Development of a Measurement Instrument for Process Debt Detection in Agile Software Development Organizations

Tomas Gustavsson
Karlstad Business School, Karlstad University
Karlstad, Sweden *tomas.gustavsson@kau.se*

Muhammad Ovais Ahmad
Karlstad University
Karlstad, Sweden *ovais.ahmad@kau.se*

Hina Saeeda
University of Gothenburg
Gothenburg, Sweden *hina.saeeda@gu.se*

Abstract

This paper explores the concept of Process Debt (PD) in Agile Software Development (ASD) organizations. Drawing on the analogy with Technical Debt, PD is defined as the challenges that emerge from suboptimal or outdated processes, which can significantly hinder an organization's adaptability and software delivery effectiveness. The study proposes a survey instrument, designed to measure various types of PD based on existing research and expert interviews. Five types of PD are identified and operationalized: Process Unsuitability Debt, Synchronization Debt, Roles Debt, (Process) Documentation Debt, and Infrastructure Debt. The instrument's reliability and validity are assessed through a multi-stage process, culminating in a field survey within two ASD organizations. The findings significantly contribute to our understanding of PD and provide the first version of a validated tool for researchers and practitioners to identify and measure PD in their organizations.

Keywords: Process Debt, Technical Debt, Agile Software Development, Survey Instrument.

1. Introduction

In recent years, substantial research has been done on the Technical Debt (TD) phenomenon, which characterizes sub-optimal technical implementations that give a short-term benefit but create a context where a long-term interest is paid [14], [16]. However, there are other challenges where the debt metaphor can help to reason about aspects other than the technical ones. One such area of growing interest in the research community is the process debt (PD) phenomenon [16], [18, 19, 20]. PD refers to the compounding inefficiencies and challenges arising from an organization's suboptimal or outdated processes [2]. As these inefficiencies accumulate, they can significantly hinder an organization's ability to adapt to changing environments and to deliver software effectively [16], [19].

Organizations must continuously design and tailor new and efficient processes to guide their software development teams [21]. In software development, many organizations are using agile methodologies [12]. When implementing new agile methods or frameworks, old processes might continue alongside the new ones. This issue is especially problematic in large organizations with complex software development processes. For example, decision-making in traditional setups, which often involves many layers of approval, can clash with Agile's preference for quick decisions made by the team [12]. Hence, Agile Software Development (ASD) organizations are at risk of accumulating PD if obsolete processes still exist in the organization.

Despite the growing academic discourse surrounding PD, there remains a notable gap concerning the quantification and measurement of PD. To date, studies exploring PD have predominantly been qualitative in nature [1], [18, 19, 20]. These studies have provided valuable insights into the characteristics and implications of PD, yet they fall short of offering a systematic, quantifiable approach to its assessment and measurement. The absence of quantitative measures limits the depth of understanding of PD and constrains organizations' ability to address and mitigate its impacts strategically.

This paper aims to bridge the gap by investigating the following research question: *Can a survey instrument be operationalized to measure PD in an ASD context?* Our suggested survey instrument is designed to measure the different facets of PD, providing a tool for practitioners and researchers in the field. Through this work, we seek to contribute to the evolving conversation on PD and to lay the groundwork for more empirical, data-driven approaches to understanding and managing this complex phenomenon.

2. Process debt in agile software development

Software development intertwines social and technical factors, both of which are crucial to project success [5]. PD occurs within this socio-technical setting, encapsulating the inefficiencies stemming from less-than-ideal processes in organizational operations. PD acquires its shape from sub-optimal process designs, deviations from ideal procedures, or infrastructure deficiencies that, while possibly beneficial in the short term, can generate long-term negative impacts for the stakeholders involved [18].

Understanding, identifying, and mitigating the consequences of PD is especially crucial in ASD organizations. In agile contexts, where the ability to respond to change rapidly is a key competitive advantage, unaddressed PD can significantly hinder an organization's agility and responsiveness [20].

At the heart of agile methodologies lie principles of continuous improvement and empowerment [12]. Agile teams thrive on collaboration, autonomy, and empowerment to make decisions that best suit their projects [7], [12]. PD can erode these foundations by creating unnecessary barriers, reducing transparency, and limiting the team's ability to self-organize [2], [18]. This can lead to decreased morale and engagement, as team members may feel constrained by outdated or inefficient processes that do not align with their ASD environment [3]. Evidently, proactive PD management is indispensable for fostering and sustaining an environment where continuous improvement and high-performance team culture can flourish [19].

