UNIVERSITY OF TURKU
Turku School of Economics

Coordination in process automation initiatives

Artefact design for enhanced collaboration

Information Systems Science

Master's thesis

Author:
Jussi Sirkiä

Supervisor:
KTT Reima Suomi

16.03.2026

Turku

Student's statement regarding the use of Artificial Intelligence (AI) for preparing and/or writing this thesis:

☐ **I have not used any AI-based tools.**

☒ **I have used AI-based tools**. Their use is documented in Appendix 3. The AI tools were used in a way that complies with academic integrity guidelines.

The originality of this thesis has been checked in accordance with the University of Turku quality assurance system using the Turnitin Originality Check service.

Master's thesis

Subject: Information Systems Science
Author: Jussi Sirkiä
Title: Coordination in process automation initiatives
Supervisor: KTT Reima Suomi
Number of pages: 82 pages (+ appendices 13 pages)
Date: 16.03.2026

Abstract

Process automation is often presented as a fast and technically straightforward route to organizational improvement. Yet many automation initiatives encounter difficulties long before the limits of the underlying technologies are reached.

This study examines that tension by approaching process automation not primarily as a technical implementation challenge, but as a coordination-intensive organizational practice. The study asks, first, how coordination breakdowns manifest in the early stages of process automation work and, second, how artefact-level structuring can be designed to mitigate them. Drawing on a design science research approach, the study combines a literature-based analysis with empirical inquiry into organizational process automation work. The empirical material consists of semi-structured interviews with members of an in-house process automation team, followed by a participatory design workshop used to derive artefact design requirements and design principles.

The findings show that early-stage coordination is often characterized by unstable or incomplete shared process understanding, alongside diffuse responsibility for defining and validating process logic. In addition, process articulation is at times constrained by social sensitivities that discourage explicit clarification. Based on these findings, the thesis develops design knowledge aimed to address the identified coordination breakdowns. The design knowledge is instantiated in design requirements and higher-level design principles which are operationalized into two practical artefacts: a pre-assessment form and a shared specification document embedded in the automation lifecycle. The study provides a grounded account of how early-stage automation work can be structured to better support the translation of business knowledge into automation logic.

Keywords: process automation, coordination breakdowns, shared process understanding, artefact design, design principles

Pro gradu -tutkielma

Oppiaine: Tietojärjestelmätiede
Tekijä: Jussi Sirkiä
Otsikko: Koordinaatio prosessiautomaatiohankkeissa
Ohjaaja: KTT Reima Suomi
Sivumäärä: 82 sivua (+liitteet 13 sivua)
Päivämäärä: 16.03.2026

Tiivistelmä

Prosessiautomaatio esitetään usein nopeana ja teknisesti suoraviivaisena keinona kehittää organisaation toimintaa. Monet automaatiohankkeet kuitenkin kohtaavat vaikeuksia jo kauan ennen kuin käytettävien teknologioiden rajat tulevat vastaan.

Tässä tutkimuksessa tätä jännitettä tarkastellaan lähestymällä prosessiautomaatiota ensisijaisesti koordinaatiointensiivisenä organisatorisena käytäntönä eikä vain teknisen toteutuksen haasteena. Tutkimuksessa kysytään ensinnäkin, miten koordinaation häiriöt ilmenevät prosessiautomaatiotyön varhaisissa vaiheissa, ja toiseksi, miten työn jäsentämistä artefaktitasolla voidaan kehittää näiden haasteiden lieventämiseksi. Tutkimus perustuu design science research -lähestymistapaan ja yhdistää kirjallisuuteen pohjautuvan analyysin empiiriseen tarkasteluun organisaation prosessiautomaatiotoiminnasta. Empiirinen aineisto koostuu organisaation sisäisen prosessiautomaatiotiimin jäsenten puolistrukturoiduista haastatteluista, joita seurasi osallistava suunnittelutyöpaja artefaktien suunnitteluvaatimusten ja suunnitteluperiaatteiden johtamiseksi.

Tulokset osoittavat, että varhaisen vaiheen koordinaatiota leimaavat usein epävakaa tai puutteellinen yhteinen prosessiymmärrys sekä epäselvä vastuu prosessilogiikan määrittelystä ja validoinnista. Lisäksi prosessin näkyväksi tekemistä rajoittavat toisinaan sosiaaliset herkkyydet, jotka voivat estää eksplisiittistä täsmentämistä. Näiden havaintojen pohjalta tutkimuksessa tuotetaan suunnittelutietoa, jonka tavoitteena on vastata tunnistettuihin koordinaation häiriöihin. Tämä suunnittelutieto jäsennetään suunnitteluvaatimuksiksi ja ylemmän tason suunnitteluperiaatteiksi, jotka konkretisoidaan kahdeksi käytännölliseksi artefaktiksi: esiarviointilomakkeeksi ja automaation elinkaareen kytketyksi yhteiseksi määrittelydokumentiksi. Tutkimus tarjoaa empiirisesti perustellun kuvauksen siitä, miten prosessiautomaatiotyön varhaisia vaiheita voidaan jäsentää siten, että liiketoimintatiedon kääntäminen automaatiologiikaksi saa parempaa tukea.

Avainsanat: prosessiautomaatio, koordinaation häiriöt, yhteinen prosessiymmärrys, artefaktien suunnittelu, suunnitteluperiaatteet

TABLE OF CONTENTS

FIGURES

TABLES

1 Introduction

1.1 Research area and motivation

Process automation has attracted significant interest in organizations seeking to automate their business processes (Ng et al., 2021). Technologies such as robotic process automation (RPA) and AI-enabled automation are widely promoted as lightweight, cost-efficient solutions for improving operational efficiency, reducing manual effort, and increasing process reliability. Unlike earlier generations of enterprise systems, these automation technologies are non-intrusive and can be deployed on top of existing information systems (Enriquez et al., 2020). Low technical barriers enable organizations to automate tasks and processes quickly and flexibly without extensive system redesign. As a result, process automation is often portrayed as a pragmatic and lightweight route to digital transformation (Penttinen et al., 2018).

Despite this promise, empirical evidence consistently indicates that process automation initiatives frequently struggle to deliver their intended outcomes. Research suggests that moving beyond initial pilots, scaling automation to further implementations, and sustaining automated processes over time require additional governance, support processes, and organizational capabilities. (Herm et al., 2023.)

High failure rates, rework, and initiatives that fail to deliver value are reported in both academic literature and practitioner discourse (Eulerich et al., 2024; Cascais Brás et al., 2025). These challenges present a paradox. Although process automation technologies are described as mature and technically capable, their organizational implementation remains fraught with problems. A common response to this paradox has been to attribute failures to shortcomings in technology selection, tool capabilities, or technical implementation, but research suggests that such explanations are insufficient (Götzen et al., 2020).

Process automation does not operate in isolation from organizational structures and routines. Instead, automation initiatives are embedded in everyday work and reshape how processes are identified, modified, and executed. Automation therefore constitutes not only a technical intervention, but an organizational practice that spans business units, automation teams, and central IT functions. From this perspective, the challenges observed in process automation initiatives cannot be fully understood through a purely technical lens. Rather, they point to difficulties in coordinating work, knowledge, and responsibilities across organizational boundaries. (Götzen et al., 2020.)

Particularly early stages of automation work require actors with different forms of expertise to align their understanding of how a process is performed, how it varies in practice, and whether it can be translated into executable logic (Kraus et al., 2024). When this alignment is weak, organizations risk automating unsuitable processes, building automations on incomplete or informal knowledge, or creating solutions that fail to meet operational expectations (Cascais Brás et al., 2025). Prior research highlights several recurring problem areas in this regard. Organizations often struggle with limited or fragmented understanding of their own processes, reliance on tacit knowledge, and insufficient documentation (Kraus et al., 2024). Decisions about what to automate require systematic evaluation, yet organizations often lack clear criteria for assessing process suitability, data requirements, and process stability before moving forward with automation (Farinha et al., 2024). At the same time, responsibilities for process ownership, automation development, and ongoing maintenance are often distributed across multiple organizational units, leading to ambiguity and coordination breakdowns (Cascais Brás et al., 2025). These issues tend to originate in early stages of automation initiatives, long before technical limitations of the automation tools themselves are reached.

Although the literature on process automation acknowledges these challenges, systematic empirical research examining how they manifest in organizational practice remains limited, particularly in the early stages of automation work (Kraus et al., 2024). Prior research commonly characterizes suitable processes as already "structured, mature, standardised, repetitive and well-documented," implying that substantial process understanding is expected to be in place before automation begins (Ng et al., 2021). However, translating process knowledge into automation-ready form often requires intensive coordination between business domain experts and automation specialists to document process steps, rules, exceptions, and feasibility (Enriquez et al., 2020). Despite this reliance on collaborative articulation, prior research provides limited methodological guidance for structuring this early coordination work, even though effective automation depends on the involvement of domain experts, end users, and technical specialists to capture and translate process knowledge properly (Syed et al., 2020).

This thesis addresses this gap by examining process automation as a coordination-intensive organizational practice and by focusing on the role of artefacts in supporting this work. Artefacts such as process descriptions, templates, and evaluation tools play a central role in mediating collaboration between business units and automation teams. When well designed, they can help make implicit process knowledge explicit, support shared understanding across organizational boundaries, and structure decision making. When absent or poorly aligned with practice, coordination increasingly

relies on informal communication and ad-hoc problem solving, increasing the risk of misunderstandings and failed automation initiatives.

On this background, this thesis adopts a two-layer research design that combines analytical inquiry and design research. First, the study seeks to examine how coordination challenges manifest in organizational process automation practice. Although prior literature identifies recurring implementation difficulties in process automation, there remains limited empirical understanding of how coordination breakdowns materialize in early-stage automation work and how they shape subsequent development and implementation. The first research question (RQ) therefore addresses the analytical instantiation of coordination breakdowns in context:

RQ1: How do coordination breakdowns manifest in early-stage process automation work?

Second, building on the identified coordination breakdowns, the study aims to develop artefact-level design responses that mitigate these instabilities. Rather than proposing abstract governance principles or high-level frameworks, the study focuses on structuring concrete artefacts that stabilize shared process understanding and clarify responsibility across organizational boundaries. The second research question therefore adopts a design-oriented perspective:

RQ2: How can artefact-level structuring be designed to mitigate these coordination breakdowns?

Together, these research questions form a sequential logic. The first question grounds the study in empirical analysis of coordination instability in early-stage automation work, while the second translates these insights into design principles and artefact instantiations intended to address the identified challenges. Rather than addressing automation success in general terms, the study focuses on recurring coordination problems that appear across organizations and that remain insufficiently supported by existing frameworks. The resulting artefacts are intended to complement existing process automation frameworks by providing concrete operational support for process explicitation, pre-automation screening, and responsibility alignment.

Methodologically, this thesis adopts a design research approach grounded in empirical inquiry. Qualitative data collected from practitioners involved in process automation is used to identify recurring coordination challenges and to inform the design of artefacts intended to address them.

The remainder of the thesis is structured as follows. Chapter 1.2 provides background on process automation technologies and their technical characteristics. Chapter 2 develops the theoretical framing of process automation as an organizational practice, synthesizing literature on coordination, artefacts, and automation failure. Chapter 4 presents the methodology of this study and chapter 5 presents the empirical findings. Chapter 6 describes the design of the proposed artefacts. Finally,

chapter 7 discusses the implications of the findings for research and practice and outlines directions for future work.

1.2 Process automation technology

Process automation technology, as discussed in this thesis, encompasses a variety of technologies designed to automate tasks previously done by humans. This chapter introduces three distinct but, in terms of their definitions, partially overlapping technologies: robotic process automation, intelligent automation, and AI agents. The common denominators of these technologies are their purpose of automating human-executed processes and the technical characteristic that their functionality and execution logic are implemented outside existing enterprise information systems. More specifically, these technologies introduce an external automation layer that operates on top of established systems, rather than being embedded within their core application logic. Due to these shared characteristics, it is arguably practical to group the technologies under the umbrella term “process automation” when discussing their shared architectural or organizational implications, which is the approach taken in this thesis.

Due to operating on top of current systems rather than modifying their internal structure, process automation enables a substantial portion of process logic and orchestration to be shifted away from core enterprise systems into this external automation layer (Enríquez et al., 2020; Herm et al., 2022). From a technical perspective, this enables system-level digitalization to proceed in incremental and distributed steps, as automation can be added without redesigning or replacing existing applications. Consequently, automated processes can be introduced with limited impact on underlying system architectures and without the need for costly, risky, and time-consuming modifications to enterprise systems (Lacity and Willcocks, 2016; Syed et al., 2020).

Robotic process automation (RPA) is commonly described as a software-based automation technology that enables software robots to execute structured, rule-based, and repetitive tasks by mimicking human interactions with existing information systems (Syed et al., 2020; Wewerka and Reichert, 2023). Some sources such as Lacity & Willcocks (2016), Götzen et al. (2022) and Herm et al. (2022) define RPA as explicitly limited to user interface interactions while others, such as Cascais Brás et al. (2025) include system integration through existing APIs as a feature allowed under the RPA term. The ambiguity of the definitions is particularly well captured by Syed et al. (2020), whose structured literature review found that out of the 125 publications reviewed, only 24 provided definitions for the term RPA. Despite the definitional ambiguity, the key definitions for RPA include

an outside-in approach in which functionality is built on top of existing systems to accomplish processes consisting of repetitive, rule-based tasks previously performed by humans.

Intelligent automation (IA) is usually positioned in the literature as an extension of RPA through embedding AI technologies such as machine learning and natural language processing. According to this viewpoint, IA expands the scope of automation beyond strictly rule-based execution by enabling the automation to handle unstructured data, undertake probabilistic classification and make context-aware decisions. Instead of being a replacement for RPA, IA builds on top of the capabilities of RPA by augmenting it with AI to enable visual perception and context aware interpretation and decision-making capabilities. (Wewerka & Reichert, 2023; Ribeiro et al., 2021, Afrin et al., 2025)

AI agents are a novel form of automation that has emerged along the rapid evolution of large language models (LLMs). Xi et al. (2025) define AI agents as artificial entities that can sense their environment, make decisions and act accordingly. By this definition AI agents are distinctly different from IA and RPA. On the other hand, Deng et al. (2025) define AI agents as software that performs tasks autonomously or makes decisions based on pre-defined inputs desired outputs, which makes the distinction between AI agents and IA ambiguous.

2 Process automation as an organizational practice

The literature indicates that process automation should not be seen only as a technical tool. Automation is part of a wider organizational system that includes technical, social, and organizational elements. This means that successful automation depends not only on whether the technology works, but also on how work is organized, how responsibilities are shared, and how people and technology work together in practice. In this sense, process automation is not just a technology, but also an organizational practice. (Götzen et al., 2020.)

A key socio-technical implication emphasized in the literature is that process automation changes how process change is organized and governed, rather than simply what is automated. Because automation logic is external from core systems, organizations can introduce, modify, or remove automation with relatively low technical barriers. This characteristic enables process changes to be initiated closer to operational work, often by business units or process owners rather than through centralized IT-driven transformation programs (Herm et al., 2022). Consequently, process automation becomes less of a one-time technical intervention and more of an ongoing organizational activity, shaped by local knowledge, operational priorities, and evolving interpretations of what constitutes an automatable process (Wewerka and Reichert, 2023).

While the previous chapter introduced the architectural implications of process automation technologies, this chapter adopts an organizational and coordination perspective to address how process automation is implemented in practise.

2.1 Coordinating process automation work across boundaries

From an organizational perspective, challenges in process automation work can be understood as coordination problems. Implementing process automation in an organization consists of interdependent activities that must be aligned across actors, roles, and units. Coordination theory provides a useful theoretical lens for examining such work.

Malone and Crowston (1994) define coordination theory as an interdisciplinary perspective that studies how interdependent activities are organized and aligned. Rather than focusing only on the internal content of individual tasks, it directs attention to the dependencies between activities and to the mechanisms that manage these dependencies. In organizations, such coordination mechanisms may include roles, routines, rules, schedules, handovers, and shared artefacts. This makes coordination theory relevant for process automation, where implementation depends not only on

technical development but also on aligning contributions across multiple actors and entities. (Malone & Crowston, 1994.)

In the context of process automation, coordination spans multiple organizational boundaries. Business units and process automation specialists typically operate with different expertise, objectives, and vocabularies. Business units possess domain specific and often tacit knowledge of how work is performed, while automation specialists bring technical expertise related to automation tools and implementation. Successful automation therefore depends on coordinating interdependencies related to process knowledge (what the process is and how it varies), timing (when information, decisions, and artefacts are needed), and responsibilities (who is accountable for defining, validating, and maintaining different aspects of the automated process). When these interdependencies are not adequately managed, coordination relies heavily on informal communication, repeated meetings, and ad-hoc problem solving. This increases effort and uncertainty and makes misunderstandings more likely. Many of the challenges seen in process automation initiatives, including poorly defined processes, misaligned expectations, late identification of infeasible solutions, and unclear ownership, are better understood as coordination failures rather than shortcomings of the automation technology itself (Crowston, 1997).

Viewing process automation work through the lens of coordination shifts analytical attention away from purely technical explanations and toward the mechanisms of collective action across organizational boundaries. This framing provides a foundation for analyzing implementation challenges and for motivating the design of artefacts intended to support coordination in early-stage process automation work. While coordination theory highlights the need to manage interdependencies across organizational boundaries, it does not by itself explain how mechanisms of coordinated action are established in practice. This requires closer attention to the role of artefacts in mediating collaboration between heterogeneous actors.

2.2 Artefacts, process knowledge, and shared understanding

The nature of process knowledge itself presents a central challenge for coordinating process automation work. Process knowledge in complex organizations is often tacit, distributed, and context-dependent, residing across multiple actors and organizational units rather than in formal documentation. This is particularly pronounced in cross-boundary processes, where no single actor has a complete view of the process. As a result, establishing a shared and explicit understanding of the process becomes a non-trivial task.

Boundary Object theory offers a useful perspective for understanding how such shared understanding can be achieved and maintained. Boundary objects are artefacts used by multiple social groups that enable coordination despite differences in expertise, perspectives, and interests. They are sufficiently flexible to be interpreted locally by different actors, yet structured enough to maintain a common identity across contexts. Rather than eliminating differences in understanding, boundary objects allow actors to work together productively without requiring full consensus. (Star & Griesemer, 1989.)

In the context of process automation, many of the artefacts commonly used in automation initiatives (e.g. process descriptions, requirement templates and various forms of documentation) can be understood as potential boundary objects. When well designed, such artefacts can support the articulation of implicit process knowledge, make assumptions explicit, and provide a shared reference point for discussion and decision-making across boundaries. However, when boundary objects are absent, under-specified, or poorly aligned with the needs of participating actors, coordination increasingly relies on informal communication, repeated meetings, and individual interpretation. This places cognitive burden on participants and can increase the likelihood of misunderstandings, late rework, and misaligned expectations. From this perspective, many of the difficulties encountered in process automation initiatives can be traced to the lack of effective artefacts that support shared understanding across organizational boundaries.

Viewing automation artefacts as boundary objects highlights their role not merely as documentation, but as active coordination mechanisms. This perspective provides a theoretical basis for examining how existing practices and frameworks support, or fail to support, their effective use. This perspective also motivates the design of new artefacts intended to improve early-stage collaboration in process automation work.

