BS. Finally, the Quality of Artifacts (Ql_{Art}) was represented by the USCt and BSCt metrics, which correspond to the constructed items correctness.

B. Study Design

The study involved the creation of User Stories and BDD Scenarios by undergraduate students majoring in Software Engineering, specifically those in Computer Science, Data Science and Artificial Intelligence, and Technology in Systems Analysis and Development programs. The development of these Software Engineering artifacts was facilitated through the students' access to two pre-existing documents: (a) a detailed description of the system's functionalities and (b) the Personal Data Inventory (PDI), which contained the information required by the LGPD. The students were randomly assigned to two groups: GROUP 01, which had access to the System Description, and GROUP 02, which had access to the Personal Data Inventory. This division was made to determine the content influence accessed by the students on their subsequent activities. The experimental design is illustrated in Figure 1.

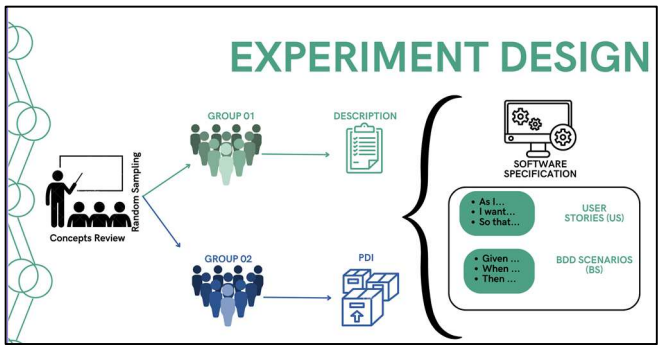

Fig. 1. Experiment Design illustration

The random grouping of students aimed to examine whether there would be any difference in the quantity and quality of User Stories (US) and BDD Scenarios (BS) created when using only the Personal Data Inventory (GROUP 02) compared to GROUP 01. It is important to note that the PDI utilized in this study was developed by an information security analyst who specializes in LGPD compliance. The analyst has extensive experience in constructing PDIs for various companies of different sizes and industries in Brazil. On the other hand, the System Description accessed by the students followed a widely used model employed by software companies. It provided detailed documentation of the system's functionalities, including the identification of primary tasks to be performed, data processing activities, preconditions for execution, and functionality restrictions. All information presented in the System Description was expressed in clear and simple Portuguese, using natural language.

The experiment was conducted as a practical activity within the Software Engineering courses offered by these institutions. The activities took place in the computing laboratories of the respective universities, with the authors and course instructors overseeing the process. The execution of these activities was divided into five sessions, with each session lasting approximately 90 minutes in each educational institution where the study was conducted:

1. Presentation of fundamental concepts related to personal data, information security, and LGPD - delivered by one of the authors (15 minutes).
2. Review of the content on creating User Stories and BDD Scenarios - presented by one of the authors of this study (20 minutes).
3. Random assignment of students to groups (10 minutes) and explanation of the experiment.
4. Distribution of shared folders on Google Drive containing (i) Task guidelines, (ii) User Story and BDD Scenario templates, (iii) System Description, and (iv) Personal Data Inventory (5 minutes).
5. Creation of User Stories and BDD Scenarios, followed by submission of the generated artifacts via email (40 minutes).

It is important to emphasize that the students had already studied the content of requirements engineering in these courses. During the presentation of the theoretical concepts prior to the task execution, there was active interaction between the students and the authors of this study to address any doubts or comments. Furthermore, while creating the User Stories and BDD Scenarios, the students had the opportunity to clarify any procedural uncertainties regarding the task execution. However, at this stage, generated artifacts correctness and completeness confirmation was not considered to avoid influencing the research results.

A fictional scenario of an information system deployed for supporting a university activity - Academic Management Software (AMS) was adopted, focusing on the functionalities of "*User Registration*" and "*Issuance of School Transcripts*". This scenario was selected because undergraduate students possess knowledge and experience using such systems. Despite the potential variation in the specific AMS used by the students, there is a familiarity with these functionalities, which facilitates comprehension of a plausible real-world application scenario.