Amidst this context, Martini et al. [18, 19] have delineated a framework classifying six types of PD, identified through empirical studies. One such type is “Process Unsuitability Debt”, which occurs when a process is not aligned with the needs of an organization. For instance, software teams in a company might be Agile but must comply with a waterfall-like process inherited from other disciplines, leading to inefficiencies and confusion. The key issue is the existence of a process that creates overhead and delays due to its unsuitability [18]. Another type is “Synchronization Debt”, which arises when multiple intertwined processes lack effective synchronization points, leading to confusion, and disrupted workflow. The lack of synchronization impacts productivity and disrupts individual stakeholders' workflows, particularly developers [19]. A third type is named “Mismatching Roles and Responsibilities” (or *Roles Debt* [19]), which occurs when there is a discrepancy between the responsibilities outlined in a process and those in the organizational structure. It can lead to confusion and inefficiency, as seen in cases where roles like Product Owner are not clearly defined or aligned with organizational expectations [18]. The fourth described debt is called “(Process) Documentation Debt” (or *Documentation Debt* [19]), which is related to the inadequacy or inaccessibility of process documentation. It can manifest as either a lack of necessary information, leading to confusion, or overly detailed documentation, which can be overwhelming and lead to important details being overlooked [18]. A fifth type is the “Infrastructure Debt”, which involves issues with the tools and physical environment used in the process. Problems arise when tools are not well integrated, outdated, or unfit for modern processes, leading to overhead and errors. Physical workspace arrangements that do not meet the needs of the team members can also contribute to this type of PD [18]. Finally, the “Activity-specific Debt" type is related to sub-optimal activities within a process. It can be specific to certain tasks, such as prioritization or certification, and is context dependent [18]. These delineated PD types pave the way for developing this study's survey instrument scales.

3. Method

In this study, we employ a methodological framework that is both systematic and grounded in existing scholarly work. Our approach was informed by the instrument development and validation framework proposed by Recker and Rosemann [22], which is esteemed for its thoroughness and applicability across various research contexts. Furthermore, to ensure the instrument's reliability and validity, were also influenced by the rigorous guidelines set forth by MacKenzie et al. [17].

Initially, we investigated the theoretical underpinnings by performing a systematic mapping review [1] including PD and other non-technical debts. To complement our review, we conducted interviews with experts in the field, which facilitated the generation of a comprehensive list of potential items capable of measuring the constructs of interest. With a preliminary item pool, we sought domain experts' expertise to assess each item's relevance and accuracy in measuring the intended constructs. This involved a systematic evaluation process, where experts were asked to rank the items based on their significance and applicability. After this expert evaluation, we engaged in a collaborative refinement process with scholars and practitioners to enhance the precision and clarity of the items. We formulated an initial version of the survey instrument, then subjected to a pre-testing phase involving experts. The objective of this phase was to solicit feedback on the survey's overall clarity and to make necessary adjustments before wider distribution. A pilot study was conducted to gather data, which was analyzed using statistical techniques to assess the reliability and validity of our instrument. Finally, we executed a field survey to measure PD empirically.

3.1. Stage One: Construct and Item Creation

The first step was to investigate reported PD types to be operationalized into constructs. To substantiate the validity of this view, we carried out a literature mapping review on PD types [1]. Our mapping showed that all reported process characteristics can be mapped to the six PD types presented above: Process Unsuitability Debt, Synchronization Debt, Roles Debt, (Process) Documentation Debt, Infrastructure Debt, and Activity-Specific Debt. The last presented debt type, Activity-Specific Debt, is related to sub-optimal activities within a process specific to certain tasks, such as prioritization or certification, and is context-dependent. An example includes a cumbersome bug-fixing documentation process in a company, which could be termed as 'bug-fixing documentation debt' [19]. The nature of this debt type, which is context specific, makes it very difficult to operationalize as a generic measurement construct. Therefore, we decided not to include this PD type in our construct development.

For item creation, indicators used to measure a construct need to show content validity, defined as the degree of correspondence between the items selected to constitute a summated scale and its conceptual definition [11]. Thus, a conceptual definition of each construct of interest is required as a list of candidate items that represent the dimensions of the construct [22]. We conducted a literature review to identify suitable candidate items based on the conceptual definitions of the five selected PD types. We searched for peer-reviewed papers containing survey scales within software development, organizational management, and project management theory for sources that operationalized constructs that could be used for the constructs we intended to measure.