2.3 Implementation challenges and failure discourse

The literature repeatedly points out that the implementation of process automation is frequently subject to significant challenges, and in many cases, outright failure. For example, Herm et al. (2022) cite estimates suggesting that up to 50% of initial RPA projects fail to deliver their intended outcomes. Similarly, Huang and Vasarhelyi (2019) note that despite the apparent simplicity and promise of RPA, many implementations struggle to move beyond pilot stages or produce sustainable value. Likewise, Syed et al. (2020) emphasize that organizations often experience difficulties when attempting to scale process automation initiatives, even though the underlying technologies are portrayed as lightweight and easy to deploy. These findings establish that process automation implementation challenges are not isolated incidents but a recurring feature in the empirical and conceptual literature on process

automation. However, as noted by Kraus et al. (2024) in the literature review of their paper, there seems to be very little systemati empirical research that analyses the nature of challenges in process automation implementation.

A systematic literature review of 113 academic papers on RPA adoption across different industries by Pramod (2021), identified process selection as a central challenge in RPA adoption. Another structured literature review on 125 academic papers on RPA by Syed et al. (2020) supports the previously stated finding by noting that the literature acknowledges that selecting the right process for RPA is crucial, but how to do this is not entirely clear. The authors warn that applying RPA to unsuitable processes increases the development effort and undermines the chance of successful implementation right from the start (Syed et al., 2020). Huang and Vasarhelyi (2019) make similar claims from an auditing industry context. According to the authors, process selection is the most important control in process automation because automating the wrong process can lead to inevitable failure of the initiative or end up executing the process incorrectly.

From a coordination perspective, these findings highlight that early-stage decisions are made under conditions of limited shared understanding, where critical interdependencies between process characteristics and automation requirements are not yet fully articulated. The literature does not provide quantitative evidence on how failures in process selection translate into implementation failures or what the costs are. However, they clearly indicate that process selection is considered a critical point of failure in the process automation literature, as automating an unsuitable process is likely to cause the project to fail from the outset.

Proper process evaluation needs to be systematic and objective. This creates a prerequisite that the process needs to be documented in a way that allows for proper evaluation. In their qualitative interview study, Kraus et al. (2024) found three process related challenges that obstruct the ability to ultimately automate a process: lack of process understanding, incomplete process analysis and insufficient process optimization. Even when a group of individuals fully and accurately understands a process, formal documentation that establishes shared understanding is still necessary to objectively evaluate the process. On the other hand, if documentation exists but fails to capture the process in sufficient detail or in its entirety, it does not provide a sound basis for evaluation. Or if the documentation is adequate but the underlying process is poorly optimized or fundamentally flawed, the resulting automation initiative will be built on a crooked foundation.

This notion of flawed or poorly optimized processes leads to a new discussion of its own. In their research, Suri et al. (2017) surveyed 42 RPA professionals across industries. When prompted for

challenges in implementing RPA, 43% of respondents pointed to the lack of standard processes as a challenge for RPA implementation. This finding indicates that bad processes are often the problem. Notably, in the same survey 14% of respondents identified “The fear to automate a messy process instead of streamlining it...” as a challenge for RPA implementation. This illustrates a tension inherent in coordination work. While some level of process stabilization is necessary to enable automation, prolonged attempts to perfect processes can delay or block collective action altogether.

In their study Farinha et al. (2024) combined a systematic literature review and a fuzzy Delphi study with RPA experts to extract and rank criteria organizations should use when selecting processes for RPA. Out of the total of 32 criteria, automation feasibility was ranked number one and accurate process description was ranked number two. This constitutes clear empirical evidence supporting the finding that for process automation success the underlying process must be feasible, and it must be accurately documented. These findings form a triad of interdependent coordination challenges at the outset of automation initiatives. In principle, a process needs to be evaluated to avoid automating unsuitable processes, but the evaluation requires that the process is properly documented and to document a process, it needs to be explicitly understood. This set of dependencies is illustrated in Figure 1.

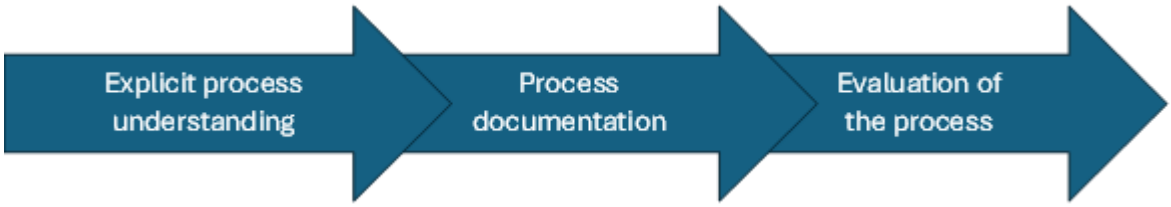


Figure 1. Coordination triad

Process understanding and documentation emerge in the literature not merely as analytical exercises, but as coordination mechanisms through which human work is made executable by software. In the study by Zhang et al. (2022) the authors emphasize that tasks are only automatable when they can be articulated as rule-based, repetitive actions with structured, machine-readable inputs. In addition, they note that many activities only become automatable after redesign clarifies and formalizes how humans perform the work. Similarly, Cascais Brás et al. (2025) report that practitioners repeatedly identify failures in RPA initiatives when process knowledge remains informal, implicit, or unevenly distributed between business and IT. Interviewees of the study stress that the business side must explicitly define rules and provide concrete examples of how work is performed manually, while IT assesses feasibility and translates this knowledge into step-by-step logic. When this translation is not

mediated by sufficiently robust artefacts that function as shared reference points, coordination breaks down, and robots are built on incomplete or ambiguous process descriptions and fail to deliver value.

The process automation literature identifies a range of organizational and governance challenges. From a coordination standpoint, these challenges relate primarily to the allocation and alignment of responsibilities across organizational boundaries. However, the findings are not generalizable, as they depend heavily on an organization's specific structure and operating context. Suri et al. (2017) report that 29% of respondents see challenges in the unclear division of responsibilities between IT and functional organizations. Herm et al. (2022) go deeper into this topic claiming that RPA initiatives driven by business units tend to lead to ambiguity over the division of responsibilities in things like development and maintenance. Cascais Brás et al. (2025) discuss a three-fold division of responsibilities. Business units are typically positioned as process owners, meaning that they are responsible for the business process and its outcomes. Process automation teams are responsible for development, maintenance and orchestration while a broader IT organization retains ownership of enterprise systems development, infrastructure, access management and security. The authors note that this overlap is often a source of confusion, since the domain of process automation spans all three areas of responsibility. The overlap increases coordination complexity and makes it difficult to establish clear handovers and accountability structures.

A lot of organizational challenges of process automation are centred around the centralization – de-centralization of choice. An extreme manifestation of de-centralized process automation is known widely as the citizen developer model. The core value proposition of the citizen developer model is high flexibility and fast delivery of process automation solutions. "Low code" RPA and AI automation tools enable development through drag and drop UIs which to some extent enables anyone to participate in process automation development. As a trade-off, this de-centralization increases governance complexity, raising risks related to inconsistent standards, weak change control, security exposure, and reduced auditability if clear roles, permissions, and centralized oversight are not in place. On the other hand, in the centralized model, process automation development and operation are concentrated within a central IT function or a "Center of Excellence," which exercises strong control over governance, standardization, and risk management. This enables consistent architecture, clear decision rights, robust change management, and improved compliance and auditability. As a trade-off, centralization can reduce flexibility and responsiveness, create delivery bottlenecks, and distance automation design from detailed business process knowledge, potentially slowing innovation and limiting local ownership. (Cascais Brás et al., 2025.)

Although not as widely reported, the process automation literature does report some technical challenges. RPA solutions are frequently quoted as fragile due to their dependence on user interfaces which are prone to change over time requiring rework in the automation. Herm et al. (2022), based on their analysis of 35 real-life RPA projects, report that operational teams frequently struggle with the instability of automated processes caused by changes in underlying systems. They note that even small interface changes, system updates, or altered data fields can interrupt automation execution, forcing local teams to spend disproportionate effort on monitoring and repair rather than on developing new automations.

Data-related issues are also reported as a recurring technical challenge at the operational level. Pramod (2021) notes that even when processes appear suitable for automation, inconsistent data formats, missing values, or undocumented exceptions frequently cause bots to fail during execution. Huang and Vasarhelyi (2019) similarly stress that automation tools assume stable and well-structured inputs, while real-world operational data often violate these assumptions.

When automation incorporates AI-based components, additional technical challenges surface at the point of use. Ng et al. (2021) report that machine-learning-based automation shifts effort from scripting rules to managing model performance, which includes monitoring accuracy, handling edge cases, and retraining models as conditions change. They emphasize that these activities are rarely embedded into everyday operational routines, leaving front-line teams unsure how to respond when automated decisions degrade or behave unexpectedly. Siderska et al. (2023) further highlight that limited explainability of AI-based decisions complicates troubleshooting, as employees cannot easily determine why a particular automated outcome occurred.

Finally, several studies highlight that testing and error handling remain underdeveloped in automation initiatives. Herm et al. (2022) report that operational teams often rely on informal testing practices, such as running bots on a small set of cases, rather than systematic validation. Eulerich et al. (2024) observe that insufficient logging and monitoring mechanisms make it difficult for local users to detect errors early, causing automation failures to surface only after downstream effects have already occurred.

These findings show that technical challenges in process automation frequently surface at points where coordination mechanisms, routines, and supporting artefacts are the weakest. Rather than stemming from fundamental limitations of process automation technologies, the problems arise from unstable target systems, imperfect data, limited transparency, and insufficient operational support

structures. This reinforces the view that technical challenges are tightly intertwined with how automation is introduced, maintained, and governed in practice.

In conclusion, when interpreted through the lenses of coordination theory and boundary object theory, the failure discourse in process automation literature consistently indicates that process automation initiatives tend to fail due to socio-technical and organizational factors rather than limitations in the automation tools themselves. Across studies, automation technologies are characterized as lightweight, mature, and technically capable of executing well-defined tasks (Lacity and Willcocks, 2016; Syed et al., 2020; Herm et al., 2022). Yet empirical evidence shows that failures emerge when these tools are embedded in organizational contexts marked by weak process understanding, fragmented ownership, unclear governance, and misaligned expectations between business and IT (Suri et al., 2017; Cascais Brás et al., 2025).

These conditions reflect persistent coordination breakdowns and the absence of effective artefacts that support shared understanding and the allocation of responsibilities. Challenges such as poor process selection, informal or incomplete documentation, unstable responsibilities, and insufficient change management systematically undermine implementation outcomes long before technical limits are reached. Even technical issues such as bot fragility, data quality problems, or limitations in AI explainability tend to manifest at the operational level because organizations lack the routines, roles, and capabilities needed to monitor, adapt, and sustain automation in everyday work (Herm et al., 2022; Ng et al., 2021).

Collectively, the failure discourse in process automation highlights not immature technology, but organizations struggling to reorganize work, knowledge, and governance around automation. Process automation therefore succeeds or fails primarily as a coordination-intensive organizational practice, in which technology amplifies existing strengths or exposes existing weaknesses in how processes are articulated, coordinated, and governed.

3 Existing process automation frameworks

This chapter examines existing process automation frameworks considering the coordination- and artefact-related challenges identified in the previous chapter. The selected process automation frameworks are evaluated through a set of analytical lenses and examined collectively to outline implications for later artefact design.

While there is a body of literature which proposes frameworks to address aspects of process automation, these frameworks vary considerably in their scope, level of abstraction, and intended use. Rather than treating these frameworks as prescriptive tools, this thesis approaches them as analytical representations of how coordination and process automation work are conceptualized in the literature. A systematic basis is therefore required to evaluate how well existing frameworks align to the challenges identified in the failure discourse. Accordingly, this thesis evaluates some existing frameworks (Table 2) using a set of analytical lenses derived from the coordination- and artefact-related challenges synthesized in chapter 2. The lenses function as analytical coding devices that make explicit how frameworks address coordination across organizational boundaries, how they support the articulation and stabilization of process knowledge, and where they leave critical interdependencies under-specified. This approach enables a structured and comparable analysis across heterogeneous frameworks without requiring each framework to be discussed exhaustively in its own terms.

3.1 Analytical lenses for framework evaluation

The first lens, process explicitation and formalization, captures how a framework supports the transformation of human work into a form that can be interpreted and executed by automation technologies. Building on the coordination challenges identified in chapter 2, automation presupposes not only that processes are known, but that they are articulated in a way that enables shared understanding across actors and subsequent formalization for rule-based execution. This lens therefore examines whether frameworks actively support the articulation and stabilization of process knowledge or implicitly assume mature and well-documented processes as a given.

The second lens, pre-automation suitability screening, addresses the widely reported risk of automating unsuitable processes. Prior research highlights process selection as a central point of failure, emphasizing that premature or ill-considered automation decisions can undermine initiatives from the outset. From a coordination perspective, this lens focuses on whether frameworks support early alignment around feasibility, risk, and exclusion criteria before development resources are

committed. It evaluates whether frameworks provide systematic mechanisms for assessing automation suitability and risk prior to implementation, or whether selection is treated primarily as prioritization among candidates without explicit exclusion or risk considerations.

The third lens, cross-unit coordination and responsibility alignment, reflects the organizational reality that process automation spans multiple units, including business functions, automation teams, and central IT. As covered in chapter 2, ambiguity in roles, decision rights, and ownership frequently leads to coordination problems and governance gaps. This lens therefore examines how frameworks support the management of interdependencies by clarifying responsibilities, decision rights, and handovers across organizational boundaries, and how explicitly such coordination is operationalized.

The fourth lens, operational artefact support, responds to the recurring observation that many frameworks remain conceptually sound yet difficult to enact in practice. Failures often emerge at the operational level, where abstract guidance does not translate into concrete actions or decisions. This lens assesses whether frameworks provide artefacts that function as shared reference points, such as templates, models, or decision aids, that support coordination and repeatable practice, or whether they rely primarily on high-level principles that require substantial local interpretation. All the lenses are shown in Table 1.

Table 1. Analytical lenses

Lens	Primary focus	Strong support	Weak support
(L1) Process explicitation and formalization	Support for making process knowledge explicit and formalizable	Acknowledges and supports process explication and shared understanding	Assumes that process documentation exists.
(L2) Suitability and risk screening	Support for assessing risks and process suitability	Screening criteria, risk & impact evaluation tools, go/no-go decisions	Prioritization only, no risk/impact evaluation or go/no-go decisions
(L3) Coordination and responsibility alignment	Roles, responsibilities and ownership	Addresses coordination of roles and responsibilities	Vague, high-level approach to governance.
(L4) Operational artefact support	Concrete tools for operational enactment	Templates, checklists, decision models etc.	High-level guidance only, dependent on interpretation

Together the lenses provide a structured means to evaluate existing frameworks against the challenges identified in the process automation failure discourse. They make explicit how frameworks support (or fail to support) coordination and shared understanding in early-stage automation work, while allowing for systematic comparison and maintaining sensitivity to differences in framework intent

and scope. The resulting analysis does not aim to rank frameworks, but to identify recurring gaps and limitations that motivate the need for additional support tailored to organizational practice.

3.2 Description of analyzed frameworks

Given the large and heterogeneous body of process automation frameworks, this thesis does not aim for exhaustive coverage. Instead, a purposeful selection strategy was applied to identify a limited set of representative frameworks. Frameworks were selected to reflect distinct problem framings in the literature, including implementation guidance, process selection, governance and responsibility allocation, and socio-technical enactment of automation, all of which address different aspects of coordinating automation work in organizations. In addition, selection considered variation in the level of abstraction and ensured scholarly relevance through peer-reviewed publication. This approach enables comparative analysis across key coordination- and artefact-related dimensions of the problem space while maintaining analytical depth. The frameworks selected for evaluation are visible in Table 2.

Table 2. Selected frameworks

Framework	Primary focus	Problem framing	Level of application	Framework category
Herm et al. (2022)	RPA implementation	Structuring RPA projects	Implementation	Implementation framework
Farinha et al. (2024)	RPA process selection and evaluation	Identifying business processes for RPA	Pre-implementation	Process selection and evaluation framework
Eulerich et al. (2024)	RPA governance	Risk mitigation for RPA	Post-implementation	Governance & control framework
Cascais Brás et al. (2025)	RPA governance and organizational integration	Aligning roles to manage risks, responsibilities, and scalability in RPA initiatives	Implementation / post-implementation	Governance framework

3.3 Framework evaluations

The framework by Herm et al. (2022) is an implementation-oriented framework for RPA projects, motivated by reported project failure rates and designed to provide methodological guidance across the phases of initialization, implementation, and scaling. The framework operates primarily at the

project and implementation level, with secondary implications for governance through constructs such as Centers of Excellence. Its core contribution lies in synthesizing findings from 35 reports on real-life RPA projects and expert interviews into a step-by-step framework intended for use across heterogeneous organizational contexts. Rather than addressing individual process articulation or automation logic, the framework focuses on stabilizing how RPA work is organized and coordinated over time.

From a process explicitation perspective (L1), the framework has weak support for process explicitation and formalization. It assumes candidate processes are already sufficiently understood and documented, offering no concrete support for articulating tacit or fragmented process knowledge into automation-ready form. Regarding pre-automation suitability and risk screening (L2), the framework provides partial support through selection criteria and pilot phases, but treats suitability primarily as portfolio prioritization rather than a structured go/no-go decision. In terms of cross-unit coordination and responsibility alignment (L3), governance is clearly acknowledged, particularly through the Center of Excellence concept, yet decision rights and operational responsibilities remain high-level and under-specified. Finally, with respect to operational artefact support (L4), the framework offers only conceptual guidance and does not prescribe concrete artefacts to support repeatable or auditable automation decisions. Overall, the framework strengthens organizational framing and procedural awareness of RPA implementation, but leaves critical coordination and translation work, where many failures empirically occur, to local interpretation rather than explicit methodological or artefact-based support.

The framework proposed by Farinha et al. (2024) is a decision-support framework for pre-implementation process selection in robotic process automation. It addresses the recurring problem that organizations automate unsuitable processes, thereby undermining RPA initiatives before development even begins. The framework provides a systematic, criteria-based method to evaluate candidate processes based on feasibility, data characteristics, stability, and expected value. Its main contribution is at the pre-implementation decision-making level, before implementation, governance, or operational execution. The framework is designed to support managerial judgments about what to automate, not how automation should be built, governed, or sustained.

Regarding process explicitation and formalization (L1), the framework treats process documentation as a prerequisite rather than a problem to be resolved. It requires accurate process descriptions but offers no guidance on how informal or tacit work should be translated into automation-ready logic. In terms of pre-automation suitability and risk screening (L2), the framework is comparatively strong,

providing a systematic, weighted set of criteria that explicitly accounts for feasibility, stability, exceptions, and data quality, and enables rejecting unsuitable candidates. However, suitability is treated as a static, upfront decision rather than a condition that may evolve over time. Concerning cross-unit coordination and responsibility alignment (L3), organizational roles and decision rights are largely implicit. Governance appears only as evaluative variables (e.g., cost, SLA impact) rather than as defined responsibilities or handoff structures. Finally, regarding operational artefact support (L4), the framework provides concrete and reusable selection artefacts (criteria lists, weights, scoring logic) but does not extend into artefacts for implementation, testing, monitoring, or maintenance. Overall, the framework is analytically rigorous as a selection instrument, but narrowly scoped, leaving key socio-technical and operational challenges unaddressed.