The "*User Registration*" functionality is a service within the software that is designed to collect and store user information, serving as a prerequisite for accessing the system in the future. An electronic form is used to gather the following required information:

1. Identification of the Educational Institution
2. Enrolled/Finished Course
3. Name of the Student
4. Student's last name
5. Social Name
6. Age
7. Date of birth
8. Place of birth
9. Marital Status
10. Genre
11. Membership
12. Address
13. Phone(s)
14. Email
15. GR (General Registration)
16. NDL (National Driver's License)
17. Password

The "Issuance of School Transcripts" functionality is characterized by generating a .pdf document that contains school-related information. Access to this functionality is restricted to logged-in users. Therefore, it is observed that this functionality has access restrictions and specific preconditions for its execution: (i) the existence of an account, (ii) successful login to the system, and (iii) completion of curricular components. The required data to be included in the School Transcript are:

1. Identification of the Educational Institution;
2. Enrolled/Finished Course;
3. Name of the Student;
4. Student's last name;

5. Social Name;
6. Registration Number;
7. Membership;
8. Entry Year;
9. Estimated Completion Date;
10. Code of Curricular Components Attended;
11. Name of Curricular Components Attended;
12. Workload of each Curriculum Component;
13. Notes of each unit on components;
14. Component Status (Approved/Failed);
15. Pending Curricular Components;
16. Grade Average in the course;
17. Space for observations;
18. Place of Issue;
19. Date of Issue.

C. Data Extraction and Analysis

After the artifacts were sent via email, the data were extracted and stored in Google Sheets spreadsheets. To collect the metrics used in this study (as presented in Table 1), the following information was stored:

- US description;
- BS description;
- Total number of US created;
- The number of US created with errors;
- Total number of BS created;
- The amount of BS created with errors.

To evaluate the quantity and quality of the Software Engineering artifacts generated by the two groups, a statistical analysis was conducted using the RStudio software[1]. Initially, the collected data normality was assessed using the Shapiro-Wilk test [18]. In this type of analysis, if the p-value is less than 0.05, it indicates data non-normality. This was the case for all data groups in this study. As a result of the data non-normal distribution, the Wilcoxon test was selected as the statistical test to evaluate the hypotheses presented below. This test is employed to assess the statistical difference between two groups with a non-parametric distribution [19] [20]. Such test was used in this study to examine whether there was a significant difference in the quantity and quality of artifacts generated by GROUPs 01 and 02.

To assess the quantity, the medians of the metrics #US and #SB for GROUPS 01 and 02 were compared. For quality analysis, the correctness was evaluated by comparing the percentage of errors found in the generated artifacts (% errors) using the metrics USCt and BSCt between the two groups. Consequently, the following hypotheses were employed for these two tests:

$$H_{01} : No_{Art}1 = No_{Art}2$$

$$H_{11} : No_{Art}1 > No_{Art}2$$

$$H_{02} : Ql_{Art}1 = Ql_{Art}2$$

$$H_{12} : Ql_{Art}1 > Ql_{Art}2$$

$No_{Art}1$: Number of Artifacts generated by Group 01

$No_{Art}2$: Number of Artifacts generated by Group 02

$Ql_{Art}1$: Quality of Artifacts generated by Group 01 (Correctness - %Errors)

$Qt_{Art}1$: Quality of Artifacts generated by Group 02 (Correctness - %Errors)

V. Results Discussion

After the creation of US and BS assessments, it can be concluded that it is indeed possible to build these artifacts using only the PDI. All students, regardless of whether they had access to the PDI only (GROUP 02) or both the PDI and system description (GROUP 01), successfully constructed at least one US and one BS for each system functionality. This positive response confirms the POUSC and POBSC metrics effectiveness. It is worth noting that the absence of US and BS in some cases may be attributed to the participants familiarity with the functionalities addressed in the study. However, it is important to emphasize that the primary objective of this study was not to assess the complexities of specifying system functionality, but rather to investigate the feasibility of creating Software Engineering artifacts using only the PDI.