Most of the existing items were developed for other contexts, not in software development organizations, and had to be adapted. For example, Doleen et al. [6] operationalized team processes in a study of a production team, and Thomas et al. [24] operationalized task coordination and synchronization in a healthcare management context. We adapted the original items for each construct to fit our conceptual definition and unit of analysis (i.e., PD). As an example, we changed the original item "For any given situation, the sequence of actions required to achieve desired outcomes is clear to our team" [24] by replacing the words "sequence of actions" with "work processes" to better fit with our investigated PD constructs.

3.2. Stage Two: Substrata Identification

The second stage is called substrata identification, in which experts are used to identify and improve items [22]. Therefore, we conducted interviews with practitioners from three large ASD organizations, in which we asked them about different kinds of non-technical debts [1], such as process debt, social debt, and people debt. We did this to cover a wide area of debts, not restricting us to PD because some reported debt types could overlap the PD phenomenon. The transcribed interviews served us in two ways. First, it helped us adapt the items that could be used to operationalize the PD items. Second, it served as a basis for developing new items using phrases or sentences practitioners used to describe PD. These transcripts and the operationalized constructs found in the literature made it possible to develop an initial list of construct items. Table 1 shows an overview of the sources based on which we created new items.

Table 1. Construct definitions and sources of initial items.

Construct	Construct definition	Initial Items adapted from:
Process Unsuitability Debt	The degree to which processes are not aligned with the needs of an organization, causing inefficiencies.	Doolen et al., [6], Thomas et al., [24], + self-created items
Synchronization Debt	The degree to which synchronization impacts productivity and disrupts workflows.	Thomas et al., [24] + self-created items
Roles Debt	The degree to which there is a discrepancy between roles and responsibilities in the organization	Gray-Stanley & Muramatsu, [10] Rizzo et al., [23], Thomas et al., [24]
Documentation Debt	The degree to which documentation does not fit the actual work processes.	Daft & MacIntosh, [4], Torkzadeh & Doll [25] + self-created items
Infrastructure Debt	The degree to which the tools are poorly integrated, outdated, or unfit for the work processes.	Torkzadeh & Doll [25], + self-created items

In Table 1, the expression "+ self-created items" means that items were constructed and formulated based on the practitioner interview transcripts. The practitioners discussed and ranked these, and from the list of highest-ranked items, we ended up with a pool of eight to ten items per construct.

3.3. Stage Three: Item Identification and Selection

The item selection stage aimed to select items with a high content validity and drop items with low validity. To do this, we asked four researchers with experience in software development and project management theory to evaluate the content validity of our items. The researchers received a questionnaire in which they were provided with the conceptual definition of each construct and the related item pool. They rated how well each item represents the intended construct on a seven-point Likert Scale. In addition, the researchers could write comments or questions that helped us select appropriate items and further improve them. For each construct, we selected the seven best items based on the highest mean (M) and median (Med) [22]. All items showed a very good content validity ($4.0 < M < 7$; $4 < Med < 5$).

3.4. Stage Four: Item Revision and Pilot Test

Based on the selected items, we developed the first version of the survey that we pre-tested to twenty practitioners from one of our industry partners. Besides a link to the online survey questionnaire, we e-mailed instructions to the practitioners that they should, for each item, make notes on thoughts or questions they might have. Eighteen e-mail answers were collected containing the participants' comments and ideas for improvement. This gave us additional information on how the survey could be improved. We revised the wording of the overall questionnaire and kept four to five items per construct. To statistically assess the reliability and validity of our instrument, we conducted a pilot test where we invited practitioners from one department at one of our industry partners to participate. Overall, 63 participants replied and filled out the survey. We then conducted an explorative factor analysis to examine the reliability and validity of the overall instrument. We found all intended constructs with loadings showing high convergent and discriminant validity. The items that indicated problems in meeting the required validity and reliability thresholds were changed and adapted. An open-ended question where practitioners were asked to provide comments further improved the wording of the items. A few items showing somewhat low factor loadings were removed, and the final version of the questionnaire contained four items for each construct. The final measurement instrument is provided in Appendix A.

3.5. Stage Five: Field Tests

We applied our measurement instrument in a field survey to demonstrate internal and external validity. Data was collected through an online survey between October and December 2023 within two large ASD companies. They used similar Agile processes and ceremonies, such as sprint planning and Daily Scrums. This was the first test to understand PD types and variances, so we intended to avoid getting noisier data by looking at more than two organizations. By looking at only two similar organizations, we could control for other context factors, such as differing Agile frameworks. Instead, we focused on understanding PD within only these two organizations, thereby increasing internal validity.