The framework proposed by Eulerich et al. (2024) is a governance-oriented RPA framework aimed at addressing risks, control deficiencies, and auditability concerns arising from the organizational use of robotic process automation. It conceptualizes RPA primarily as a governance, internal control, and risk management problem that emerges once bots are embedded in operational processes, rather than as a project-level implementation challenge. The framework targets organizations facing uncontrolled bot proliferation, unclear responsibilities, and weak internal controls, particularly where bots affect financially significant processes and reporting-related controls. Its primary level of application is governance and assurance, seeking to standardize oversight across decentralized units rather than to guide automation design or execution. As such, the framework positions itself as a central policy structure for governing RPA across the organization.

With respect to process explicitation and formalization (L1), the framework remains largely implicit, assuming that process knowledge is already sufficiently articulated and stabilized to support governance decisions. Regarding pre-automation suitability and risk screening (L2), it provides partial support through usage decision controls, but focuses on compliance and assurance risks rather than on operational feasibility, process volatility, or data-related fragility. In terms of cross-unit coordination and responsibility alignment (L3), governance areas and control requirements are clearly articulated. However, roles, decision rights, and handovers between business units, automation teams, and IT remain defined only at a high level, leaving substantial room for interpretation in practice. Finally, with respect to operational artefact support (L4), the framework offers reusable control requirements but relies heavily on local interpretation, providing limited concrete artefacts to support day-to-day coordination and operational work. Overall, the framework provides strong post-implementation governance support, but remains structurally detached from the artefact-level and coordination challenges that dominate early-stage process automation failure.

The framework by Cascais Brás et al. (2025) is a governance-oriented framework for RPA adoption and scaling, aimed at addressing coordination, role clarity, compliance, and risk issues that arise when automation spans business units, governance structures, and IT. It frames RPA as an organizational integration and governance challenge rather than a standalone technical initiative, emphasizing governance structures, decision rights, role clarity, and business-IT alignment. The framework operates primarily at the governance and organizational level, focusing on the conditions that support implementation, scaling, compliance, and sustainable use rather than process design or engineering.

With respect to process explicitation and formalization (L1), the framework treats process understanding as a prerequisite but leaves the articulation of process knowledge implicit. Regarding pre-automation suitability and risk screening (L2), it emphasizes process selection and risk awareness conceptually yet provides no systematic or enforceable mechanism to exclude unsuitable candidates upfront. In contrast, cross-unit coordination and responsibility alignment (L3) is addressed explicitly and in depth, with clear articulation of roles, decision rights, and governance models across business units, automation teams, and IT. Finally, in terms of operational artefact support (L4), the framework remains largely conceptual, discussing governance mechanisms such as access control and logging without translating them into concrete, reusable artefacts for everyday coordination and automation work. Overall, the framework is strong in aligning governance and responsibilities across organizational boundaries, but weak in operationalizing process articulation and artefact-level support that could mitigate early-stage automation failures. The evaluation of each framework is summarized in Table 3.

Table 3. Frameworks through analytical lenses

Source	(L1) Process explicitation and formalization	(L2) Suitability and risk screening	(L3) Coordination and responsibility alignment	(L4) Operational artefact support
Herm et al. (2022)	Assumes process descriptions	Prioritization only	Responsibilities under specified.	No concrete artefacts
Farinha et al. (2024)	Acknowledges, but does not support	Strong support	Limited, high-level	Only for process selection
Eulerich et al. (2024)	Assumes process descriptions	Strong evaluation of risks, missing feasibility evaluation	Partial. Missing role allocation.	Control requirements but reliant on interpretation
Cascais Brás et al. (2025)	Acknowledges, but does not support	Conceptual acknowledgement	Strong support for roles and governance	No concrete artefacts

3.4 Cross-framework synthesis

The evaluation of existing process automation frameworks reveals a set of recurring structural patterns that closely mirror the implementation challenges identified in the failure discourse. While the reviewed frameworks differ in scope, level of abstraction, and intended use, they exhibit consistent limitations in how they address the coordination- and knowledge-related conditions under which process automation succeeds or fails in practice. Across frameworks, process explicitation and formalization is predominantly treated as an implicit prerequisite rather than as an organizational challenge to be actively supported. Most frameworks assume that candidate processes are already sufficiently understood, documented, and stable at the point where automation decisions are made. However, chapter 2 demonstrates that weak process understanding, incomplete documentation, and tacit knowledge are common root causes for automation failure. The collective absence of mechanisms to articulate and stabilize process knowledge across actors indicates a systematic blind spot in how frameworks support shared understanding at the boundary between human work and executable logic.

A second recurring pattern concerns pre-automation suitability screening. Although process selection is widely acknowledged as critical, existing frameworks tend to frame suitability primarily as a matter of prioritization among candidate processes, while giving limited attention to its role as a risk-sensitive gatekeeping mechanism. Explicit rejection mechanisms, structured go/no-go decisions, or dynamic reassessment of suitability over time are largely absent. As a result, early coordination around feasibility, data quality, and process stability is weakly supported. In contrast, the literature reviewed in chapter 2 consistently emphasizes these factors as decisive early risks.

With regard to cross-unit coordination and responsibility alignment, most frameworks recognize the involvement of multiple organizational actors and acknowledge governance challenges conceptually. However, responsibility allocation is often articulated at a high level, leaving decision rights, handoffs, and accountability for development, maintenance, and operational ownership under-specified. This reflects a broader tendency to assume that coordination will be resolved through organizational maturity or local adaptation, despite empirical evidence that fragmented ownership and unclear responsibilities are persistent sources of failure in decentralized automation settings.

Finally, the synthesis highlights a pronounced gap in operational artefact support. Even frameworks that are methodologically rich or empirically grounded tend to remain abstract in their prescriptions, offering stages, principles, or control areas without translating them into concrete artefacts that can be enacted in everyday automation work. This places the burden of coordination and interpretation

on local actors, leading to variability in practice and undermining repeatability and comparability across automation initiatives. This finding aligns with the observation in chapter 2 that many failures materialize at the operational level, where abstract guidance fails to support testing, monitoring, maintenance, and adaptation over time.

These patterns indicate that existing frameworks provide valuable conceptual structure, yet systematically under-address the early and operational coordination work through which automation is made feasible, stable, and governable in practice. Rather than a lack of frameworks, the literature exhibits a lack of artefact-level support at precisely those points where empirical evidence shows automation initiatives are most fragile. This synthesis therefore motivates the need for artefacts that explicitly bridge process understanding, suitability assessment, responsibility alignment, and operational enactment within the organizational context of process automation.

3.5 Implications for artefact design

The limitations of existing process automation frameworks do not stem from a lack of conceptual understanding, but from insufficient operational support at critical points of automation practice. Viewed through a coordination and artefact perspective, these limitations reflect weak support for managing interdependencies and establishing shared understanding across organizational boundaries. In response, the artefacts developed in this thesis are intended to complement existing frameworks by addressing the specific gaps that recur across the literature.

The artefacts must explicitly support process explicitation and formalization as an organizational activity. It cannot be assumed that process knowledge is already articulated or shared. Support is needed in making tacit work practices explicit and in articulating rules, inputs, exceptions, and variations at a level of granularity suitable for automation assessment. This positions the artefacts at the boundary between business knowledge and automation logic, where they function as shared reference points for translating human work into evaluable and executable representations.

The artefacts must also enable systematic pre-automation suitability and risk screening. Given that existing frameworks often conflate suitability with prioritization, the artefacts should support early coordination around feasibility by introducing explicit screening logic that allows organizations to reject, defer, or redesign automation candidates before development begins. This includes making feasibility, data quality, process stability, and exception handling visible as decision-relevant factors, and treating suitability as a condition that may evolve over time rather than as a one-off judgment.

The artefacts should further support cross-unit coordination and responsibility alignment in a concrete and enforceable manner. Rather than addressing governance only at a conceptual level, responsibilities, decision rights, and handoffs should be made explicit across business units, automation teams, and IT. This includes clarifying who is responsible for providing process knowledge, assessing feasibility, making automation decisions, and owning automated processes over time, particularly in decentralized or hybrid operating models.

Finally, the artefacts must provide operationally usable outputs that can be embedded into everyday automation work. To mitigate the gap between abstract guidance and practice, the artefacts should materialize coordination into tangible and reusable outputs, such as structured templates, decision aids, or documentation formats that support repeatable and comparable automation decisions. These outputs should minimize the need for local interpretation and integrate with existing routines for development, testing, monitoring, and maintenance.

These implications position the artefacts as socio-technical coordination mechanisms that stabilize shared understanding and structured handoffs in early and operational stages of process automation. By grounding artefact design in the recurring shortcomings of existing frameworks, this thesis aims to provide actionable support at precisely those points where the literature indicates the highest risk of failure.

4 Methodology

4.1 Research approach and methodological positioning

The primary objective of this study is to design artefacts that address coordination challenges in early stages of process automation work. Rather than seeking to explain or predict organizational phenomena, the study adopts a constructive orientation in which empirically identified coordination breakdowns are translated into design-oriented responses. In this sense, the research is concerned with how recurring problems related to process articulation, responsibility alignment, and early-stage structuring can be addressed through artefacts that support collaboration between business units and process automation specialists. For this objective, design science research (DSR) is used as the main methodological frame of the study.

However, this choice is not self-evident. First, the empirical setting and research process bear resemblance to action research. Second, DSR itself has been subject to critique concerning how theory and contribution are framed in the literature. These two considerations highlight the need to clarify the methodological choice explicitly rather than treat DSR as a default.

This study is conducted in a real organizational setting, addresses a practical problem, and relies on close engagement with practitioners in order to understand the problem and shape the solution design. These features align closely with Baskerville's (1999) formulation of action research. Action research is also described as especially relevant for understanding change processes in social settings, which further supports its relevance in the present case. This is also reinforced by the researcher's insider position as a member of the process automation team, which is in line with the organizationally embedded and collaborative problem solving that is identified as characteristic of action research. (Baskerville, 1999.)

Regardless, this study is not positioned primarily as action research. Although it has several action-research resembling features, its main scholarly emphasis is not on carrying out and evaluating a full intervention cycle in the classical action research sense. Baskerville describes action research as a cyclical process involving diagnosing, action planning, action taking, evaluating, and specifying learning (Baskerville, 1999). This study does overlap with diagnosing and action planning, and it is informed by practitioner collaboration. However, its primary contribution is the design of artefacts grounded in empirically instantiated problem understanding, rather than in the cyclical implementation and evaluation of organizational intervention. For this reason, DSR remains the most appropriate primary frame.

Also, DSR literature has been criticized for using the concept of theory too loosely. The critique points out that DSR contributions have at times been framed in overly theory-centric terms to elevate their perceived relevance by labelling them as theory. Therefore, more careful and restrained use of the concept of design theory has been called for. (Iivari, 2020.)

In line with this critique, this study adopts DSR with methodological caution, which shapes how the contributions are presented. This study does not claim to produce design theory in the ambitious sense criticized by Iivari (2020). Instead, DSR is used as a framework for deriving artefacts, design requirements, and design principles from an empirically grounded problem context. This positioning is consistent with the argument that DSR can be understood as producing design knowledge about novel artefacts with practical utility, and that it is often more precise to speak of design knowledge than to overextend the language of theory (Iivari, 2020). The contribution of the study is therefore best understood as practice-oriented design knowledge concerning how early-stage automation work can be more effectively structured.

4.2 Research design and process

The research design follows a structured Design Science Research process in which empirical inquiry and artefact design are clearly separated but systematically connected. This structuring supports grounding the artefact development in validated organizational problems rather than in assumed needs or ad-hoc solution ideas. In line with established DSR guidance, the research process emphasizes problem understanding and instantiation prior to artefact design, thereby strengthening the relevance and rigor of the resulting artefacts (Hevner & Chatterjee, 2010).

The study proceeds through two main empirical phases. In the first phase, challenges related to process automation implementation identified in the literature are instantiated within a specific organizational context through semi-structured interviews. The purpose of this phase is not to discover entirely new problem categories, but to examine how well-documented coordination and artefact-related challenges manifest in practice, how they are experienced by practitioners, and which aspects are particularly salient in the case organization. This phase corresponds to what design science literature describes as problem identification and problem understanding, which forms the foundation for any subsequent design activity (Hevner & Chatterjee, 2010).

The second empirical phase consists of a participatory design workshop, which builds directly on the findings of the interview analysis. Consolidated problem statements derived from the first phase are used as structured input to the workshop. The workshop is designed to translate empirically grounded

coordination breakdowns into explicit design requirements for artefact development. Importantly, the workshop does not aim to generate finished solutions or artefact designs. Instead, it focuses on articulating coordination needs, constraints, and expectations that any proposed artefact should address. This approach aligns with DSR principles that emphasize iterative refinement of problem understanding and solution requirements before committing to concrete design decisions (Hevner & Chatterjee, 2010).

A key design principle of the research process is the deliberate separation between problem instantiation and artefact design. By first validating and contextualizing literature-based challenges through interviews, and only then engaging participants in requirement elicitation, the study avoids premature solution bias. At the same time, the close coupling between the two phases ensures continuity. The workshop is not an independent data collection exercise, but a structured extension of the interview analysis within the same design science cycle. The resulting design requirements emerge from an explicit chain of reasoning linking prior theory, empirical problem instantiation, and participatory refinement, providing a robust foundation for the subsequent artefact development.

4.3 Empirical context and participant selection

The empirical part of the study was conducted in a large multi-industry organization that operates a centralized, in-house process automation team. The team is responsible for developing, deploying, and maintaining process automation solutions, including robotic process automation and intelligent automation, for multiple internal business units such as finance, human resources, and other operational functions. Business units propose processes for automation, while the process automation team supports process analysis, development, testing, deployment, and ongoing maintenance. The researcher works as a developer in this team but remained solely in the researcher role during the empirical work.

This organizational setting provides a suitable context for examining coordination challenges at the intersection of business process knowledge and automation work. The study adopts a single-case research design in order to achieve the contextual depth required for empirically grounded artefact construction within a design science research approach. Rather than aiming for statistical generalization across organizations, the focus is on achieving analytical depth that enables the identification and articulation of coordination challenges as they manifest in everyday practice. The selected case offers rich access to actors directly involved in early-stage process automation work, making it well suited for the purposes of this study (Yin, 2012).

The interview group included all members of the organization's process automation team except the researcher, resulting in a total of seven participants. The group included the process automation platform product owner (who also acts as the team's managerial lead), an analyst/agile master, a lead developer, two internal developers, and two developers employed by an external consultancy who work full-time as integrated members of the team. While job titles provide a general indication of responsibilities, all participants are involved, to varying degrees, in process definition, automation development, testing, and maintenance activities.

Interviewing the full process automation team ensured comprehensive coverage of perspectives within the unit and eliminated intra-team sampling bias. The heterogeneity of roles within the team is a strength of the study, as it enables the examination of coordination challenges across different responsibilities and viewpoints within the same organizational context. Despite differences in formal roles, team members share responsibility for translating business process knowledge into executable automation logic and for sustaining automations over time. This shared involvement creates overlapping yet differentiated perspectives on process understanding, artefact use, and responsibility allocation.

From a coordination perspective, members of the process automation team occupy a boundary-spanning position between business units and automation technologies. They interact closely with business representatives to elicit and formalize process knowledge while simultaneously engaging with technical constraints, platform capabilities, and operational requirements. As such, they are directly exposed to coordination breakdowns related to implicit process knowledge, inadequate artefacts, and unclear expectations across organizational boundaries. Their collective experiences provide a rich empirical basis for instantiating literature-based challenges in the specific organizational setting. The decision to focus on the process automation team as a whole, rather than sampling individual roles or including business unit representatives at this stage, aligns with the research design outlined in chapter 4.2.

The purpose of the interview phase is to instantiate coordination and artefact-related challenges as they manifest in automation work practice, particularly at the point where implicit business processes are translated into formal automation logic. The perspectives of business units are incorporated indirectly through the team's accounts and are addressed more explicitly in the subsequent design workshop through the formulation of artefact design requirements.

4.4 Data collection: semi-structured interviews

The primary empirical data for the study were collected through semi-structured interviews with members of the process automation team. In total, seven interviews were conducted, with one interview per participant. All interviews took place within a single week in February 2026. Conducting the interviews within a short and concentrated time frame reduced the risk of organizational changes influencing participants' accounts and supported the internal consistency of the data. All interviews were conducted remotely. Each interview lasted between 31 and 49 minutes, with an average duration of approximately 40 minutes. The interview length was sufficient to cover all predefined themes while allowing participants to elaborate on their experiences and to introduce relevant issues beyond the interview guide when appropriate. All interviews were conducted in Finnish, which was the native working language of both the interviewer and the participants. Conducting the interviews in the participants' primary language supported nuanced expression and reduced the risk of meaning loss in discussing complex work practices.

The interviews were audio-recorded with the participants' consent. No video data were collected. Audio recordings were transcribed using automated transcription tools, after which the transcripts were manually reviewed and edited by the researcher. The transcription approach can be characterized as near verbatim. Spoken language was transformed into coherent written standard language while preserving the original wording, terminology, and meaning as accurately as possible. Minor adjustments were made to remove filler expressions and to improve readability, without altering the substantive content of participants' statements. This approach ensured a balance between analytical precision and practical usability of the transcripts.

A semi-structured interview guide was used as a common framework across all interviews. The guide was designed to ensure that key themes derived from the literature were addressed consistently while allowing flexibility for follow-up questions and participant-led elaboration. After the first interview, a small number of questions were removed from the guide based on initial observations regarding redundancy and relevance. No role-specific interview guides were created. However, the interviewer applied mild role-sensitive emphasis when appropriate, for example by probing more deeply into strategic or coordination-related issues with participants holding managerial or coordination-oriented roles. Overall, the interview guide was followed systematically, but deviations were allowed whenever participants raised issues deemed relevant to the research focus. The interviews were explicitly framed as a problem-focused inquiry. At the beginning of each interview, participants were informed that the purpose of the study was to identify and understand recurring challenges in process

automation work rather than to evaluate individual performance, specific projects, or to propose solutions. This framing was intended to encourage open reflection on difficulties and limitations in current practices without fear of personal or organizational judgment.

Participation in the interviews was voluntary, and all participants provided informed verbal consent prior to the interviews. Participants were informed about the anonymization of the data and the research setting. The organization and all individuals are anonymized in the thesis. The selection of participants was not influenced by managerial decision-making as all members of the team were included. The researcher's insider position provided familiarity with the organizational context while requiring particular attention to reflexivity during data analysis. This was addressed by focusing the interviews on concrete practices and experiences and by grounding the analysis explicitly in theory-derived constructs rather than personal assumptions.

4.5 Interview guide development

The interview guide was developed deductively based on the literature review presented in Chapter 2. Its purpose was to operationalize coordination- and artefact-related challenges identified in prior research into empirically observable themes that could be examined in organizational practice. Rather than eliciting solution ideas or evaluating existing automation initiatives, the interview guide was explicitly designed to support problem instantiation by focusing on current practices, recurring difficulties, and points of friction in process automation work.