In total, 184 software engineering artifacts were created during the study, comprising 64 US and 120 BS. These artifacts were generated by the 34 students who participated in the research and were studying Software Engineering. The distribution of these artifacts is illustrated in Graph 2.

The chart clearly shows that there are more BS artifacts than US artifacts. This observation aligns with the expectations set during the content explanation prior to the experiment execution. It was mentioned that typically, at least 1 US should be created for each functionality, while at least 2 BS should be created for each story. These two scenarios are usually used to describe the main execution behavior of the functionality and at least one alternative flow. However, it is important to note that the students had the freedom to create as many US and BS as they deemed necessary within the given time frame. It was made clear to the students that they could construct any artifact if the provided information was insufficient to complete the task.

[1] RStudio. RStudio: Integrated Development Environment for R. Available at https://www.rstudio.com/ . Accessed on: May 10, 2023.

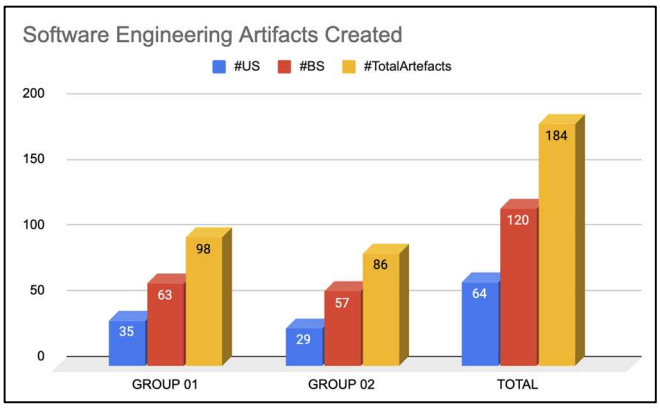

Fig. 2. Total of Software Engineering Artefacts created

Since the data does not exhibit a normal distribution, the Wilcoxon Test, which is suitable for non-parametric evaluation, was employed. When formulating the null hypothesis (H_0) for the Wilcoxon test, it is assumed that there is no difference between the two paired samples. The alternative hypothesis (H_1), on the other hand, suggests the presence of a systematic difference between the samples.

The p-value represents the probability of obtaining the observed test statistic under the null hypothesis. Thus, if the p-value is lower than the chosen significance level (0.05), one can reject the null hypothesis and conclude that there is sufficient evidence to support the assertion that the differences between the samples exhibit a systematic tendency.

By utilizing this statistical analysis, a p-value associated with each test was calculated, and the results are presented in Table II (third column). The first column indicates the specific null hypothesis being tested, while the second column refers to the group of metrics under examination. It is evident that for all cases, the p-value exceeded 0.05, rendering it impossible to reject the null hypotheses formulated in this research. Hence, there was no statistically significant difference observed in terms of the quantity and quality of Software Engineering artifacts generated by GROUP 01, which had access to a detailed system description, and GROUP 02, which solely relied on the LGPD PDI.

Despite the disparity in artifact creation quantities (Figure 3), where GROUP 01 exhibited the highest output, the Wilcoxon Test results indicate that there is no significant distinction between the groups. It is worth emphasizing that GROUP 02 had their initial exposure to a PDI, necessitating a learning curve until they adapt to the information within the document, including details that may be irrelevant for the creation of Software Engineering artifacts.

It was possible to calculate a p-value for each analysis, and the results are presented in Table II (third column). The first column indicates the null hypothesis being tested, while the second column refers to the group of metrics under examination. It can be observed that, in all cases, the calculated p-values were greater than 0.05, **making the null hypotheses rejection impossible.** Therefore, there was no statistically significant difference found in terms of the quantity and quality of Software Engineering artifacts generated by GROUP 01, which had access to a detailed system description, and GROUP 02, which solely relied on the LGPD PDI.

Despite the discrepancy in the number of artifacts created (Figure 3), with GROUP 01 having the highest quantity, the Wilcoxon Test results demonstrate the absence of a significant difference between the groups. It is important to emphasize that GROUP 02 had its initial exposure to a PDI. Consequently, there is a learning curve involved until they adapt to the information contained in the document, even including aspects that are unnecessary for the creation of Software Engineering artifacts.