Table 2. Factor loadings and reliability measures.

Construct	**Item**	**Loading**	**Sig.**	**Cronbach's Alpha**	**Composite Reliability (CR)**	**Average Variance Extracted (AVE)**
Process Unsuitability Debt	PU_1	0.830	< .001	0.80	0.81	0.54
	PU_2	0.700	< .001			
	PU_3	0.735	< .001			
	PU_4	0.709	< .001			
Roles Debt	RD_1	0.627	< .001	0.74	0.77	0.55
	RD_2	0.625	< .001			
	RD_3	0.819	< .001			
	RD_4	0.852	< .001			
Synchronization Debt	SD_1	0.687	< .001	0.80	0.81	0.58
	SD_2	0.801	< .001			
	SD_3	0.740	< .001			
	SD_4	0.822	< .001			
Documentation Debt	DD_1	0.892	< .001	0.81	0.83	0.69
	DD_2	0.928	< .001			
	DD_3	0.820	< .001			
	DD_4	0.662	< .001			
Infrastructure Debt	ID_1	0.840	< .001	0.82	0.83	0.62
	ID_2	0.769	< .001			
	ID_3	0.808	< .001			
	ID_4	0.716	< .001			

We asked practitioners in different roles within the software development organizations to participate in our study. To contact potential participants, the target organizations distributed the link to our online questionnaire to their employees. To motivate participation, we offered insights into the results and sent a reminder two weeks after the initial contact. The link was distributed to 487 employees, of which 197 filled out our questionnaire, which resulted in a response rate of 40.6%. Since our analysis required the absence of missing data, we list-wise deleted cases for which one or multiple items had missing values. This decreased the sample size for the analysis from 197 to 184, which is less than 7%, and should not be a concern as no specific variable was concerned since the missing values followed a random pattern [11].

The practitioners who participated in our study were from two Swedish companies. Regarding work experience, there was a relatively even distribution among the participants, as 16.8% had more than 30 years of experience, 26.7% had more than 20 years, 19.6% had more than 10 years, and 36.9% had less than 10 years of work experience. Most participants were team members, i.e., designers, developers, and testers working in the teams (60.9%); the others were Scrum masters (13.0%), product owners (8.2%), managers (8.2%), and stakeholders (people in other roles interested in the results) in the organization (10.8%).

We started by examining the validity and reliability of our measurement instrument through confirmatory factor analysis (CFA) in the statistical software tool R. Each item was modeled as a reflective indicator of its hypothesized latent construct, and all latent constructs were allowed to co-vary. No identification issues appeared, such as large standard errors for one or more coefficients, negative error variances, or standardized loadings higher than 1 [15]. A solution was found in 9 iterations. Factor loadings are displayed in Table 2.

According to Hair et al. [11], all indicator loadings should be at least .5 and ideally .7 or higher. All our measures met these thresholds and were significant at $p < .001$, with only RD_1, RD_2, SD_1, and DD_4 being below .7. Overall, the measures seem strongly related to their associated construct. Likewise, all constructs showed sufficient reliability with a Cronbach's alpha > 0.7 and Composite Reliability > 0.5 [13].

Convergent validity was tested using three criteria of Fornell and Larcker [9]. First, as seen from Table 2, all item loadings were significant, and all exceeded the threshold of 0.6. Second, construct Composite Reliability exceeded 0.8 for all constructs. Third, the Average Variance Extracted (AVE) was higher than 0.5. Thus, also convergent validity was ensured. Discriminant validity was tested by comparing the square root of each Average Variance Extracted (AVE) in the diagonal with the correlation coefficients of the other constructs (see Table 3). The square root of AVE should be greater than the corresponding correlation coefficients. As each column in Table 3 shows, the top value (diagonal value) is the highest among all other values, signifying that these constructs are not related.

Table 3. Factor correlation matrix with square roots of AVE.

	PU	**RD**	**SD**	**DD**	**ID**
Process Unsuitability Debt (PU)	**0.73**				
Roles Debt (RD)	0.71	**0.74**			
Synchronization Debt (SD)	0.70	0.71	**0.76**		
Documentation Debt (DD)	0.60	0.50	0.56	**0.83**	
Infrastructure Debt (ID)	0.63	0.45	0.58	0.49	**0.79**

Overall, the analysis demonstrated our developed measurement instrument's reliability and validity but raised concerns about model fit. Goodness of fit statistics for the overall structural model was as follows: GFI = 0.98, CFI = 0.92, RMSEA = 0.108, $\chi 2 = 564$, df = 179, p = < 0.001. While the GFI suggests a good fit and the CFI a reasonable fit, the high RMSEA value raises concerns about the model's adequacy. This might suggest that while the model captures some aspects of the data well, it might miss out on certain nuances or complexities inherent in the data.