The guide consisted of ten thematic sections: (1) interviewee background and role in process automation initiatives, (2) entry points of automation initiatives, (3) process understanding, (4) artefacts in use, (5) pre-automation suitability assessment, (6) responsibilities and expectations, (7) testing and late-stage process clarification, (8) informal practices and workarounds, (9) summary and reflection, and (10) closing. Together, these themes covered the full early lifecycle of process automation work, from initial contact with business units to testing and stabilization, while maintaining a clear focus on coordination and shared understanding.

Questions related to process understanding and documentation operationalized challenges associated with tacit and fragmented process knowledge. Themes focusing on artefacts addressed the role of boundary objects in mediating collaboration across organizational boundaries. Questions concerning suitability assessment and responsibilities reflected coordination breakdowns associated with premature automation decisions and unclear ownership. The inclusion of testing and late-stage

clarification as a distinct theme enabled examination of how early-stage ambiguities manifest downstream in development and validation activities.

Most interview questions were formulated as open-ended prompts (e.g., “Describe…”, “How…”, “When…”), encouraging participants to articulate their experiences in their own terms and to provide concrete examples from practice. The guide included a small number of reflective questions toward the end of each interview, inviting participants to synthesize their experiences and to identify recurring sources of difficulty across automation initiatives. No questions explicitly asked participants to propose solutions or improvements, ensuring that the interviews remained focused on problem manifestation rather than design ideation.

The interview guide was refined iteratively during the data collection phase. After the first interview, a small number of questions were removed because they did not produce analytically relevant insights. These questions primarily concerned descriptions of current practices that yielded self-evident or repetitive observations and did not contribute meaningfully to the instantiation of literature-based challenges. This refinement did not alter the overall thematic structure of the guide but improved its focus and analytical efficiency.

No role-specific versions of the guide were created. The same thematic structure was applied across all interviews, supporting comparability of responses. At the same time, the semi-structured format allowed the interviewer to adjust emphasis and pose follow-up questions based on the participant’s role and experience.

The finalized interview guide is included in Appendix 1 as the original Finnish language version and in Appendix 2 as the translated English version.. Its development and use aimed to foster systematic reflection of the theoretical framing of the study while remaining grounded in participants’ lived experiences of process automation work. As such, the guide functioned as a key methodological bridge between the literature-based problem framing and the empirical analysis presented in chapter 5.

4.6 Data analysis

The interview data was analyzed using a theory-informed qualitative analysis approach. The purpose of the analysis was not to inductively generate new theoretical constructs, but to instantiate and contextualize coordination- and artefact-related challenges identified in the literature within the case organization. The analysis thus serves as a bridge between the literature-based problem framing and

the subsequent design workshop, providing an empirically grounded basis for artefact requirement elicitation.

Prior to coding, an analytical framework was established based on the key challenges synthesized in chapter 2. These challenges relate to process explicitation and shared understanding, pre-automation suitability assessment, artefacts as coordination mechanisms, responsibility allocation, late discovery of process-related issues, and the use of informal practices and workarounds. Rather than functioning as predefined coding categories, these challenges were used as sensitizing concepts that guided analytical attention toward coordination- and artefact-related phenomena without prescribing how empirical observations should be categorized. This framework defined the analytical scope of the study and ensured that the analysis remained focused on known problem areas relevant to early-stage process automation work, while allowing empirical patterns to emerge inductively from the data.

The analysis proceeded in multiple stages. In the first stage, all interview transcripts were read in full to gain an overall understanding of participants' accounts and to familiarize the researcher with recurring patterns across interviews.

In the second stage, the transcripts were coded using the predefined analytical categories. Coding focused on identifying concrete instances where participants described difficulties, breakdowns, or inefficiencies that corresponded to the literature-based challenges. At this stage, coding emphasized how challenges manifested in practice rather than merely that they existed. Codes captured specific forms of problem manifestation, such as late emergence of process exceptions, reliance on undocumented assumptions, or ad-hoc interpretation of business rules by automation team members.

In the third stage, the initial codes were examined and refined through comparison across interviews. Similar codes were grouped together to identify recurring patterns and to abstract from individual experiences to organization-level observations. This stage resulted in a set of second-order themes that describe stable and recurring ways in which coordination challenges materialize in the case organization. These themes represent the organizational instantiation of the abstract challenges discussed in the literature.

In the final stage of the analysis, the identified themes were interpreted in relation to their consequences for process automation work and their relevance for artefact design. For each theme, attention was given to the organizational conditions under which the challenge arises and to the practical implications it has for development effort, collaboration, and project outcomes.

Based on this interpretation, the analysis produced a set of consolidated problem statements that articulate the core coordination challenges observed in the case organization. These problem statements synthesize empirical observations across interviews and are formulated at a level of abstraction suitable for guiding subsequent design activities. The analysis deliberately avoids proposing solutions or design features. Instead, it focuses on making coordination breakdowns explicit and analytically grounded. The consolidated problem statements derived from the analysis serve as the primary input to the design workshop described in Section 4.7, where they are refined and translated into explicit artefact design requirements. In this way, the analysis supports a clear separation between problem instantiation and solution design while maintaining continuity between empirical findings and artefact development.

4.7 Design workshop and artefact requirement elicitation

Following the completion of the interview analysis, a participatory design workshop was conducted with the members of the process automation team. The purpose of the workshop was to transform the empirically instantiated coordination challenges into operational design requirements for artefact development. In line with the design science research logic adopted in this study, the workshop functioned as a deliberate transition from problem instantiation to solution space exploration.

Prior to the workshop, the results of the interview analysis were consolidated into a set of problem statements representing recurring coordination breakdowns observed across interviews. These statements were not presented as abstract theoretical categories, but as operationalized descriptions of practical situations in which insufficient process understanding, unclear responsibility allocation, or structural weaknesses had materialized. The operationalization of analytical categories into discussion prompts was an intentional methodological step. Rather than asking participants to agree with abstract characterizations, the workshop prompts focused on identifying specific lifecycle stages where breakdowns become visible, concrete triggers revealing insufficient process articulation, points at which responsibility boundaries become blurred, and situations where process uncertainty leads to rework or late-stage changes. This framing was designed to anchor discussion in observable coordination mechanisms rather than general perceptions.

The workshop lasted approximately two hours and was conducted in a meeting room at the office of the partner company. All workshop participants had previously taken part in the interviews, ensuring familiarity with the overall research topic while enabling collective reflection on synthesized findings. The facilitation followed a structured yet dialogical format. The session was divided into three

sequential phases: Phase 1, validation and refinement of problem statements. Participants were invited to confirm, nuance, or challenge the consolidated problem descriptions. This phase served to ensure collective recognition of the issues and to prevent the researcher's interpretation from remaining unexamined. Phase 2, identification of coordination breakdown mechanisms. The discussion then focused on explicating how and when coordination failures typically emerge in automation initiatives. Rather than generating solutions, participants were asked to describe thresholds, implicit assumptions, responsibility transitions, and decision practices that characterize current operations. Phase 3, articulation of design-relevant structural conditions. In the final phase, participants were guided to reflect on what structural conditions would need to be in place to prevent or mitigate the previously identified breakdowns. The facilitation explicitly avoided asking for concrete artefact ideas. Instead, the emphasis was on specifying necessary conditions, decision points, information requirements, and accountability structures.

Throughout the workshop, attention was deliberately directed toward structural and procedural factors rather than individual performance. This framing reduced the risk of defensive reactions and maintained the focus on coordination design rather than personal critique. The workshop discussion was documented through structured notes capturing refinements to the problem statements, descriptions of recurring breakdown mechanisms and articulated structural conditions for improved coordination. No voting or prioritization mechanisms were used. Instead, shared understanding emerged through iterative clarification during discussion.

After the workshop, the documented material was analyzed by the researcher. The articulated structural conditions were synthesized into a set of ten operational design requirements. These requirements were formulated at a level of abstraction that specifies what any artefact must make explicit, enforce, enable, or clarify in order to address the empirically observed coordination challenges. Subsequently, the operational design requirements were further abstracted into four design principles. This abstraction step aimed to identify higher-level normative design orientations underlying multiple individual requirements. The design principles serve as conceptual bridges between empirically grounded coordination problems and the concrete artefacts developed in the following phase of the study.

The transition from workshop discussion to design requirements and further to design principles was conducted iteratively, fostering traceability between interview findings, workshop articulations, operational design requirements, and abstract design principles. This traceability supports methodological rigor by demonstrating how artefact design is systematically grounded in empirical

evidence rather than researcher intuition. The workshop did not result in finished artefact designs. Instead, it produced structured design knowledge that constrains and informs the subsequent artefact construction phase, described in the following chapter.

4.8 Ethical considerations and research rigor

Ethical considerations were taken into account throughout the study. Participation in both the interviews and the design workshop was voluntary, and all participants were informed orally about the purpose of the study, the research setting, the use of the collected data, and the anonymous reporting of the findings before the interviews were conducted. Verbal consent was obtained from all participants. They were also informed that they could withdraw from the study if they wished. No separate formal ethical review was conducted for the study. However, the study did not involve minors, vulnerable groups, or topics that would be expected to cause significant harm to participants. Still, care was taken to ensure that participation would not create unnecessary personal or professional risks.

The empirical material of the study consisted of interview recordings, interview transcripts, and workshop documentation. All material was stored in the partner company's cloud environments. The organization and the participants were anonymized in the thesis. In addition, role identifiers were removed from direct quotations to reduce the risk of indirect identification. This was important because the study focused on a single team with a relatively small number of members, which could otherwise make participants identifiable through their role. During both the interviews and the workshop, the study was presented as an effort to understand structural and coordination-related challenges in process automation work, rather than to evaluate individual employees or their performance. This was considered important for creating a setting where participants could speak openly.

The rigor of the study was supported by several choices made in the research design. First, the study covered the process automation team in full. This made the empirical material more comprehensive and reduced the risk that important viewpoints within the team would be left out. Second, the interview guide was developed based on the literature review, which supported alignment between the theoretical background, the empirical data collection, and the later analysis. At the same time, the semi-structured format made it possible for participants to describe their own experiences and to raise issues that were especially relevant in their daily work. Third, all interviews were conducted within a short period of time to support consistency across the interviews. Fourth, the interviews were conducted in the native language of the participants to make it easier to discuss detailed work practices

in a natural way. In addition, the interviews were recorded, transcribed automatically, and then checked manually, which strengthened the reliability of the material.

The researcher was also a member of the focal team, which needs to be acknowledged when assessing the study. This insider position supported access to the case context and helped in understanding the practical details of the work. At the same time, it also created a risk that some assumptions could remain too implicit or that some issues could appear more self-evident than they actually were. This risk was addressed by focusing the interviews on concrete experiences and work practices, by grounding the analysis in concepts derived from the literature, and by describing the progression from empirical observations to consolidated problem statements in a transparent way. In addition, the design workshop was not only used for eliciting design requirements, but also for refining and challenging the researcher's interpretations. In this sense, the workshop also functioned as a form of participant validation.

The credibility of the study is strengthened by the close connection between the literature review, the interview study, the analysis process, and the workshop-based refinement of the findings. In addition, participants, data collection, and the analysis process are described in sufficient detail to showcase how the resulting design requirements and design principles were derived from the empirical material. Overall, these choices strengthen the trustworthiness of the study and make the basis of the artefact design transparent.

5 Empirical findings

5.1 Instantiated coordination challenges

This subchapter presents the empirical findings derived from the semi-structured interviews conducted with members of the process automation team. The purpose of the analysis was not to generate new theory, but to instantiate and contextualize coordination- and artefact-related challenges identified in the literature within the case organization. Therefore, the findings describe how abstract coordination challenges materialize in concrete automation work practices and how they manifest in the early stages of process automation initiatives.

The analysis resulted in the identification of three recurring and interrelated second-order themes that characterize coordination breakdowns in the studied context. The most central of these concerns is the lack of explicit and shared process understanding prior to automation development. Interview data consistently indicate that process knowledge is at times tacit, fragmented, and unevenly distributed across organizational actors. While such characteristics are typical in complex organizational processes, challenges arise when this knowledge is not consolidated into a sufficiently explicit and shared representation before development activities commence. As a result, defining the process becomes iterative and continues throughout automation development, testing, and even production use, complicating feasibility assessment and increasing rework.

The second theme concerns ambiguous ownership and responsibility at the boundary between business units and the process automation team. In particular, responsibilities in articulating process definitions and constructing comprehensive test cases often remain unclear. The translation of business reality into automation-ready logic requires joint contributions from both sides, yet this boundary work lacks clearly defined ownership structures. This ambiguity adds to the instability of process understanding and creates governance gaps in early-stage automation work.

The third theme highlights process articulation as a socially sensitive interaction. Although less frequently articulated than the preceding challenges, the interviews reveal that the elicitation and clarification of process knowledge are not purely technical exercises. Repeated questioning, probing for exceptions, and requests for formalization may be experienced as burdensome, frustrating, or even undermining professional identity. These social dynamics influence how openly and thoroughly process knowledge is shared, thereby shaping the quality of process explicitation. These three themes describe a coordination-intensive problem space in which early-stage process automation work is constrained less by technical limitations than by difficulties in stabilizing shared process knowledge

across organizational boundaries. The following sections examine each theme in detail, analyzing how these coordination challenges materialize in practice and articulating consolidated problem statements that serve as the foundation for subsequent design requirement elicitation.

5.1.1 Lack of explicit and shared process understanding

The most central coordination challenge identified in the interview data concerns the difficulties establishing an explicit, sufficiently detailed and shared understanding of the business process prior to automation development. While tacit knowledge and distributed expertise are inherent characteristics of complex organizational processes, the findings indicate that these characteristics become problematic in the context of process automation, where human work must be translated into executable logic. In large organizations, process knowledge is rarely centralized or exhaustively documented. Instead, it is distributed across individuals, roles, and units, and embedded in routines, exceptions, and experiential judgment. From this perspective, the presence of tacit and fragmented knowledge is not itself a dysfunction.

However, the automation context imposes additional constraints. To implement a process as software, the underlying logic, rules, inputs, and relevant variations must be articulated in a form that enables unambiguous interpretation. The empirical findings suggest that this consolidation and stabilization of process knowledge often remains incomplete at the point when development work begins. The weight of this is illustrated, among others, by the following comments:

> *"Automation project problems most often stem from the failure to properly describe the process that the automation is supposed to implement. The process description often contains gaps and misunderstandings."*

> *"The most common problem is insufficient or incomplete specifications."*

A recurring pattern across interviews was that process definition did not precede development in a stable and consolidated form. Instead, process understanding evolved iteratively alongside development and testing activities. Initial descriptions provided by business units were frequently high-level, partial, or dependent on implicit assumptions. As developers began to formalize the process into automation logic, ambiguities and gaps surfaced, prompting additional rounds of clarification. This iterative clarification was not limited to early development stages. In several accounts, interviewees described how new interpretations, exceptions, or previously unarticulated variations emerged during testing or even after deployment into production. In effect, the specification of the process continued throughout the automation lifecycle, rather than being largely

stabilized beforehand. The result was a recurring cycle of definition, development, redefinition and adjustment, in which process articulation and technical implementation became tightly intertwined. These findings were illustrated by, for example, the following comments:

> *"Quite often there are still uncertainties when development begins. The process often becomes more refined only during development, and there are usually many gaps in the specifications even though development has already started."*

> *“Exceptions almost always emerge only during development, when a specific case is encountered in the course of the development work.”*

A critical consequence of this pattern is the delayed visibility of process maturity. When a sufficiently explicit and shared understanding is not achieved, it becomes difficult to assess early on whether a candidate process is stable, standardized, and suitable for automation. Only during implementation does the full extent of variability, ambiguity, or instability become visible.

At that point, development resources have already been committed, and structural weaknesses in the underlying process may only be recognized after significant effort has been invested. The interviews also revealed that continuous reliance on meetings, ad hoc communication, and individual memory often compensates for the absence of stable artefacts. Rather than functioning as shared reference points, process descriptions frequently remain provisional and dependent on ongoing dialogue. While such communication may temporarily bridge knowledge gaps, it increases cognitive load and makes coordination vulnerable to misunderstandings and personnel changes. In this sense, coordination is achieved, but at the cost of increased fragility and reduced repeatability.

The instability of shared process understanding has several concrete implications for automation work. First, it increases rework. As new exceptions or rules are discovered late, previously implemented logic must be revised. Second, it complicates testing. When relevant variations are not fully identified at the outset, test cases may fail to capture critical scenarios, leading to defects surfacing in later stages. Third, it obscures early suitability screening. Without a sufficiently consolidated process representation, the automation team cannot reliably evaluate whether the process is mature, rule-based, and stable enough to justify automation. This is illustrated in, for example, the following response:

> *“Very often the true nature of the process only becomes clear during development work or even after it has been completed.”*

The findings do not suggest that perfect process explicitation is achievable or even desirable. Complete elimination of tacit knowledge would be unrealistic and potentially counterproductive. However, the data indicate that the threshold for "sufficient" explicitation in automation contexts is frequently not met before development begins. The absence of mechanisms to consolidate fragmented knowledge into a shared and explicit representation leads to a structural misalignment between the requirements of automation and the state of process articulation.

5.1.2 Ambiguous ownership and responsibility

The instability of shared process understanding is compounded by a second, closely related coordination challenge, ambiguity in ownership and responsibilities at the boundary between business units and the process automation team. Both parties are jointly involved in early-stage automation work, but the interviews indicate that responsibility for critical coordination tasks, particularly process definition and test case construction, remains structurally under-specified.

Process automation initiatives span multiple domains of expertise. Business units possess domain-specific knowledge of how work is performed, including contextual judgment, variations, and operational constraints. The process automation team, in turn, holds expertise in formalization, technical feasibility, and the requirements of automation platforms. The translation of business reality into automation-ready representations thus requires joint contributions. However, the findings suggest that this boundary work is rarely governed by clearly defined ownership structures. Instead, it occupies a grey area between established responsibilities.

One manifestation of this ambiguity concerns the ownership of process definition. Business units are typically accountable for the underlying business process and its outcomes while the automation team is responsible for implementing the automation solution. Yet the intermediate step of articulating the process in a form that is both in line with the business reality and sufficiently precise for technical implementation does not fall entirely within either domain.

Interview accounts describe situations in which business representatives provide descriptions that reflect their operational understanding but lack the granularity or formal structure required for automation. On the other hand, automation specialists may identify logical gaps or inconsistencies but lack the authority or contextual knowledge to definitively resolve them. As a result, the responsibility for producing a consolidated and automation-ready process specification becomes diffuse as illustrated by the following remark in an interview:

> *"In general, responsibility for these matters should lie with the business, but unfortunately the business often does not have sufficient knowledge of the process or the willingness to take responsibility for advancing the project, and in practice the responsibilities shift to the process automation team."*

This diffusion of responsibility can lead to several outcomes. In some cases, the automation team assumes a disproportionate role in interpreting and structuring business knowledge, effectively performing analytical work that extends beyond technical implementation. In other cases, gaps remain unresolved because each side implicitly expects the other to clarify them. The absence of explicit ownership over the translation process reinforces the instability described in section 5.1.1, as process articulation becomes contingent on informal negotiation rather than structured governance.