TABLE II. WILCOXON TESTS RESULTS

H_0	**Statistics Evaluation (Groups 01 and 02)**	
	Wilcoxon Test for	***P- value***
H_{01}	#US	0.3456
H_{01}	#BS	0.7028
H_{02}	USCt	0.4076
H_{02}	BSCt	0.7247

On the contrary, it is noteworthy that, for the second functionality, GROUP 02, which had access solely to the PDI, was able to construct a greater number of US and BS. Thus, it can be inferred that the utilization of this Information Security artifact is effective in the Requirements Engineering process.

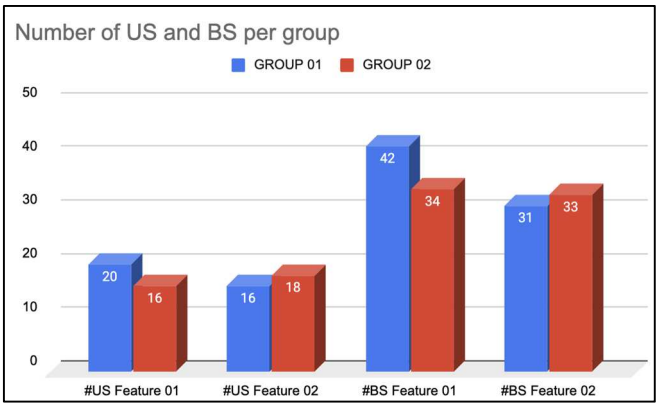

Fig. 3. Number of US and BS created per group

Although nearly 200 Software Engineering artifacts were created, 3 US and 23 BS were excluded from the statistical analysis. These artifacts deviated from the study predefined structure or did not contain the main task specification that the functionality should perform in the case of US, or the expected behaviors (main and alternative) in the activities execution. As shown in Figure 4, it can be observed that most errors occurred in the construction of BS. This outcome is expected as BS is more intricate to comprehend, model, and write, often dealing with non-functional requirements that possess this characteristic.

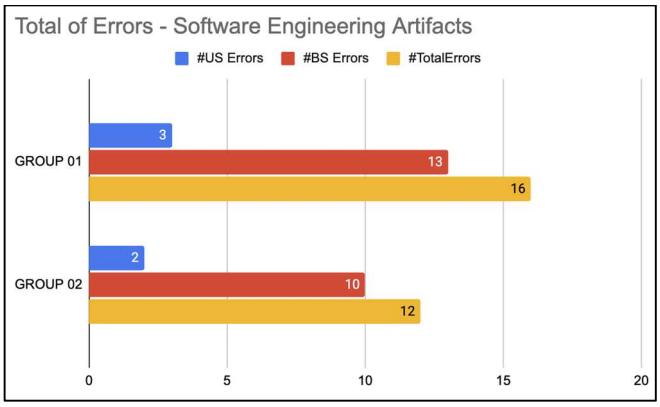

Fig. 4. Errors found in Software Engineering Artefacts

Furthermore, the inclusion of information security requirements in the construction of BS was necessary for both GROUP 01 and GROUP 02. This requirement may have contributed to students making incorrect or incomplete specifications. Interestingly, GROUP 02, which solely relied on the PDI, made fewer errors in terms of structure and specification compared to the students who had access to the system description. This finding reaffirms the effectiveness of using the PDI for the creation of US and BS. However, it is important to note that statistically, no significant difference was observed in terms of quantity and quality of the Software Engineering artifacts created, as indicated by the Wilcoxon tests.

VI. Threats to Validity

Inherent to any experiment, this work faces several threats to validity. Firstly, in terms of **construct** validity, it is acknowledged that prior knowledge of the functionality scenarios may have facilitated the construction of US and BS in both groups. However, the experiment goal was not to evaluate the features complexity or the difficulties students might encounter when creating US and BS. Moreover, the presence of structural and material errors made by students in both groups suggests that familiarity with the functionalities does not always guarantee easy and accurate specifications.