4. Discussion

The effects of PD extend beyond immediate operational inefficiencies to impact the broader software business, underscoring the need for strategic approaches to identify, measure, and address these inefficiencies [18]. Mitigation requires a thorough understanding of the PD causes and a commitment to investing in resources and training to alleviate its impact [19]. In this paper, we developed a new instrument to measure PD, which we believe may be useful for measuring the different types of PD in an organization. By detailing and operationalizing five distinct PD types, we have developed a survey instrument that, in its first iteration, is usable for measuring and understanding the PD phenomenon.

We contribute to the growing field of non-technical debt [1] research by introducing a practical measurement instrument for addressing PD within ASD organizations. Moreover, our study's findings present the interplay between different types of PD. By doing so, we underscore the importance of a multidimensional approach to PD management. In practical terms, the instrument developed through this research holds the potential to improve organizations' approaches to identifying and managing PD significantly. Organizations can better prioritize interventions, allocate resources more effectively, and ultimately enhance their agility and competitiveness in the fast-paced software development arena by providing a means to measure and analyze PD systematically.

4.1. Limitations

Our study is not without limitations. We examined our measurement instrument in only two case organizations and suggest that future research apply it to other organizations to demonstrate external validity. This is a first step towards creating a good measurement instrument for PD. The reliability and validity of the measurement instrument show some goodness of fit statistics, particularly the GFI (0.98) and CFI (0.92). However, the RMSEA and $\chi 2$ test suggest that the model needs to be refined to improve the instrument [7] further. We worded several items in our measurement instrument in relative rather than absolute terms. This allows for wider applicability of the metrics yet also induces potential response bias as the individual response anchor may vary.

5. Conclusion

This paper contributes to the field of non-technical debt research and introduces a novel survey instrument designed to measure PD in ASD organizations. The instrument operationalizes and quantifies five specific types of PD: Process Unsuitability Debt, Synchronization Debt, Roles Debt, Documentation Debt, and Infrastructure Debt. Our validation within two ASD organizations demonstrates that the instrument can capture the nuances of PD and offers a systematic approach for organizations to identify PD. By providing organizations with the means to systematically measure PD, our instrument aids in prioritizing mitigation strategies, thereby enhancing organizational agility.

Further research and refinement of the measurement instrument will enhance its precision, and future studies are encouraged to apply this instrument in diverse settings to validate its applicability.

Acknowledgements

This research was performed within the NODLA research project, funded by the KK Foundation in Sweden.

References

1. Ahmad, M.O., Gustavsson, T.: The Pandora's box of social, process, and people debts in software engineering. Journal of Software: Evolution and Process, 36(2), (2022)
2. Alves, N.S., Ribeiro, L.F., Caires, V., Mendes, T.S., Spínola, R.O.: Towards an ontology of terms on technical debt. In: 2014 Sixth Int. WS on Man. T. D., 1-7 (2014)
3. Besker, T., Ghanbari, H., Martini, A., Bosch, J.: The influence of Technical Debt on software developer morale. J. Syst. Softw., 167, (2020)
4. Daft, R. L., MacIntosh. N. B.: A tentative exploration into the amount and equivocality of information processing in organizational work. ASQ 26, 207-224 (1981)
5. Dittrich, Y., Floyd, C., Klischewski, R.: Social Thinking-Software Practice. MIT Press (2002)
6. Doolen, T.L., Hacker, M.E., Van Aken, E.M.: The impact of organizational context on work team effectiveness. IEEE T. E. M., 50(3), 285-296 (2003)
7. Drury, M., Conboy, K., Power, K.: Obstacles to decision making in Agile software development teams. Journal of Systems and Software, 85(6), 1239-1254 (2012)
8. Evermann J., Tate M.: Fitting covariance models for theory generation. Journal of the Association for Information Systems. 12(9), 632–661 (2011)
9. Fornell C., Larcker D.F.: Structural equation models with unobservable variables and measurement error: Algebra and statistics. J. Market. Res. 18(3), 382–388 (1981)
10. Gray-Stanley, J.A., Muramatsu, N.: Work stress, burnout, and social and personal resources among direct care workers. Res. Developm. Dis., 32(3), 1065-1074 (2011)