A similar pattern emerges in relation to test case creation. Comprehensive testing requires identification of relevant process variations and edge cases. Business units typically possess detailed knowledge of the different scenarios that may occur in practice. However, not all variations are equally relevant from an automation perspective. Some differences are semantically meaningful for human actors but irrelevant for the execution logic of the automation, while others fundamentally alter the required system behaviour. The construction of adequate test cases therefore depends on the integration of contextual knowledge of possible variations and technical knowledge of which variations are consequential for automation logic. The interviews indicate that responsibility for integrating these perspectives is often unclear:

> *"In black-and-white terms, it is the responsibility of the business to understand the process that is being automated. They should understand the target systems and be able to provide test data ... It often feels like the responsibility ends up with the developer. Especially obtaining test data or accessing test systems frequently becomes the developer's responsibility."*

Business representatives may assume that the automation team will determine what needs to be tested, while the automation team relies on business actors to enumerate relevant scenarios. Without explicit coordination mechanisms, critical cases may be overlooked or identified only at later stages.

> *"In practice, test cases or requirements are usually not defined in advance. I have never once received meaningful test data in a situation where a clear initial state and expected end state were provided."*

This asymmetry in knowledge and responsibility creates uncertainty around early-stage testing. The lack of clarity about who is accountable for defining, validating, and approving test cases increases the likelihood of incomplete coverage and late-stage defect discovery. Moreover, when issues surface during testing or production, responsibility for the oversight may become contested, further complicating collaboration.

The ambiguity in ownership and responsibility also has broader governance implications. When critical coordination tasks at the boundary between business and automation remain under-specified, accountability becomes diffuse. Decision rights regarding process readiness, feasibility, and go/no-go judgments may be exercised informally rather than through structured mechanisms. This increases reliance on personal initiative and informal escalation paths, reducing transparency and repeatability across automation initiatives.

The findings do not suggest that formal role definitions are entirely absent. Instead, the ambiguity is about the coordination work between teams that doesn't clearly fit within existing roles or functional boundaries. Process definition and test case construction are neither purely business tasks nor purely technical tasks. Without explicit recognition of this boundary work as a distinct responsibility domain, coordination remains dependent on ad hoc arrangements and individual competence.

5.1.3 Process articulation as a socially sensitive interaction

In addition to structural issues related to shared understanding and responsibility, the interviews reveal a third, more subtle coordination challenge. Process articulation is not a neutral technical exercise but a socially sensitive interaction. Although this theme appeared less frequently than the preceding two, it provides an important lens for understanding why process explicitation may remain incomplete even when both parties recognize its importance.

The articulation of process knowledge typically unfolds through iterative questioning, clarification, and probing. Automation analysts seek to uncover rules, exceptions, decision criteria, and implicit assumptions in order to translate business practice into formal logic. In practice, this often involves repeated requests for clarification and detailed follow-up questions. Several interviewees described this elicitation process as cognitively demanding and socially delicate. Mapping an unfamiliar process through questioning was likened to "probing in the dark," where the analyst does not yet know which aspects are critical and must explore through trial and error:

> *"The information often has to be 'milked' from the business unit, and that is frequently a challenge. We may not necessarily know how to ask everything. ... Sometimes no matter how much we ask, we still do not receive all the essential information."*

> *"The business may consider many things to be self-evident, and when the topic is new to you, you can initially feel quite lost. That's why you have to dare to ask questions that may feel stupid and dig deeper to clarify things."*

The elicitation process is characterized by asymmetry in knowledge and expertise. Business specialists possess deep domain knowledge and contextual experience. Automation analysts possess expertise in formalization and technical implementation but may lack familiarity with the specific business domain. As a result, the same issue may need to be revisited multiple times from different angles before it is sufficiently understood for implementation.

Interviewees described how this dynamic can generate social friction. From the perspective of the automation specialist, repeated questioning may evoke feelings of inadequacy or incompetence, particularly when domain-specific terminology or contextual nuances remain unclear. From the perspective of the business expert, the need to repeatedly explain aspects of what is perceived as routine or self-evident work may cause frustration. What is intended as analytical clarification may be experienced as an indication that the other party "does not understand" or is "not listening.". Moreover, domain expertise is often closely tied to professional identity. Mastery of one's core process is frequently seen as a marker of competence and authority. When the automation team challenges descriptions, requests more precision, or exposes inconsistencies, these interventions may be interpreted not merely as technical requirements but as implicit criticism of the business unit's expertise. In extreme cases, the act of questioning can be perceived as questioning the competence of the expert.

> *"It can feel uncomfortable for us to question them extensively and in relation to every single process."*

> *"From the business perspective, this may feel uncomfortable, as they may feel that we are questioning them or their competence, even though in reality we are simply trying to gain a sufficient understanding of the process so that we can build the automation correctly."*

These social dynamics have practical consequences for process articulation. If clarification is experienced as burdensome, repetitive, or identity-threatening, participants may provide abbreviated explanations or avoid deeper exploration of ambiguities. Instead of collaboratively refining process representations, interactions may shift toward minimizing friction. This can lead to superficial agreement without fully resolved understanding. The interviews suggest that such dynamics are rarely explicit conflicts but rather manifest as subtle tension, fatigue, or reluctance. This is illustrated, for example, by the following remark during one of the interviews:

> *"They (business unit personnel) do not always like it when we ask what happens before a certain step, why something is done in a particular way, or why, for example, a certain Excel file is used. They find it uncomfortable, even though the intention is not to question their way of working, but to understand the background so that we may be able to implement the automation in a different way if necessary."*

Over time, this may constrain the depth of elicitation. Complex exceptions may remain underexplored, ambiguous practices may be left implicit, and unresolved uncertainties may be carried forward into development. In this sense, the social fragility of articulation directly influences the technical robustness of the resulting automation.

The findings do not imply that process articulation is inherently adversarial. In many cases, collaboration between business and automation specialists was described as constructive and mutually supportive. However, the data indicate that articulation requires sustained effort, psychological safety, and trust. Without these conditions, the cognitive and emotional burden associated with repeated probing may limit openness and thoroughness. Emotional factors arose in, for example, the following interview remark:

> *"It is difficult to move development work forward if the developer becomes frustrated and constantly has to send questions via Teams or email and clarify matters that the business unit should have provided upfront."*

5.1.4 The interrelated nature of the coordination challenges

The three coordination challenges identified above do not represent isolated phenomena. Rather, they form an interrelated configuration that characterizes early-stage process automation work in the case organization. Together, they describe a reinforcing pattern in which difficulties in stabilizing shared process understanding are compounded by ambiguous ownership structures and socially sensitive interaction dynamics.

At the core of this configuration lies the instability of shared process understanding. When tacit and fragmented knowledge is not consolidated into a sufficiently explicit and shared representation prior to development, process definition becomes iterative and continuously evolving.

However, this instability is not merely a cognitive or informational issue. It is structurally reinforced by the absence of clearly defined ownership for the translation of business knowledge into automation-ready logic. When responsibility for this boundary work remains ambiguous, no single actor is accountable for ensuring that process representations reach a sufficient level of clarity and completeness before development proceeds. As a result, ambiguities persist and are carried forward into later lifecycle stages.

Simultaneously, the attempt to stabilize shared understanding is shaped by the social dynamics of articulation. Repeated probing, clarification, and reformulation are necessary for formalization, yet these activities may generate frustration, defensiveness, or fatigue. The social sensitivity of process articulation constrains how deeply ambiguities are explored and how openly uncertainties are acknowledged. In contexts where articulation is experienced as burdensome or identity-threatening, participants may converge prematurely on seemingly sufficient descriptions. This, in turn, reinforces the structural instability of process understanding identified in Section 5.1.1.

The three themes thus interact in a mutually reinforcing manner. Unstable shared understanding increases the need for boundary-spanning articulation work. Ambiguous ownership weakens governance over that work. Social fragility constrains its effectiveness. The combined effect is that early-stage automation initiatives operate in an environment where coordination is achieved primarily through ongoing communication and local negotiation rather than through stabilized artefacts and clearly defined responsibility structures.

5.2 Design requirements

Building on the coordination challenges instantiated in section 5.1, this subchapter presents the operational design requirements elicited through the participatory workshop described in section 4.7. The design requirements are displayed in Table 4. The most central cluster of design requirements addresses the lack of explicit and shared process understanding identified in section 5.1.1. The design requirements in this cluster support explicit articulation, uncertainty identification, and systematic suitability evaluation in the project phases preceding development.

Table 4. Design requirements

No.	Design requirement description
1	The artefacts need to make explicit to what level the process is understood and pinpoint known uncertainties before development starts.
2	The artefacts must establish what is the scope of acceptable refinement of the process during development and what is not.
3	The artefacts must provide an explicit structure for evaluating the maturity and other suitability aspects of the process before development begins.
4	The artefacts must create a structured opportunity to assess the overall coherence and optimization of the end-to-end process before committing to automation.
5	The artefacts need to elicit a comprehensive high-level description of the process from the business unit during the pre-definition phase before the joint automation specification meetings are held.
6	The artefacts need to make explicit the boundaries of responsibilities between the business unit and the process automation team.
7	The artefacts need to explicitly identify the business process experts needed for the initiative and their areas of expertise regarding the process.
8	The artefacts must explicitly name people accountable for the initiative.
9	The artefacts must require formal acknowledgement and approval of the documented process description by the designated business representatives before development begins.
10	The artefacts need to support a simplified pipeline for low-risk and straightforward automation cases.

Design requirement 1: The artefacts need to make explicit to what level the process is understood and pinpoint known uncertainties before development starts. This requirement makes the artefacts adapt to the reality of incomplete process understanding. Evident by the interviews and workshop discussions, exhaustive process documentation is not a reasonable assumption in complex processes. Artefacts must therefore be able to explicitly indicate the degree of confidence in different parts of the process specification. Known ambiguities, open questions, and unresolved assumptions must be made visible rather than implicitly carried forward. This creates a shared basis for assessing the process specification and automation readiness.

Design requirement 2: The artefacts must establish what is the scope of acceptable refinement of the process during development and what is not. The interviews and the workshop established that some degree of process definition refinement during development is unavoidable and thus should not be

categorically pathologized. Instead, this design requirement introduces an explicit distinction between what lies within the scope of acceptable iteration and what fundamentally changes the scope or core logic of the process. Without this distinction, acceptable iterative clarifications risks masking deeper process instability and definition failures as normal workflow.

Design requirement 3: The artefacts must provide an explicit structure for evaluating the maturity and other suitability aspects of the process before development begins. The workshop revealed that automation initiatives handed to the process automation team are not always built on stable and mature processes. Artefacts must therefore provide a structured mechanism to examine suitability criteria such as process maturity, stability, data quality and exception handling.

Design requirement 4: The artefacts must create a structured opportunity to assess the overall coherence and optimization of the end-to-end process before committing to automation. Interviews and workshop discussions emphasized that task automation may obscure deficiencies in the broader end-to-end process. Process automation provides a natural point to reflect and critically examine the process in its entirety. Therefore, artefacts must require explicit reflection on whether the broader process is coherent and sufficiently optimized to justify automation.

Design requirement 5: The artefacts need to elicit a comprehensive high-level description of the process from the business unit during the pre-definition phase before the joint automation specification meetings are held. This requirement responds to two empirically identified shortcomings that obstruct the effectiveness of the joint process specification sessions. Demanding a comprehensive high-level description requires the business unit to start articulating the automatable process prior to the joint meetings. This is expected to improve business unit readiness at the outset of joint process specification. Also, the existence of the high-level description is expected to improve the early process understanding of the process automation team analysts and developers. This is expected to improve their readiness to ask the right questions during the joint specification sessions.

The second cluster of requirements addresses the ambiguity in ownership and responsibility identified in Section 5.1.2. The translation of business knowledge into automation-ready representations occupies a boundary space between the business units and supporting operations. This cluster of requirements aims to ensure that artefacts explicitly structure governance at that boundary.

Design requirement 6: The artefacts need to make explicit the boundaries of responsibilities between the business unit and the process automation team. This requirement simply and directly responds to the diffusion of responsibility observed especially in process definition and testing activities.

Artefacts must clarify which tasks belong to the business unit, which to the process automation team and which require joint action. This articulation reduces reliance on implicit expectations and individual initiative.

Design requirement 7: The artefacts need to explicitly identify the business process experts needed for the initiative and their areas of expertise regarding the process. Effective process articulation depends on access to relevant domain expertise. The interviews revealed that the necessary domain experts are not always identified in a timely manner, which can lead to shortcomings in process specification and its validation. Explicitly identifying the process experts helps to ensure that the correct people are involved in the specification of the process and the validation of the automation.

Design requirement 8: The artefacts must explicitly name people accountable for the initiative. In addition to functional roles like analyst and developer, the automation initiative needs explicit responsible individuals. This strengthens accountability and reduces the risk of critical coordination responsibilities remaining unclaimed.

Design requirement 9: The artefacts must require formal acknowledgement and approval of the documented process description by the designated business representatives before development begins. Formal acknowledgement creates a governance checkpoint which shall not be passed until the process specification is deemed complete and accurate. Ensuring formal acknowledgement reduces the likelihood of knowingly passing ambiguity onwards to the development phase.

While structured explicitation and governance are necessary, the workshop also emphasized that not all automation initiatives carry equal risk or complexity. Overly rigid procedures imposed on low-risk and straightforward automation initiatives can become unnecessary overhead.

Design requirement 10: The artefacts need to support a simplified pipeline for low-risk and straightforward automation cases. This requirement responds to the need for dynamic procedure calibration. Artefacts must allow differentiation between complex, high-risk processes requiring full procedural rigor and simpler routine-like cases where a streamlined pathway is appropriate. Without this flexibility and responsiveness, governance mechanisms risk becoming discouraging barriers that risk being ultimately rejected by the organization.

5.3 Design principles

While the design requirements are empirically grounded and practically actionable, they remain closely tied to the specific organizational context in which they were elicited from. To elevate the

contribution beyond context-bound prescriptions, this section abstracts the operational design requirements into a set of higher-level design principles. The purpose of this abstraction is not to generalize specific artefact features, but to articulate the underlying normative orientations that should guide the structuring of automation initiatives in coordination-intensive settings. Whereas design requirements describe what must be implemented in this context, design principles specify how automation initiatives should be structured in order to systematically reduce coordination fragility in early-stage process automation work. The design principles are displayed in Table 5.

Table 5. Design principles

No.	Design principle description
1	The artefacts need to make explicit to what level the process is understood and pinpoint known uncertainties before development starts.
2	The artefacts must establish what is the scope of acceptable refinement of the process during development and what is not.
3	The artefacts must provide an explicit structure for evaluating the maturity and other suitability aspects of the process before development begins.
4	The artefacts must create a structured opportunity to assess the overall coherence and optimization of the end-to-end process before committing to automation.

The abstraction from requirements to principles follows a structured logic. Each design principle synthesizes multiple related requirements into a coherent stance that addresses a distinct dimension of the coordination problem space. The principles do not prescribe specific lifecycle models, artefacts, or organizational forms. Rather, they define structural constraints and evaluative criteria that any artefact ecosystem must embody if it is to prevent the recurring breakdown patterns observed in the empirical material. In this sense, the design principles represent the primary theoretical contribution of this study. They articulate transferable design knowledge concerning how early-stage automation work should be organized when process knowledge is tacit and distributed, responsibilities span organizational boundaries, and automation requires deterministic formalization of human work. The artefacts presented in chapter 6 instantiate these principles within the case organization, but the principles themselves operate at a level of abstraction intended to inform automation governance and artefact design beyond the studied context.

Design principle 1: Automation initiatives must be grounded in an explicitly articulated and critically assessable process representation that stabilizes shared understanding and makes both uncertainty and acceptable scope of change visible prior to development. This principle synthesizes the requirements related to process explicitation and suitability assessment (design requirements 1-4).

The empirical findings demonstrate that unstable shared process understanding frequently undermines early-stage automation work. To address this, artefacts must not merely document process steps but structure the process as an explicit object of evaluation.

From a coordination perspective, the principle supports the stabilization of shared understanding by transforming implicit assumptions into collectively assessable representations. It aligns with the view of artefacts as boundary objects that support coordination across heterogeneous expertise domains. By providing a shared evaluative structure while allowing role-specific interpretation, the process specification functions as such a boundary object. For this stabilizing function to materialize, the process specification must meet several conditions.

First, it must articulate the process at a level of formal precision that enables developers to derive unambiguous execution logic suitable for implementation.

Second, it must remain intelligible and assessable to domain experts, enabling them to evaluate whether the representation accurately reflects operational reality. Without this dual intelligibility, the artefact risks becoming either technically precise but detached from practice, or operationally meaningful but technically underspecified.

Third, the specification must make explicit both the limits of current knowledge and the uncertainties inherent in the process. Known ambiguities, open questions, and assumptions must be externalized rather than implicitly carried forward. In addition, the acceptable scope of refinement during development must be defined in advance, distinguishing legitimate clarification from structural redefinition.

Fourth, the specification must represent the process in a form that allows critical evaluation of its end-to-end maturity and coherence prior to automation. This reduces the risk that structurally immature or incoherent processes become reinforced and normalized through automation.

Design principle 2: Automation initiatives must be supported by explicit governance structures that clarify responsibility boundaries, identify accountable individuals, and institutionalize formal acceptance as a prerequisite for advancing between significant stages of the initiative. The second principle abstracts the requirements concerning responsibility alignment and formal validation (design requirements 6-9). The empirical findings in section 5.1.2 demonstrate that early-stage automation work requires boundary-spanning coordination and is particularly vulnerable to ambiguity in ownership and responsibility. Responsibility allocation in defining processes and

constructing test cases was shown to be particularly dependent on informal negotiation and personal initiative rather than structured governance.

This principle asserts that this ambiguity must be addressed through explicit governance mechanisms to clarify boundaries of responsibility. First, responsibility boundaries must be clarified. Artefacts must make visible which actor is responsible for which aspect of process articulation, validation, and decision-making. Without explicit boundary definition, interdependent tasks risk remaining partially owned or implicitly delegated.

Second, accountable individuals must be identified. Accountability cannot reside solely at the level of abstract roles or units but must be concretized in designated actors who bear responsibility for specific artefact states and decisions. Naming accountable individuals reduces responsibility diffusion and strengthens traceability across initiative stages.

Third, governance must institutionalize formal acknowledgment as a structured commitment mechanism. The requirement that accountable individuals formally acknowledge the adequacy and readiness of the initiative before progressing between significant stages transforms governance from passive documentation into active responsibility enactment. Formal acknowledgment does not merely signal approval of a document but constitutes an explicit affirmation that the current state of process articulation, evaluation, and clarification is sufficient to justify further commitment of resources. From a coordination perspective, such structured points of commitment function as governance gates that stabilize inter-organizational collaboration. By requiring explicit acknowledgment of readiness, the principle reduces the likelihood that unresolved ambiguities are silently carried forward. It also creates a transparent decision moment at which accountability for proceeding is consciously assumed.

Importantly, the principle does not prescribe a specific lifecycle model. Significant stages may vary across organizations and methodologies, including iterative or agile contexts. The core requirement is that progression between materially consequential stages be contingent upon explicit, accountable acknowledgment of adequacy and readiness.

Design principle 3: The primary responsibility for defining and validating the process must reside with the organisational actor that holds substantive ownership, accountability, and contextual expertise over the process and its outcomes, while the automation team provides methodological and technical support. A recurring pattern in the interviews indicates that the responsibility for clarifying difficult ambiguities and uncertainties of the process logic is very commonly passed on to the process automation team.