Regarding **internal** validity, the selection of undergraduate students may have influenced the quantity and quality of the generated artifacts. To mitigate this bias, students from the specific discipline of Software Engineering were targeted, as they had already studied Requirements Engineering and possessed a basic understanding of the main concepts applicable in practice. Furthermore, an explanation was provided on the LGPD, US, and BS key concepts to ensure a homogenous level of theoretical knowledge among the participants.

Regarding **external** validity, it is recognized that non-parametric tests, such as the Wilcoxon test adopted in this study, possess less strength and statistical robustness. However, the analyzed data dispersion led to the choice of this test. Additionally, this initial study was conducted in two different institutions, located in different states, with distinct courses, to enhance the potential representativeness within the experimental context. Furthermore, replications are being carried out to explore alternative statistical methods and data analysis techniques.

Finally, concerning the threat to **conclusion** validity, the adopted statistical test does not impose a minimum sample size requirement. Therefore, despite this limitation, it has been statistically demonstrated that there is no difference between the quantity and quality of artifacts generated by the two groups. Additionally, the qualitative analysis conducted through the review of all US and BS created serves as complementary evidence that supports the research conclusions themselves.

VII. Concluding Remarks

Considering the growing demand for collection, sharing, and processing of personal data in increasingly intelligent technological solutions, it becomes imperative to protect data subjects. To achieve this, regulations such as the LGPD are being adopted worldwide, emphasizing the need to consider privacy throughout the software lifecycle. Specifically, the law mandates constructing and maintaining a Personal Data Inventory (PDI) by every software company. As the PDI is typically developed by information security analysts rather than software engineers, it is necessary to investigate whether the information contained in it can aid in the system understanding and specification during the Requirements Engineering process.

While studies have been conducted to enhance the understanding, documentation, and development of software quality requirements, particularly in relation to data privacy, there remains a need for research that demonstrates the potential use of artifacts generated by information security professionals in software engineering. Therefore, this study evaluated 184 artifacts created by 34 undergraduate students in Brazil, utilizing either the PDI or the detailed system description as reference material for their construction.

Considering that PDI creation is mandatory for all Brazilian software companies, it can be employed in the agile requirements specification process, specifically in the creation of User Stories (US) and Behavior Driven Development Scenarios (BS). It is important to note that this requirement is not exclusive to Brazil and is also present in other privacy laws worldwide, such as the GDPR's “Registration of Treatment Activities” document.

Throughout the qualitative and statistical analyses conducted in this study, six metrics were observed to address the research questions raised. The results demonstrate that it is feasible to develop US and BS using the PDI (RQ01 and RQ02). Furthermore, both the quantity (RQ03) and quality (RQ04) of these artifacts are comparable to those generated using more traditional documents in the requirements specification process, such as system descriptions in natural language.

Therefore, answering RQ01 and RQ02, it was observed that US and BS could be successfully constructed using both the PDI and the detailed system descriptions (POUSC and POBSC metrics). Additionally, the number of US and BS evaluation created by the two groups using the Wilcoxon Test revealed no significant difference between them (metrics #US and #BS) – assessment of the RQ03.

The same trend was observed in the assessment of artifact quality – discussion about the RQ04. A metric was established to measure the percentage of structural or material errors found in US and BS created by the students. The evaluation of both groups indicated no difference in the number of errors committed by the students. These results provide evidence that the PDI not only aids but can also be utilized

independently in the process of developing software engineering artifacts.

As practical implications for software engineering, it is essential to consider artifacts produced by professionals from other domains within software companies, including information security. These documents may contain valuable information for conducting activities inherent to the software development process. In the case of the PDI, it contains crucial details about the protection and privacy of personal data, which can assist software engineers in understanding, eliciting, and documenting non-functional requirements, such as security requirements.