11. Hair, J.F.J., Black, W.C., Babin, B.J., Anderson, R.E.: Multivariate Data Analysis. New Jersey, Prentice Hall (2010)
12. Hoda, R., Norsaremah, S., Grundy, J.: The rise and evolution of agile software development. IEEE Software, 35(5), 58-63 (2018)
13. Jöreskog, K.G., Sörbom, D., du Toit, S., du Toit, M.: LISREL 8: New Statistical Features. Lincolnwood, Illinois: Scientific Software International (2001)
14. Kruchten, P., Nord, R.L., Ozkaya, I.: Technical debt: from metaphor to theory and practice. IEEE Softw. 29(6), 18-21 (2012)
15. Lei, P.W., Wu, Q.: Introduction to structural equation modeling: Issues and practical considerations. Edu. Measurement: Issues and Practice. 26(3), 33–43 (2007)
16. Lenarduzzi, V., Besker, T., Taibi, D., Martini, A., Fontana, F.A.: A systematic literature review on technical debt prioritization. J Syst Softw. 17(1), (2021)
17. MacKenzie, S.B., Podsakoff, P.M., Podsakoff, N.P.: Construct Measurement and Validation Procedures in MIS and Behavioral Research. MISQ. 35(2), 293–334 (2011)
18. Martini, A., Besker, T., Bosch, J.: Process debt: A first exploration. In: 2020 27th Asia-Pacific Software Engineering Conference (APSEC). IEEE (2020)
19. Martini, A., Stray, V., Besker, T., Brede, M. N., Bosch, J.: Process Debt: Definition, Risks and Management. Available at SSRN: https://ssrn.com/abstract=4328073 (2023)
20. Martini, A., Stray, V., Moe, N. B.: Technical-, social- and process debt in large-scale agile: an exploratory case-study. In: Int. Conf. on ASD, 112-119, (2019)
21. Pedreira, O., Piattini, M., Luaces, M. R., Brisaboa, N. R.: A systematic review of software process tailoring. SIGSOFT Softw. Eng. Notes, 32(3), (2007)
22. Recker, J., Rosemann, M.: A measurement instrument for process modeling research: development, test and procedural model. SJIS. 22(2), 3-30 (2010)
23. Rizzo, J., House, R.J., Lirtzman, S.I.: Role conflict and ambiguity in complex organizations. Administrative Science Quarterly, 15, 150–163 (1970)
24. Thomas, C. L., Spitzmueller, C., Amspoker, A. B.,... Hysong, S. J.: A systematic literature review of instruments to measure coordination. J. Health. Mgmt. 63(3), (2018)
25. Torkzadeh, G., Doll, W. J.: The development of a tool for measuring the perceived impact of information technology on work. Omega, 27(3), 327-339 (1999)

Appendix A: Final Measurement Instrument Items (R = Reversed item)

Construct	Item	Items (1 = Strongly disagree … 7 = Strongly agree)
Process Unsuitability Debt	PU_1	Our work processes align with the business needs of the organization
	PU_2	Our work processes are updated and well-suited for our current way of working.
	PU_3	People are often unsure about which process to follow for specific tasks. (R)
	PU_4	The processes we must follow often result in duplicated work. (R)
Roles Debt	RD_1	My assigned tasks align with my official role in the organization.
	RD_2	I know what my responsibilities are
	RD_3	There is ambiguity about who is responsible for specific activities in our processes. (R)
	RD_4	People often perform tasks outside their designated roles due to unclear responsibilities. (R)
Synchronization Debt	SD_1	Multiple work processes overlap in a way that creates additional administrative overhead. (R)
	SD_2	Our work processes often conflict with each other, leading to inefficiencies. (R)
	SD_3	Lack of proper coordination disrupts my individual workflow. (R)
	SD_4	Poor coordination has led to errors that could have been avoided. (R)
Documentation Debt	DD_1	Our work processes are well-documented and easy to understand.
	DD_2	The level of detail in our process documentation is well-balanced.
	DD_3	Our process documentation is frequently updated to be relevant.
	DD_4	The terminology used in our process documentation often leads to misunderstandings. (R)
Infrastructure Debt	ID_1	We have suitable tools in place that increase the efficiency of our processes.
	ID_2	Our organization invests in updating or acquiring new tools to help make our processes work better
	ID_3	We have an effective tool integration that minimizes task-switching and streamlines operations.
	ID_4	Tools are often misused due to a lack of better alternatives. (R)