This does not necessarily reflect deliberate avoidance of responsibility but rather appears as a structural consequence of how automation initiatives unfold. The technical implementation is a phase where ambiguity is no longer tolerable. Unresolved ambiguities ultimately materialize at this point, because the tacit judgment or contextual discretion of human actors is not transferred to the software. In this sense, responsibility does not get consciously transferred but gradually gravitates towards the phase where the process is operationalized to deterministic form. The key concern is the risk of making technical experts responsible for resolving under-specified business logic. Automation experts generally possess methodological and technical expertise but limited substantive authority over the business process. Consequently, forcing resolution at this stage increases coordination friction and risks decisions being made without full contextual grounding. This principle therefore seeks to ensure that ambiguities are resolved at the point in the organization where substantive ownership and contextual expertise reside, rather than being deferred until they become unavoidable.

The principle is intentionally formulated at a level of abstraction that does not tie this responsibility to a specific organizational unit. In many contexts, the "business unit" holds substantive ownership of business processes but the decisive criterion is not formal organizational placement. Instead, the key factor is substantive accountability for process outcomes and possession of contextual domain expertise. Anchoring definitional responsibility on the actor that understands the process and bears process consequences strengthens the alignment of the process specification and the underlying business realities. This alignment also gives substantive meaning to the formal acknowledgment mechanisms articulated in design principle 2. Formal acceptance functions as genuine responsibility enactment only when exercised by actors who hold legitimate ownership over process content. Without such alignment, the governance introduced by design principle 2 risks becoming merely procedural rather than a mechanism of accountable commitment.

Design principle 4: The level of structure and procedural rigor in automation initiatives should be calibrated according to the risk, maturity, and complexity of the process. The analysis of the interview data suggests that insufficient early-stage structure leads to ambiguity, responsibility diffusion and iterative rework. However, workshop discussions also highlighted that imposing uniform procedural depth on all automation initiatives may introduce unnecessary friction in low-risk or well-understood cases which may ultimately undermine artefact adoption. Calibrating structure and governance to the specific automatable process is therefore not a matter of convenience, but a normative design orientation.

The required depth of process articulation, suitability assessment, and formal validation must reflect the characteristics of the process on three contextual dimensions. The first dimension concerns process maturity. A mature process is one that has become institutionalized through repeated enactment and exhibits consistent execution patterns. This stability reduces ambiguity in process definition, decreases the likelihood of late-stage clarification and rework and lowers the coordination effort required to translate the process into executable automation logic.

The second dimension is risk exposure. When automation errors carry significant financial, regulatory, or operational consequences, even minor specification or validation deficiencies can result in disproportionate costs. In high-risk contexts, early-stage governance functions as a risk containment mechanism. Ambiguities must be resolved before implementation because corrective iteration after deployment may incur disproportionate cost or liability. In low-risk contexts, limited iterative refinement may be acceptable, and procedural depth can be reduced accordingly without compromising organizational outcomes.

The third dimension is structural complexity. Processes characterized by dense interdependencies, numerous exception pathways, or cross-functional integration are inherently more difficult to articulate into a coherent and comprehensive specification. Such structural complexity increases the cognitive and coordination burden in early-stage process articulation. As a result, critical variations and dependencies may remain unrecognized, and partial understandings may be prematurely treated as complete.

Calibration does not imply optional governance. Rather, it ensures that early-stage stabilization mechanisms are implemented at a depth aligned with coordination demand. Without calibration, governance may become either underpowered in high-demand contexts or unnecessarily burdensome in low-demand contexts.

6 Operationalization of early-stage coordination

6.1 Artefacts as instantiations of design knowledge

This chapter presents the artefacts developed in this study as the concrete instantiation of the design knowledge articulated in chapter 5. While chapter 5 abstracted empirically grounded coordination challenges into operational design requirements and higher-level design principles, the present chapter demonstrates how these principles materialize in practice in the partner organization through structured artefacts embedded in the early stages of process automation work.

The artefacts introduced here are not mere documentation templates. They function as socio-technical coordination mechanisms intended to stabilize shared process understanding, clarify responsibility boundaries, and structure decision-making across lifecycle transitions. In line with the design science research approach adopted in this thesis, their purpose is to intervene in the coordination-intensive problem space identified in the empirical findings, particularly the instability of shared process understanding, the ambiguity of ownership at the business unit–process automation team boundary, and the absence of explicit governance checkpoints.

Two primary artefacts constitute the core of the designed solution: a revised pre-assessment form and a shared specification document implemented in Confluence, a collaborative workspace and documentation platform. These artefacts are supported by lightweight technical integrations, including automated creation of project identifiers and specification pages, which together form an artefact ecosystem. Rather than operating independently, the artefacts are structurally connected and embedded in the lifecycle of process automation initiatives. They shape how information is articulated, transferred, validated, and formally acknowledged throughout early stages of the process automation lifecycle.

The artefacts were developed iteratively in collaboration with the process automation team's product owner and lead analyst. Both experts confirmed that the redesigned artefacts correspond to the articulated design requirements and reflect the intended design principles. The evaluation conducted at this stage constitutes an expert-based alignment assessment. The artefacts were examined against the identified coordination problems and the derived requirements to ensure conceptual and structural consistency. The artefacts have not yet been systematically evaluated through longitudinal deployment or quantitative performance measurement. Consequently, this chapter does not claim demonstrated improvements in automation outcomes. Instead, it presents artefacts that are designed to mitigate the coordination breakdowns identified in the empirical analysis.

The contribution of this chapter therefore consists of two parts. First, it provides a detailed description of the artefact ecosystem and its integration into the organizational workflow of automation initiatives. Second, it makes explicit how the artefact structures instantiate the design principles derived in the previous chapter. By tracing the link between coordination challenges, design principles, and concrete artefact mechanisms, the chapter establishes the artefacts as operational embodiments of the study's design knowledge.

6.2 The artefact ecosystem and information architecture

The designed solution does not consist of isolated documents but of an interconnected artefact ecosystem that structures how process knowledge is introduced, stabilized, and transferred across lifecycle stages. The ecosystem integrates business unit input, structured evaluation, collaborative specification, and formalized progression gates into a coherent information architecture. Its purpose is to reduce fragmentation of coordination work and to establish a shared reference point that persists throughout the automation lifecycle.

At the entry point of an automation initiative stands the revised pre-assessment form (Appendix 4). When a business unit identifies a candidate process for automation, it submits structured information through this form. The submission triggers the automatic creation of a unique identifier and a corresponding Jira Epic, a high-level work item in the organization's project tracking system. This automation enables traceability between the initial business request and all subsequent development activities. The pre-assessment thus functions not only as an intake instrument but also as the formal initiation mechanism of the automation lifecycle. Once the request for automation is accepted by the process automation team, the Jira Epic is transitioned from backlog to "In Progress" -state. This triggers a second automated integration which creates a dedicated process specification page in Confluence. Relevant information provided in the pre-assessment form is automatically transferred to predefined sections of the specification document. This way early business articulation becomes structurally embedded in the shared specification artefact rather than remaining confined to an intake form or informal discussion.

The Confluence-based specification document functions as the central coordination object of the ecosystem. In comparison to the previous operating model, where documentation was primarily owned and maintained by the process automation team, the redesigned artefact establishes a shared workspace used jointly by business representatives and automation specialists. Role-based sections make explicit which party is responsible for providing, refining, or validating specific information. In this sense, the document operates as a persistent boundary object that supports collaboration across

heterogeneous expertise domains while maintaining a stable structure that anchors discussion and decision-making. The collaborative use of the artifacts is illustrated in Figure 2.

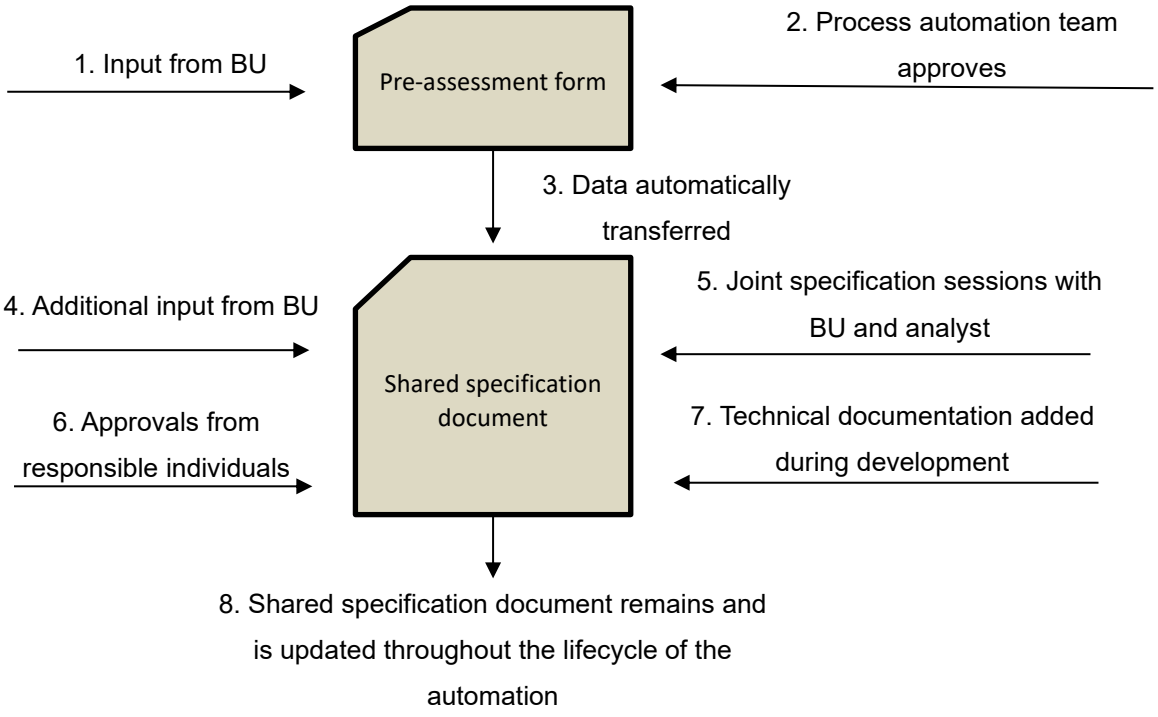


Figure 2. The use of the artefacts

The information architecture can therefore be understood as a staged consolidation process. Initial process articulation originates in the pre-assessment form. This articulation is transferred and deepened in the shared specification document through collaborative definition sessions. As the initiative progresses to development, testing, and production, the same artefact continues to function as a reference baseline for implementation and validation. The document is enriched rather than replaced, ensuring continuity between early assumptions and later technical realization.

Importantly, the ecosystem also establishes a clear authoritative source for coordination. While informal communication channels such as meetings, chat discussions, and personal notes remain part of everyday work, they no longer serve as primary repositories of process definition. Instead, the Confluence specification document is positioned as the structured and stable reference point for agreed process knowledge, approval decisions, and documented uncertainties. This structural centralization reduces reliance on fragmented artefacts and individual memory, which were identified in the empirical analysis as recurring sources of coordination fragility.

The technical integrations that support the ecosystem, such as automatic Jira Epic creation, automatic specification page generation, and structured data transfer, do not in themselves constitute the

theoretical contribution of this study. Instead, they enable consistent enactment of the designed coordination structures. By embedding artefact creation and linkage directly into lifecycle transitions, the organization reduces the likelihood that specification work remains ad hoc, optional, or detached from formal progression. In this way, the integrations strengthen enactability and traceability, ensuring that the artefact design is structurally coupled to everyday workflow rather than dependent solely on voluntary adherence.

6.3 Artefacts across the automation lifecycle

The artefact ecosystem described in the previous section is embedded in the lifecycle of automation initiatives and structures how transitions between stages occur. Rather than serving as static documentation repositories, the artefacts actively regulate progression through calibrated approval gates and structured readiness criteria. In this sense, they function not only as boundary objects but also as governance instruments that formalize when and on what basis resource commitments are made. The lifecycle is divided into four primary transitions: pre-assessment to definition, definition to development, development to testing, and testing to production. At each transition, the artefacts provide both informational structure and a formalized "approval to proceed" mechanism. Importantly, the depth and strictness of these approvals are calibratable depending on the maturity, risk profile, and complexity of the automation candidate.

6.3.1 First lifecycle transition: Pre-assessment to definition.

The process automation lifecycle starts after a business unit submits the pre-assessment form. At this stage, the artefact ensures that a sufficient initial articulation of the process exists before joint specification work begins. The pre-assessment form requires the business unit to describe the process extensively but at a high level. Before joint definition workshops are scheduled, an analyst from the process automation team evaluates whether the submitted information reaches a sufficient level of clarity to justify collaborative specification sessions. The "approval to proceed" at this stage signals that the process has been articulated at least to a degree that enables structured joint refinement. Depending on the project context, this approval may involve both automation team representatives and designated business experts. This gate institutionalizes an important shift from the previous operating model. Previously, the business unit's initial process description constituted a starting point that could remain insufficient as the initiative proceeded to the collaborative definition meetings. Now, explicit readiness for collaborative definition must be formally acknowledged before proceeding.

6.3.2 Second lifecycle transition: Definition to development.

The transition from definition to development represents a critical commitment point. During collaborative workshops, the specification document is populated and refined jointly by business representatives and automation specialists. Process logic, inputs, outputs, exception handling, contextual dependencies, and responsibilities are made explicit within the structured Confluence template.

Two structural features are central at this stage. First, the specification document includes role-based responsibility tables that identify accountable individuals and their areas of expertise. Second, it contains a dedicated section for documenting known uncertainties, including open questions and explicitly accepted assumptions. This ensures that incomplete knowledge is surfaced rather than implicitly carried forward. Progression to development requires formal acknowledgment from relevant roles. From the automation team perspective, the developer and analyst confirm that the specification provides a sufficiently reliable basis for implementation. From the business perspective, lead process experts or designated subject matter experts validate that the documented process representation corresponds to operational reality.

The approval mechanism does not imply that all uncertainties must be eliminated. Rather, it requires that remaining ambiguities must be explicitly documented and that their scope understood. In this way, the artefacts distinguish between acceptable iterative refinement and structural instability that would fundamentally alter the automation scope.

6.3.3 Third lifecycle transition: Development to testing.

Once development begins, the specification document continues to function as the authoritative baseline. The developer implements the automation according to the documented process logic and enriches the artefact with technical implementation details. The transition to testing requires confirmation that the implemented solution corresponds to the approved specification. Here, the artefact serves as the evaluative anchor as testing activities are assessed against the documented process representation. The developer’s approval signals technical conformity to the specification, while business experts participate in validating functional correctness. By maintaining a shared artefact across definition and development, the ecosystem reduces the risk that implementation diverges from earlier agreements or that undocumented reinterpretations occur.

6.3.4 Fourth lifecycle transition: Testing to production.

The final transition from testing to production formalizes operational acceptance. At this stage, designated business unit representatives confirm that the automation behaves as intended in the relevant environment. Depending on the complexity and risk profile of the initiative, this may involve staged iterative validation in first test and then production environments. The “approval to proceed” at this stage institutionalizes accountability for operational readiness. It ensures that the decision to deploy is not merely implicit or based on informal agreement but is explicitly acknowledged by accountable actors.

Importantly, this lifecycle structure is not intended to function as a rigid waterfall model. The artefacts support iterative refinement where necessary. However, progression between materially consequential stages requires explicit acknowledgment that the current state of articulation, implementation, and validation is sufficient. This combination of flexibility and formalized commitment operationalizes the calibrated governance logic articulated in the design principles. Through these lifecycle embeddings, the artefacts transform coordination from an ongoing, primarily communication-driven activity into a structured sequence of collaboratively assessed states. Each transition requires visible articulation, accountable ownership, and explicit acknowledgment, thereby reducing the likelihood that ambiguities, responsibility diffusion, or undocumented assumptions silently propagate across stages.

6.4 Artefact 1 – Pre assessment form

The revised pre-assessment form constitutes the entry-point artefact of the automation lifecycle. Its primary function is to structure early articulation of the candidate process before collaborative specification begins. In the previous operating model, the pre-assessment form primarily functioned as a lightweight intake instrument. The level of detail provided by business units varied considerably, which often resulted in insufficient shared understanding at the outset of definition workshops. The redesigned form addresses this structural weakness by deepening and systematizing early process articulation. The purpose of the artefact is not to produce a complete and automation-ready specification. Rather, it is designed to elevate the minimum level of explicit process understanding before resource-intensive collaborative work begins. In doing so, it supports early suitability assessment and improves the readiness for collaborative work of both business representatives and automation analysts. The revised pre-assessment form structures the articulation of several

dimensions of the candidate process. These are collectively represented in Table 6. The full contents of the form are shown in Appendix 4.

Table 6. Dimensions of the pre-assessment form

Dimension	Addressed in fields
A high-level description of the process and its purpose	5, 6
Identification of exceptions and deviations from the standard workflow	7, 8
Description of upstream and downstream context	9, 11
Description of input data	10
Benefits of the automation	20, 21, 22, 23, 24
Target systems	16
Identification of relevant process experts	14, 15
Information on test environments, test data, and access rights	17, 18, 19

By requiring articulation across these dimensions, the form moves beyond a simple "automation request submission". It encourages business units to reflect on the process structure, its stability, and its operational environment before the automation team becomes deeply involved. This early articulation directly supports the coordination challenges identified in the empirical findings. First, it reduces the likelihood that fundamental aspects of the process remain entirely implicit at the start of definition workshops.

Second, it exposes early signals of process maturity, exception density, and system complexity that are relevant for automation suitability assessment. Compared to the previous version, the redesigned form introduces greater granularity and clearer separation of concerns. Questions have been refined, subdivided, and expanded to encourage more precise articulation. In particular, the explicit focus on up- and downstream process context and system-level dependencies strengthens the analytical basis for early assessment of end-to-end process maturity and coherence.

The form also integrates more explicit identification of responsible individuals and subject matter experts. This addresses the empirically observed challenge of unclear ownership at early stages by requiring the business unit to name accountable actors before joint work begins. In addition, the structured collection of process characteristics enhances the ability of the automation team to estimate the degree of structural complexity and potential operational impact. Questions concerning exception handling, up- and downstream dependencies, the existence of a stable manual process, the number of automation output dependencies, the availability of test environments, and the thematic domain of

the process are directly relevant factors for calibrating procedural rigor in subsequent stages. While the form remains flexible and does not require exhaustive documentation, its structure makes superficial articulation more visible by exposing areas requiring clarification before proceeding.

Beyond articulation, the pre-assessment form plays a central role in early suitability screening and backlog prioritization. The structured information provided serves as the basis for the automation team's preliminary assessment of feasibility, complexity, and expected organisational impact. Workload estimation and prioritization decisions are thus grounded in systematically collected input rather than informal descriptions.

The form also serves an important function in preparing for collaborative definition workshops. By requiring business units to more extensively articulate the process in advance, the artefact aims to increase their cognitive engagement before joint definition sessions. At the same time, it provides the analyst with an initial conceptual model of the process, enabling more targeted and analytically informed questioning. This aims to reduce the "exploratory probing" dynamic described in the empirical findings and improves the quality of early dialogue.