In Brazil, ANPD specifies that the PDI should include the following information: (1) the process linked to the processing of personal data, (2) the processing of personal data, (3) the actors involved (processing agents), (4) the treatment purpose, (5) the legal basis supporting the treatment (as per articles 7 and 11 of the LGPD), (6) the personal data processed, (7) the category of personal data subjects, (8) the retention period for personal data, (9) the institutions with which personal data is shared, (10) the methods employed for international data transfers, and (11) the information security measures adopted.

Numerous studies have highlighted the inherent complexity of data privacy requirements, making them challenging to understand, elicit, document, and validate with users. However, PDI use can help clarify service details and the corresponding service restrictions necessary to safeguard the privacy of personal data. Furthermore, once the elements between the PDI and the templates employed for constructing software engineering artifacts are mapped, strategies can be devised to avoid redundant information gathering processes for meeting the PDI requirements by security analysts and US and BS creation by software engineers.

In many scenarios, direct user involvement during the LGPD implementation process (conducted by information security analysts) or requirements gathering (conducted by software engineers) is often limited due to logistical constraints such as time, travel, and scheduling. In such cases, the PDI adoption developed by the information security team can serve as an alternative when direct user participation is unfeasible. Alternatively, collaborative meetings involving both software engineering and information security professionals can be organized to understand and document the necessary artifacts for each area of activity within the company.

Finally, it is evident that students lack knowledge about the importance of guaranteeing data security and privacy from the design phase (*privacy by design*) and throughout the software lifecycle (*privacy by default*). These principles are explicitly advocated by the LGPD, and software engineers aiming to offer products and services in Brazil must adhere to them.

Given this new reality, undergraduate courses need to incorporate disciplines that address these issues more prominently into their curricula. Additionally, the new technical skills necessary for software engineering activities must be taught, including the development of APIs focused on software security, architectural design solutions for personal data security, specification of security requirements, adoption of data security frameworks, and security testing techniques such as vulnerability and penetration tests, among others.

While this study contributes significantly to software engineering, it is important to acknowledge its limitations, which are inherent to research of this nature and call for future work. Therefore, additional data collection is underway in other Brazilian universities to increase sample size and consider other variables such as students' knowledge levels, undergraduate courses, and training curricula.

A similar study is being conducted to observe the construction of the same artifacts by engineers with different levels of experience using the PDI, focusing on professionals in the requirements specification field. Additionally, a survey is being administered to gather the opinions of these software engineering professionals.

Likewise, it is crucial to explore whether software engineering artifacts can be adopted by information security analysts. This investigation is particularly relevant in Brazil, where despite the LGPD being enacted in 2018, a significant number of software companies are yet to comply with it. Consequently, the PDI is either in the early stages of development or has yet to be created, despite being legally required. Therefore, if it is determined that software engineering can assist information security, these institutions can utilize US and BS as the starting point for constructing their companies' PDIs.

This study did not specifically address the quantity and quality of security requirements in the creation of US and BS artifacts. Thus, further studies, already started, will assess the PDI's role in improving the specification of non-functional security requirements are necessary.

Although the study focused on a Brazilian legislation (LGPD) the production and availability of software as a product and service have a global impact. Discussions on the international sharing and transfer of personal data by software companies are ongoing worldwide. The LGPD, along with the GDPR and other regulations concerning personal data privacy, has been implemented in several countries and will continue to impose increasingly strict requirements on all types of companies, including software companies. In Brazil, personal data protection has become a fundamental right since 2020, leading to more assertive inspections and collections by regulatory authorities concerning personal data security.

ACKNOWLEDGMENT

This work is partially supported by INES 2.0 (www.ines.org.br), CNPq grant 465614/2014-0, FACEPE grants APQ-0399-1.03/17 and APQ/0388-1.03/14, and CAPES grant 88887.136410/2017-00. Sérgio Soares is partially supported by CNPq grant 306000/2022-9.

REFERENCES

[1] Lei Geral de Proteção de Dados (LGPD), Lei nº 13.709, de 14 de agosto de 2018.

[2] General Data Protection Regulation (GDPR), Regulation (EU) 2016/679, 27 April 2016.