The pre-assessment form therefore operates as a preliminary boundary object. It does not yet constitute a fully shared specification, but it establishes a structured representation that both sides can build upon. By elevating the baseline of shared understanding before formal definition begins, the artefact supports the stabilization of coordination in the earliest stage of the automation lifecycle. It is important to note that the form does not eliminate variability in the quality of business input, nor does it guarantee that the underlying process is mature or stable. Its contribution lies in making early articulation more structured, comparable, and visible, thereby improving the conditions under which subsequent collaborative specification work takes place.

6.5 Artefact 2 – Shared specification document

The Confluence-based specification document constitutes the central coordination artefact of the redesigned automation lifecycle. Whereas the pre-assessment form structures initial articulation, the specification document serves as the authoritative and persistent representation of the process throughout definition, development, testing, and deployment. Its design reflects a deliberate shift from fragmented, role-specific documentation toward a shared boundary object jointly shaped and validated by business representatives and automation specialists. In the previous operating model, the specification document was primarily maintained by the automation team. Business participation was largely confined to meetings and informal validation, while the document itself functioned mainly as

an internal implementation artefact. The redesigned version repositions the document as a shared workspace and governance instrument, explicitly allocating responsibility, embedding approval mechanisms, and structuring the management of uncertainty.

The specification document is generated automatically when an automation initiative transitions into active work. Selected information from the pre-assessment form is transferred into predefined sections of the template, ensuring continuity between early articulation and formal specification. The document is structured into clearly delineated sections, some primarily filled by business representatives and others by members of the automation team. High-level process description, contextual information, and business-relevant justifications are contributed by the business unit. Detailed process definitions, exception tables, business rules, scheduling requirements, and reporting structures are collaboratively refined, while technical architecture and implementation details are added by developers during and after development. This role-based segmentation clarifies who is responsible for providing which type of information. It transforms the document into an explicit coordination interface rather than a passive repository. Each section is not merely informational but tied to accountability and validation expectations.

A key innovation of the redesigned specification document lies in its embedded governance structures. Dedicated tables identify key personnel from both the automation team and the business unit, including their roles, areas of expertise, core responsibilities, and authority to grant approval to proceed between lifecycle stages. The “approval to proceed” mechanism is integrated directly into the document and functions as a calibrated governance gate. Approval is not tied to a fixed set of roles but is adaptable depending on project characteristics such as risk exposure, process complexity, and maturity. Approvals can be solicited from different project roles such as analysts, developers, lead process experts, subject matter experts or target system experts.

By requiring key individuals to explicitly acknowledge readiness before progression, the artefact institutionalizes accountability which no longer remains implicit or collectively diffused. Instead, accountability becomes visible and attributable. This directly addresses the ambiguity of ownership identified in the empirical findings. Importantly, approval does not imply absolute completeness. Rather, it signifies an informed decision that the current state of articulation and validation is deemed sufficient to justify moving to the next lifecycle stage under known conditions.

Recognizing that complete definitive specification is rarely achievable in complex processes, the specification document includes a dedicated section for documenting known uncertainties in the process description. This part of the document is designed to capture both open questions and

explicitly accepted assumptions or scope limitations. By externalizing uncertainty, the artefact distinguishes between acknowledged incompleteness and latent ambiguity. It is designed to make visible which aspects of the process remain unresolved and under what conditions iterative refinement during development is considered acceptable. This mechanism is designed to operationalize the distinction between manageable refinement and structural instability. Rather than attempting to eliminate all uncertainty before development, the artefact aims to ensure that uncertainty is collectively recognized and bounded. By doing so, it is designed to reduce the likelihood that critical ambiguities remain implicit until late-stage implementation or production use.

The specification document fulfils a dual role. During definition, it operates as a collaboratively constructed process representation. During development and beyond deployment, it evolves into a combined functional and technical documentation artefact. Developers enrich the document with information concerning technical structures, dependencies, configuration parameters, and implementation details. Maintenance documentation, known issues, and operational instructions are incorporated into the same artefact. This continuity reduces fragmentation between design intent and technical realization. By maintaining a single, evolving coordination object across stages, the artefact supports traceability between business articulation, implementation logic, testing validation, and operational maintenance.

The document thereby serves not only as a boundary object during early coordination but also as a durable knowledge base throughout the lifecycle of the automation. It is important to note that the redesigned specification document does not eliminate iterative clarification. Some refinement remains inherent to complex development work. However, by structuring articulation, embedding accountability, and explicitly managing uncertainty, the artefact reduces uncontrolled iteration and late-stage reinterpretation. Its contribution lies in stabilizing coordination conditions rather than enforcing rigid procedural completeness. The template of the Shared specification document is included in Appendix 5.

6.6 Instantiating the design principles

The preceding sections have described the artefacts and their embedding in the automation lifecycle. This section showcases how the described artefact ecosystem instantiates the design principles derived in Chapter 5.

Design principle 1: Automation initiatives must be grounded in an explicitly articulated and critically assessable process representation that stabilizes shared understanding and makes both uncertainty

and acceptable scope of change visible prior to development. This design principle emphasizes the need for explicit and critically assessable representations of candidate processes prior to automation development. This is instantiated across both primary artefacts. In the pre-assessment form, structured questions concerning process flow, inputs, outputs, exceptions, system dependencies, and up- and downstream context elevate early articulation beyond informal descriptions. The shared specification document takes the description further in depth and detail. The high-level description is complemented by detailed step-by-step process definitions, exception tables, business rules, scheduling requirements, and contextual dependencies. The artefact thereby supports layered articulation from preliminary overview to operational granularity. The inclusion of a dedicated known uncertainties section operationalizes the principle of making uncertainty and acceptable scope of change visible. By documenting open questions and explicitly accepted assumptions, the artefact differentiates between acknowledged incompleteness and latent ambiguity. This mechanism ensures that uncertainty is visible and bounded rather than silently embedded in development work.

Design principle 2: Automation initiatives must be supported by explicit governance structures that clarify responsibility boundaries, identify accountable individuals, and institutionalize formal acceptance as a prerequisite for advancing between significant stages of the initiative. This principle is directly instantiated in the role tables embedded in the specification document. Key personnel from both the automation team and the business unit are identified along with their areas of expertise, core responsibilities, and authority to grant approval to proceed between stages. Responsibility is therefore not inferred but formally declared. The "Approval to proceed" mechanism further operationalizes governance. Progression from pre-assessment to definition, definition to development, development to testing, and testing to production requires explicit acknowledgment from relevant roles. This transforms stage transitions from informal agreements into accountable decisions.

This governance is structured but not rigid. The requirement for approvals can be calibrated depending on project characteristics. In low-risk or mature contexts, fewer actors may be required to grant approval. In complex or high-risk contexts, multiple domain experts may be involved. This design balances clarity of accountability with procedural flexibility. Through these structures, responsibility, accountability and formal acceptance are instantiated visibly and attributably rather than relying on implicit coordination.

Design principle 3: The primary responsibility for defining and validating the process must reside with the organisational actor that holds substantive ownership, accountability, and contextual expertise over the process and its outcomes, while the automation team provides methodological and

technical support. Previously, responsibility for process articulation frequently gravitated toward the automation team, making the analysts and developers effectively responsible for building a documentation of the business process. The redesigned artefact ecosystem aims to restructure this responsibility distribution through concrete procedural mechanisms. First, the revised pre-assessment form requires the business unit to provide more detailed and structured articulation of the candidate process at the initiation stage. Compared to the previous operating model, this increases the scope and depth of information that must be provided before the automation team begins collaborative definition work. As a result, part of the articulation effort that previously emerged during analyst-led workshops is shifted upstream to the business unit.

Second, the Confluence specification document is no longer solely the responsibility of the automation team. Role-based sections explicitly require business representatives to contribute to and complete defined parts of the documentation before workshops and prior to stage transitions. The responsibility for describing core process logic, contextual dependencies, and operational constraints is therefore structurally assigned to the business side rather than informally assumed by automation specialists.

Third, the embedded "Approval to proceed" mechanisms formalize business accountability for documentation sufficiency and correctness. Named business actors must explicitly confirm that the specification accurately reflects the operational process before the initiative advances to development or production. This requirement ensures that substantive validation of the specification remains with domain experts. This design prevents the automation team from becoming the sole owner of process definition. Instead, the automation team contributes methodological and technical expertise, while the business retains authority over substantive process content.

Design principle 4: The level of structure and procedural rigor in automation initiatives should be calibrated according to the risk, maturity, and complexity of the process. As a basis for calibrating the level of structure and procedural rigor, the pre-assessment form collects information such as exception frequency, dependencies and downstream impacts of the process. These provide early signals of the expected coordination demand which the process automation analysts and developers can use to adjust the depth of specification and governance during the initiative. Also, the embedded "Approval to proceed" mechanism calibrates its depth and granularity by the amount of involved business units and subject matter experts. For straightforward processes with limited scope, just one lead process expert can be enough to validate the process specification while cross functional and broadly impactful processes can require the validation of several process experts. By combining

structured articulation with adaptable governance thresholds, the artefacts operationalize proportional rigor. Governance is neither optional nor uniformly imposed. It is scaled to contextual demands.

To summarize, the artefact ecosystem translates the abstract design principles into operational structures. Explicit representation (design principle 1), accountable governance (design principle 2), substantive ownership anchoring (design principle 3), and calibrated rigor (design principle 4) are not implemented as separate mechanisms but embedded across the pre-assessment form, the shared specification document, and the lifecycle approval gates. The artefacts therefore constitute a coherent instantiation of the study's design knowledge. Their contribution lies not in isolated structural features but in the way these features collectively reshape early-stage coordination in process automation initiatives.

6.7 The scope and limitations of the artefacts

While the artefact ecosystem operationalizes the design principles derived in this study, its effectiveness is contingent upon specific organizational conditions. This section outlines the scope conditions under which the artefacts are expected to function as intended and clarifies the limitations of the current instantiation.

First, the artefacts assume the availability and engagement of responsible domain experts within the business unit. The pre-assessment form and the shared specification document are structured to anchor ownership in named individuals. However, if designated process owners or subject matter experts lack time, authority, or willingness to engage in articulation and validation work, the structural mechanisms alone cannot ensure substantive ownership. The artefacts make accountability visible, but they do not substitute for organizational commitment.

Second, the ecosystem does not eliminate structural constraints related to technical environments. Challenges such as missing test systems, inadequate test data, or restricted access to systems cannot be resolved solely through coordination mechanisms. While the revised pre-assessment form surfaces these issues earlier in the lifecycle, the artefacts do not remove the underlying infrastructural dependencies.

Third, the effectiveness of calibrated governance depends on appropriate judgment in assigning approval roles and rigor levels. The design allows flexibility in determining how many actors are required to grant approval and at which depth specification must be completed. If calibration is applied too loosely in complex contexts, coordination fragility may persist. Conversely, excessive

rigor in low-risk initiatives may introduce unnecessary friction. The artefact ecosystem provides the structural possibility for proportional governance but does not automate the calibration decision itself.

Fourth, the artefacts have been evaluated through expert alignment rather than longitudinal deployment. The product owner and lead analyst confirmed that the redesigned artefacts correspond to the identified design requirements and principles. However, systematic empirical assessment of their impact on development efficiency, rework frequency, or production stability has not yet been conducted. The present contribution therefore remains at the level of designed coordination mechanisms rather than empirically validated performance improvements.

Finally, the artefacts are embedded in a specific organizational and technological context. Their instantiation relies on the integration of intake forms, project management tooling, and collaborative documentation platforms. While the underlying coordination principles are transferable, the specific implementation described here reflects the tooling environment and governance culture of the case organization.

7 Conclusion and discussion

7.1 Summary of findings

This study adopted a two-layer research design combining empirical problem instantiation with artefact-oriented design. The first research question addressed the analytical task of identifying how coordination breakdowns manifest in early-stage process automation work of the studied context. The second research question focused on translating these empirical findings into artefact-level design responses intended to mitigate the identified instabilities. This section synthesizes the findings in relation to each research question.

RQ1: How do coordination breakdowns manifest in early-stage process automation work?

The empirical analysis indicates that coordination breakdowns in early-stage process automation work manifest as instability in shared process representations, diffusion of responsibility across organizational boundaries, and socially constrained articulation of process knowledge. These manifestations are not isolated issues but structurally interrelated expressions of coordination fragility in the early phases of automation initiatives.

First, breakdowns materialize as unstable or incomplete shared process understanding. Process knowledge is often fragmented, tacit, or unevenly distributed across actors. In the absence of structured and consolidated representations, partial understandings are prematurely treated as complete. This instability does not necessarily surface immediately during early discussions, but becomes visible in later development, testing, or production stages when implicit assumptions are confronted. These downstream issues can be interpreted as delayed indications of early-stage representational incompleteness.

Second, coordination breakdowns manifest as ambiguity in responsibility and ownership. While automation initiatives formally involve both business units and automation specialists, accountability for defining, validating, and maintaining the process logic remains insufficiently anchored. This diffusion of responsibility weakens commitment to thorough articulation and increases the likelihood that clarification is deferred until later stages. The absence of structurally embedded ownership mechanisms reinforces instability in shared representations.

Third, the analysis highlights that process articulation is socially sensitive. Requests to clarify or formalize processes may be interpreted as criticism, control, or exposure of deficiencies. This social

fragility constrains the depth and rigor of early-stage articulation work which contributes to conditions where coordination relies excessively on informal communication and iterative clarification rather than stable artefacts. The pattern is not limited to procedural deficiencies. Rather, coordination breakdowns emerge from the interplay between structural ambiguity and social interaction.

RQ2: How can artefact-level structuring be designed to mitigate these coordination breakdowns?

Building on the identified manifestations of coordination breakdown, the second research question examined how artefact-level structuring can be designed to mitigate these instabilities in early-stage process automation work. The findings suggest that stabilization requires deliberate structuring of the artefacts that mediate collaboration between business and automation actors.

At the representational level, artefact structuring is designed to formalize shared process definitions. Structured documentation templates, explicit articulation of process steps and variations, and systematic recording of assumptions externalize tacit and fragmented knowledge into collectively accessible representations. By shifting coordination away from iterative verbal clarification toward structured consolidation within shared artefacts, the design aims to reduce the likelihood that implicit assumptions remain unexamined. This externalization also reshapes the interaction context in which articulation takes place. When clarification requirements are embedded in predefined artefact fields rather than raised as direct interpersonal challenges, the exchange of information becomes less personalized.

The pre-assessment form and shared specification document are intended to mediate critical exchanges through structured prompts instead of spontaneous probing. This depersonalization is designed to reduce the likelihood that requests for clarification are interpreted as criticism or competence evaluation. The pre-assessment form also aims to improve readiness for collaborative work by prompting advance reflection on key process characteristics. Entering joint workshops with more structured initial input is intended to enable more deliberate participation and reduce situations in which complex questions must be addressed reactively. Similarly, improved initial information aims to provide automation analysts with a clearer starting point, potentially decreasing repeated probing and iterative clarification cycles during meetings.

In this way, artefact-level structuring seeks to stabilize both the representational and relational dimensions of early-stage coordination. Responsibility diffusion is addressed by embedding named accountability and explicit approval points directly into the artefact ecosystem. Clearly defined

ownership roles and documented validation mechanisms aim to anchor commitment to the articulated process definition. Rather than relying on informal expectations, responsibility becomes visible and formally acknowledged, strengthening the link between process articulation and organizational authority.

Finally, the incorporation of explicit sections for documenting assumptions, open questions, and known limitations is intended to normalize bounded incompleteness. By formally recognizing that early-stage process definitions may contain unresolved elements, the artefact structure seeks to legitimize transparent articulation of uncertainty. When incompleteness is documented rather than implicitly concealed, ambiguity becomes visible and collectively acknowledged coordination work, reframing articulation from evaluative scrutiny toward collaborative stabilization. Effective mitigation further requires calibration of procedural rigor. Artefact-level structuring must remain proportionate to the maturity, risk, and complexity of the process under consideration. The designed artefact ecosystem therefore incorporates adjustable depth in articulation and validation so that stabilization mechanisms remain commensurate with coordination demands.

7.2 Theoretical contributions

This study contributes to the literature on process automation and coordination in three interrelated ways. Rather than proposing a new lifecycle model or technological framework, the contribution lies in reframing early-stage automation challenges as coordination instability and articulating artefact-level design principles that address this instability.

First, the study offers a coordination-centered reinterpretation of recurring automation difficulties. Prior literature frequently attributes implementation challenges to technical limitations, tool maturity, or organizational resistance. While these factors are relevant, the present findings indicate that many downstream implementation issues originate in early-stage coordination fragility. Instability in shared process representations, diffusion of responsibility, and socially constrained articulation shape the conditions under which development begins. By analytically isolating these mechanisms, the study shifts explanatory emphasis from technical capability to the structuring of collaborative work preceding implementation. This reframing highlights early-stage stabilization as a foundational governance concern rather than a preparatory formality.

Second, the study extends boundary object theory within the context of process automation. Existing discussions emphasize the role of shared artefacts in facilitating communication across professional boundaries. The findings suggest that in automation initiatives, boundary objects must serve not only

as representational intermediaries but also as institutional commitment devices. Artefacts that merely support mutual understanding are insufficient if responsibility for validation and maintenance remains ambiguous. Effective boundary objects in this context must therefore integrate representation, accountability, and formalized approval mechanisms. This integration positions artefacts simultaneously as mediators of knowledge and as governance instruments embedded in organizational structures.

Third, the study contributes design knowledge in the form of structured principles for artefact-level stabilization in early-stage automation work. The articulated design principles specify how process representations can be formalized, how responsibility can be anchored, how bounded uncertainty can be legitimized, and how procedural rigor can be calibrated to contextual demands. These principles are analytically grounded in empirical findings yet abstracted beyond the specific artefact instantiation. As such, they offer transferable design guidance for organizations seeking to structure early-stages of automation initiatives without prescribing a rigid lifecycle model.

In summary, the theoretical contribution of this study lies in linking coordination theory, boundary object theory, and design science research within the domain of process automation. By conceptualizing early-stage automation as a problem of representational and institutional stabilization, the study advances understanding of how organizational structuring precedes and shapes technical implementation outcomes within the domain of process automation.

7.3 Practical implications

The practical relevance of this study lies less in the specific artefact instantiation and more in the orientation it proposes toward early-stage automation work. The findings suggest that organizations may benefit from reconsidering how they conceptualize the preparatory phase of automation initiatives.

A central implication is that early-stage automation should be treated as a governance-intensive phase rather than merely a preparatory administrative step. Before development begins, organizations must ensure that shared understanding and responsibility structures are sufficiently stabilized. When these foundations remain implicit, downstream implementation difficulties are likely to surface regardless of technical competence. Allocating deliberate attention to how collaborative work is structured in this phase may therefore reduce the need for reactive correction later.

This study also highlights the importance of making responsibility for process definition explicit. In cross-functional automation initiatives, accountability for articulation quality can easily become

diffused. Without clear ownership of definition validity, clarification tends to be deferred, and assumptions remain embedded in informal exchanges. Organizations may therefore need to examine not only who develops automations, but who is formally accountable for the correctness and completeness of the process logic that underpins them.

Another implication concerns the social conditions under which process articulation occurs. Clarification work can carry interpersonal sensitivities, particularly when process knowledge is unevenly distributed. Structuring preparatory reflection and documentation prior to joint workshops may improve readiness and reduce friction during collaborative sessions. Attention to interaction design in this phase may be as important as the technical tools used in later development.