[3] McGruer J. Emerging Privacy Legislation in the International Landscape: Strategy and Analysis for Compliance. Wash J Law Tech Arts. 2020;15:120. Available from: https://digitalcommons.law.uw.edu/wjlta/vol15/iss2/3

[4] Alkubaisy, D., Piras, L., Al-Obeidallah, M.G., Cox, K. And Mouratidis, H. 2022. A framework for privacy and security requirements analysis and conflict resolution for supporting GDPR compliance through privacy-by-design. In Ali, R., Kaindl, H. and Maciaszek, L.A. (eds.). Evaluation of novel approaches to software

engineering: revised selected papers from 16th International conference on Evaluation of novel approaches to software engineering 2021 (ENASE 2021), 26-27 April 2021, [virtual conference]. Communications in computer and information science, 1556. Cham: Springer [online], pages 67-87. Available from: https://doi.org/10.1007/978-3-030-96648-5_4

[5] Senarath, A.R., Arachchilage, N.A.G., 2018b. Understanding user privacy expecta- tions: A software developer's perspective. Telemat. Inform. 35, 1845–1862. http://dx.doi.org/10.1016/j.tele.2018.05.012.

[6] Senarath, A., Grobler, M., Arachchilage, N.A.G., 2019. Will they use it or not? Investigating software developers' intention to follow privacy engineering methodologies. ACM Trans. Priv. Secur. 22, 1–30. http://dx.doi.org/10.1145/ 3364224.

[7] ANPD. Guia de Elaboração de Inventário de Dados Pessoais. Brasília, DF: ANPD, [2023]. Disponível em: https://www.gov.br/governodigital/pt-br/seguranca-e-protecao-de-dados/ppsi/guia_inventario_dados_pessoais.pdf. Acesso em: May 01, 2023.

[8] A Mihelič A, Vrhovec S, Hovelja T. Agile development of secure software for small and medium-sized enterprises. Sustainability. 2023;15(1):801.

[9] Georges T, et al. Guiding feature models synthesis from user-stories: an exploratory approach. Synthesis. 2023;30:31.

[10] Parsa S. Acceptance testing and behavior driven development (BDD). In: Software Testing Automation: Testability Evaluation, Refactoring, Test Data Generation and Fault Localization. Cham: Springer International Publishing; 2023. p. 79-158.

[11] Morais C. Desmistificando a LGPD: entenda como a Lei Geral de Proteção de Dados Pessoais pode ser aplicada no dia a dia das empresas e das pessoas. Editora Dialética; 2023.

[12] Peixoto M, et al. The perspective of Brazilian software developers on data privacy. Journal of Systems and Software. 2023;195:111523.

[13] Peixoto, Mariana, et al. "The perspective of Brazilian software developers on data privacy." Journal of Systems and Software 195 (2023): 111523.

[14] de Melo Filho DR, et al. Metodologia Scrum: Uma aliada na implementação da LGPD. Research, Society and Development. 2023;12(4):e22712441189-e22712441189.

[15] Cardoso DL, Cardoso T. Adequação da LGPD via "Projetos Ágeis Scrum". Boletim do Gerenciamento. 2023;35(35):28-41.

[16] Camêlo MN, Alves CF. G-Priv: Um Guia para Apoiar a Especificação de Requisitos de Privacidade em Conformidade com a LGPD. iSys-Brazilian Journal of Information Systems. 2023;16(1):2-1.

[17] Basili V, Caldiera G, McGarry F, Rombach HD. GQM^+ strategies--aligning business strategies with software measurement. In: First International Symposium on Empirical Software Engineering and Measurement (ESEM 2007). IEEE; 2007.

[18] Royston P. Approximating the Shapiro-Wilk W-test for non-normality. Statistics and computing. 1992;2:117-119.

[19] Cuzick J. A Wilcoxon-type test for trend. Statistics in medicine. 1985;4(1):87-90.

[20] Divine G, Norton HJ, Hunt SL, Dienemann JA. A review of analysis and sample size calculation considerations for Wilcoxon tests. Anesthesia & Analgesia. 2013;117(3):699-710.