The findings also suggest that governance mechanisms in automation initiatives should be proportionate rather than uniform. The required depth of articulation and validation depends on contextual characteristics such as process maturity and complexity. Overly rigid control structures risk rejection, while insufficient structure leaves coordination vulnerable. Organizations may therefore benefit from developing adjustable governance approaches that align procedural rigor with coordination demand.

Overall, these implications indicate that the effectiveness of technical automation efforts is partly contingent on how early-stage coordination is structured and institutionalized.

7.4 Limitations and directions for future research

The scope of this study is defined by many limitations which should be considered when interpreting the contributions. At the same time, these boundaries indicate directions for further investigation.

The empirical analysis was conducted within a single organizational context. The identified manifestations of coordination breakdown reflect patterns observed in one automation team and its associated business units. While the analytical framing draws on established theoretical perspectives, the empirical findings illustrate how coordination instability manifested in this specific setting rather than constituting an exhaustive taxonomy of breakdown types. Future research could examine whether similar coordination patterns emerge across organizations with different governance models, levels of automation maturity, or industry contexts.

The artefact design was evaluated through expert review and conceptual alignment assessment rather than longitudinal implementation. The study does not provide empirical evidence of improved automation outcomes following deployment of the proposed artefact ecosystem. Instead, the

evaluation focused on whether the artefact structures logically addressed the identified coordination breakdowns. Longitudinal studies examining how such artefact structuring influences implementation stability, rework cycles, or stakeholder perceptions over time would provide valuable empirical validation of the proposed design principles.

Although the articulated design principles are abstracted beyond the specific templates presented, their operationalization remains influenced by the case organization's governance structures and tool ecosystem. Future research could explore alternative instantiations of the same principles in different technological or organizational environments, thereby refining understanding of which aspects of the design are context-dependent and which remain stable across settings.

Finally, the study deliberately focused on early-stage coordination work. It does not examine in detail how technical development practices, change management processes, or post-deployment governance interact with early-stage structuring. Investigating how stabilization mechanisms in early phases shape downstream implementation dynamics would deepen understanding of the longitudinal interplay between coordination design and technical execution.

These directions suggest that further research is needed not only to evaluate the impact of artefact-level structuring but also to refine theoretical understanding of coordination stabilization in automation contexts. The present study provides an analytically grounded starting point for such inquiry.

References

Afrin, S., Roksana, S., & Akram, R. (2025). AI-Enhanced Robotic Process Automation: A Review of Intelligent Automation Innovations. IEEE Access, 13, 173–197. https://doi.org/10.1109/ACCESS.2024.3513279

Baskerville, R. L. (1999). Investigating Information Systems with Action Research. Communications of the Association for Information Systems, 2(1). https://doi.org/10.17705/1CAIS.00219

Cascais Brás, J., Pereira, R. F., Melo, M., Bianchi, I. S., & Ribeiro, R. (2025). Balancing Business, IT, and Human Capital: RPA Integration and Governance Dynamics. Information (Switzerland), 16(9). https://doi.org/10.3390/info16090793

Deng, Z., Guo, Y., Han, C., Ma, W., Xiong, J., Wen, S., & Xiang, Y. (2025). AI Agents Under Threat: A Survey of Key Security Challenges and Future Pathways. ACM Computing Surveys, 57(7), 1–36. https://doi.org/10.1145/3716628

Enríquez, J. G., Jiménez-Ramírez, A., Domínguez-Mayo, F. J., & García-García, J. A. (2020). Robotic Process Automation: A Scientific and Industrial Systematic Mapping Study. IEEE Access, 8, 39113–39129. https://doi.org/10.1109/ACCESS.2020.2974934

Eulerich, M., Pawlowski, J., Waddoups, N. J., & Wood, D. A. (2022). A Framework for Using Robotic Process Automation for Audit Tasks. Contemporary Accounting Research, 39(1), 691–720. https://doi.org/10.1111/1911-3846.12723

Eulerich, M., Waddoups, N., Wagener, M., & Wood, D. A. (2023). The Dark Side of Robotic Process Automation (RPA): Understanding Risks and Challenges with RPA. Accounting Review, 35(3). https://doi.org/10.2308/HORIZONS-2022-019

Eulerich, M., Waddoups, N., Wagener, M., & Wood, D. A. (2024). Development of a Framework of Key Internal Control and Governance Principles for Robotic Process Automation (RPA). Journal of Information Systems, 38(2), 29–49. https://doi.org/10.2308/ISYS-2023-067

Farinha, D., Pereira, R., & Almeida, R. (2024). A framework to support Robotic process automation. Journal of Information Technology, 39(1), 149–166. https://doi.org/10.1177/02683962231165066

Figueiredo, A. S., & Pinto, L. H. (2020). Robotizing shared service centres: Key challenges and outcomes. Journal of Service Theory and Practice, 31(1), 157–178. https://doi.org/10.1108/JSTP-06-2020-0126

Götzen, R., Stamm, J. V., Conrad, R., & Stich, V. (2022). Understanding the Organizational Impact of Robotic Process Automation: A Socio-Technical Perspective. In L. M. Camarinha-Matos,

A. Ortiz, X. Boucher, & A. L. Osório (Eds.), IFIP Advances in Information and Communication Technology: AICT-662 (pp. 106–114). Springer International Publishing. https://doi.org/10.1007/978-3-031-14844-6_9

Herm, L.-V., Janiesch, C., Helm, A., Imgrund, F., Hofmann, A., & Winkelmann, A. (2023). A framework for implementing robotic process automation projects. Information Systems and E-Business Management, 21(1), 1–35. https://doi.org/10.1007/s10257-022-00553-8

Hevner, A., & Chatterjee, S. (2010). Design Science Research in Information Systems. In A. Hevner & S. Chatterjee (Eds.), Design Research in Information Systems: Theory and Practice (pp. 9–22). Springer US. https://doi.org/10.1007/978-1-4419-5653-8_2

Huang, F., & Vasarhelyi, M. A. (2019). Applying robotic process automation (RPA) in auditing: A framework. International Journal of Accounting Information Systems, 35, 100433. https://doi.org/10.1016/j.accinf.2019.100433

Iivari, J. (2020). Editorial: A Critical Look at Theories in Design Science Research. Journal of the Association for Information Systems, 21(3). https://doi.org/10.17705/1jais.00610

Kraus, P., Fißler, E., & Schlegel, D. (2024). A typology of challenges in the context of robotic process automation implementation projects. International Journal of Organizational Analysis, 32(11), 60–73. https://doi.org/10.1108/IJOA-11-2023-4100

Lacity, M. C., & Willcocks, L. P. (2016). A New Approach to Automating Services. MIT Sloan Management Review. https://sloanreview.mit.edu/article/a-new-approach-to-automating-services/

Lacity, M., & Willcocks, L. (2021). Becoming Strategic with Intelligent Automation. MIS Quarterly Executive, 20(2), 169–182.

Malone, T. W., & Crowston, K. (1994). The interdisciplinary study of coordination. ACM Comput. Surv., 26(1), 87–119. https://doi.org/10.1145/174666.174668

Ng, K. K. H., Chen, C.-H., Lee, C. K. M., Jiao, J. (Roger), & Yang, Z.-X. (2021). A systematic literature review on intelligent automation: Aligning concepts from theory, practice, and future perspectives. Advanced Engineering Informatics, 47, 101246. https://doi.org/10.1016/j.aei.2021.101246

Pramod, D. (2021). Robotic process automation for industry: Adoption status, benefits, challenges and research agenda. Benchmarking: An International Journal, 29(5), 1562–1586. https://doi.org/10.1108/BIJ-01-2021-0033

Ribeiro, J., Lima, R., Eckhardt, T., & Paiva, S. (2021). Robotic Process Automation and Artificial Intelligence in Industry 4.0 – A Literature review. Procedia Computer Science, CENTERIS 2020 - International Conference on ENTERprise Information Systems / ProjMAN 2020 -

International Conference on Project MANagement / HCist 2020 - International Conference on Health and Social Care Information Systems and Technologies 2020, CENTERIS/ProjMAN/HCist 2020, 181, 51–58. https://doi.org/10.1016/j.procs.2021.01.104

Siderska, J., Aunimo, L., Süße, T., Stamm, J. V., Kedziora, D., & Aini, S. N. B. M. (2023). Towards Intelligent Automation (IA): Literature Review on the Evolution of Robotic Process Automation (RPA), its Challenges, and Future Trends. Engineering Management in Production and Services, 15(4), 90–103. https://doi.org/10.2478/emj-2023-0030

Star, S. L., & Griesemer, J. R. (1989). Institutional Ecology, `Translations' and Boundary Objects: Amateurs and Professionals in Berkeley's Museum of Vertebrate Zoology, 1907-39. Social Studies of Science, 19(3), 387–420. https://doi.org/10.1177/030631289019003001

Suri, V. K., Elia, M., & van Hillegersberg, J. (2017). Software Bots—The Next Frontier for Shared Services and Functional Excellence. In I. Oshri, J. Kotlarsky, & L. P. Willcocks (Eds.), Global Sourcing of Digital Services: Micro and Macro Perspectives (pp. 81–94). Springer International Publishing. https://doi.org/10.1007/978-3-319-70305-3_5

Syed, R., Suriadi, S., Adams, M., Bandara, W., Leemans, S. J. J., Ouyang, C., ter Hofstede, A. H. M., van de Weerd, I., Wynn, M. T., & Reijers, H. A. (2020). Robotic Process Automation: Contemporary themes and challenges. Computers in Industry, 115. https://doi.org/10.1016/j.compind.2019.103162

Wewerka, J., & Reichert, M. (2023). Robotic process automation—A systematic mapping study and classification framework. Enterprise Information Systems, 17(2), 1986862. https://doi.org/10.1080/17517575.2021.1986862

Winograd, T. (with Flores, F.). (1990). Understanding computers and cognition: A new foundation for design (4th pr.). Addison-Wesley.

Xi, Z., Chen, W., Guo, X., He, W., Ding, Y., Hong, B., Zhang, M., Wang, J., Jin, S., Zhou, E., Zheng, R., Fan, X., Wang, X., Xiong, L., Zhou, Y., Wang, W., Jiang, C., Zou, Y., Liu, X., … Gui, T. (2025). The rise and potential of large language model based agents: A survey. Science China Information Sciences, 68(2), 121101. https://doi.org/10.1007/s11432-024-4222-0

Yin, R. K. (2012). Case study methods. In APA handbook of research methods in psychology, Vol 2: Research designs: Quantitative, qualitative, neuropsychological, and biological (pp. 141–155). American Psychological Association. https://doi.org/10.1037/13620-009

Zhang, C. (Abigail), Thomas, C., & Vasarhelyi, M. A. (2022). Attended Process Automation in Audit: A Framework and A Demonstration. Journal of Information Systems, 36(2), 101–124. https://doi.org/10.2308/ISYS-2020-073

Appendices

Appendix 1 Interview guide

Haastattelurunko

Tämän haastattelun tarkoituksena on ymmärtää, miten prosessiautomaatiohankkeet etenevät käytännössä - erityisesti varhaisissa vaiheissa. Tarkoituksena ei ole arvioida yksittäisiä projekteja tai henkilöiden suoriutumista, vaan etsimme toistuvia toimintamalleja ja haasteita yhteistyössä prosessiautomaation sidosryhmien välillä

Tarkentavat kysymykset ovat erotettu pääkysymyksistä sisennyksellä ja (p) merkinnällä.

Haastateltavan taustatiedot ja rooli prosessiautomaatiohankkeissa

Mikä on tämänhetkinen roolisi prosessiautomaatiotiimissä?

Kuinka kauan olet työskennellyt tässä roolissa?

Millainen roolisi tyypillisesti on prosessiautomaatiohankkeiden eri vaiheissa (esim. esiselvitys, määrittely, kehitys, testaus, ylläpito)?

Automaatiohankkeen lähtöpiste

Kuvaile miten automaatioaloitteet tyypillisesti tulevat prosessiautomaatiotiimin tietoon?

Minkälaista tietoa prosessista on yleensä saatavilla tässä vaiheessa?

- (p) Kuinka vakiintunut tämä aloitusvaihe on?
- (p) Mitä tietoa useimmiten puuttuu?
- (p) Kuinka selkeä käsitys prosessista sinulla on tässä vaiheessa?

Prosessien ymmärtäminen

Kuvaile kuinka hyvin automatisoitavat prosessit ovat tyypillisesti ymmärrettyjä, kun kehitystyö alkaa?

Miten prosessitieto siirtyy liiketoiminnalta automaatiotiimille?

- (p) Onko prosessi dokumentoitu? Missä muodossa?
- (p) Mitkä osat prosessista jäävät usein epäselviksi tai implisiittisiksi?
- (p) Miten poikkeukset ja variaatiot tulevat esiin?

Oletko kohdannut tilanteita, joissa prosessin todellinen luonne selviää vasta kehitystyön aikana tai jälkeen?

Käytössä olevat artefaktit

Mitä dokumentteja, lomakkeita tai muita artefakteja käytetään prosessin kuvaamiseen ennen automaatiokehitystä?

(p) Kuka tuottaa nämä dokumentaatiot?

(p) Kuinka hyödyllisiä nämä dokumentaatiot ovat käytännössä?

(p) Ovatko ne enemmän muodollisuus vai todellinen työväline?

(p) Entä tilanteet, joissa dokumentaatioita ei käytännössä ole lainkaan – miten silloin toimitaan?

Prosessien soveltuvuuden arviointi

Miten yleensä arvioidaan, onko prosessi soveltuva automaatioon?

(p) Ketkä osallistuvat tähän arviointiin?

(p) Onko olemassa selkeää “no-go” -päätöstä?

(p) Mitä kriteerejä käytetään – tietoisesti tai tiedostamatta?

Missä vaiheessa mahdolliset ongelmat tai soveltumattomuus tyypillisesti havaitaan?

Onko sinulla kokemuksia tapauksista, joissa prosessi osoittautui huonoksi automaatiokohteeksi vasta myöhäisessä vaiheessa?

Vastuut ja odotukset

Miten vastuut jakautuvat liiketoiminnan ja automaatiotiimin välillä?

Kuka vastaa:

(p) prosessin määrittelystä?

(p) testitapausten luomisesta?

(p) automaation validoinnista?

Kuinka selkeitä liiketoiminnan odotukset ovat projektin alussa?

Missä kohtaa väärinymmärryksiä syntyy useimmiten?

Testaus ja myöhäinen prosessien selkiytyminen

Miten käyttäjätestaus (UAT) yleensä toteutetaan?

(p) Onko testitapaukset määritelty etukäteen?

(p) Tuleeko uusia prosessivaatimuksia esiin testauksen aikana?

Epäviralliset käytännöt

Kun muodolliset käytännöt eivät toimi tai niitä ei ole, miten tilanne yleensä ratkaistaan?

Onko syntynyt epävirallisia toimintatapoja, joilla työtä saadaan eteenpäin?

Yhteenveto ja reflektio

Mistä automaatiohankkeiden ongelmat useimmiten mielestäsi johtuvat?

Jos voisit parantaa yhtä asiaa automaatiohankkeiden varhaisvaiheessa, mikä se olisi?

Lopetus

Onko jotain oleellista, joka jäi käsittelemättä?

Onko olemassa automaatiohankkeita tukevia dokumentteja tai käytäntöjä, jotka ovat erityisen hyödyllisiä?

Appendix 2 Interview guide English translation

Interview guide

The purpose of this interview is to understand how process automation projects progress in practice, especially in their early stages. The aim is not to evaluate individual projects or people's performance, but rather to identify recurring patterns of action and challenges in collaboration between process automation stakeholders.

Follow-up questions are distinguished from the main questions by indentation and the marker (p).

Interviewee's background and role in process automation projects

What is your current role in the process automation team?

How long have you been working in this role?

What is your typical role in the different phases of process automation projects (e.g. pre-study, requirements definition, development, testing, maintenance)?

Starting point of an automation project

Describe how automation initiatives typically come to the attention of the process automation team.

What kind of information about the process is usually available at this stage?

- (p) How established or standardized is this initiation phase?
- (p) What information is most often missing?
- (p) How clear an understanding of the process do you have at this stage?

Understanding of processes

Describe how well the processes to be automated are typically understood when development work begins.

How is process knowledge transferred from the business side to the automation team?

(p) Is the process documented? In what form?

(p) Which parts of the process often remain unclear or implicit?

(p) How do exceptions and variations become visible?

Have you encountered situations where the true nature of the process only becomes clear during or after development?

Artefacts in use

What documents, forms, or other artefacts are used to describe the process before automation development begins?

(p) Who produces this documentation?

(p) How useful is this documentation in practice?

(p) Is it more of a formality or a real working tool?

What about situations where there is practically no documentation at all — how are those handled?

Assessment of process suitability

How is it usually assessed whether a process is suitable for automation?

Who participates in this assessment?

Is there a clear “no-go” decision?

What criteria are used — consciously or unconsciously?

At what stage are potential problems or unsuitability typically identified?

Do you have experience of cases where a process turned out to be a poor automation target only at a late stage?

Responsibilities and expectations

How are responsibilities divided between the business side and the automation team?

Who is responsible for:

(p) defining the process?

(p) creating test cases?

(p) validating the automation?

How clear are the business side’s expectations at the beginning of the project?

At what point do misunderstandings most often arise?

Testing and late clarification of processes

How is user acceptance testing (UAT) usually carried out?

(p) Are test cases defined in advance?

(p) Do new process requirements emerge during testing?

Informal practices

When formal practices do not work or do not exist, how is the situation usually resolved?

Have informal ways of working emerged that help move the work forward?

Summary and reflection

In your view, what are the most common causes of problems in automation projects?

If you could improve one thing in the early phase of automation projects, what would it be?

Closing

Is there anything essential that we have not covered?

Are there any documents or practices that support automation projects and are particularly useful?

Appendix 3 Explanation of the use of AI

Declaration on the Use of Artificial Intelligence (AI)

I have used generative artificial intelligence to support my thesis process at different stages and for different purposes. The tools I used, the purposes for which they were used, and the measures I took to verify the outputs produced by AI are described below. I also affirm that I have used AI tools with appropriate care, have disclosed their use in accordance with the applicable guidelines, and accept full responsibility for the content of this thesis in its entirety.

1. The AI tool used: Scopus AI

- **The stages in which the tool was used:**
  Thesis topic ideation and the search for source literature.
- **The purpose of use:**
  Scopus AI was used to quickly gain an overview of existing literature around the topics of this thesis and supplement the manual identification of relevant sources done by the researcher. The tool was not used to as generative AI to output any text, but rather as an AI powered search engine to find relevant research articles and broader clusters of research in

the Scopus database.

Example prompts:

"What has been researched about Robotic process automation (RPA)?"
(Prompted in October 2025)
"Is there research on the difference between centralized and decentralized RPA in In an organization?"
(Prompted in November 2025)

No personal data or any other sensitive information was processed with this AI tool.

- **Verification of results:**

  The Scopus AI tool outputs AI generated text with references to actual research articles. All articles chosen as sources for this study were manually read and evaluated by the researcher. This method of using Scopus AI to identify sources was used in a limited manner to avoid introducing bias from the AI tool.

2. The AI tool used: ChatGPT (GPT-5.1 & GPT-5.4)

- **The stages in which the tool was used:**

  Ideation on how to conduct the research.

  Rephrasing limited sections of the researchers own text and correcting grammar.

- **The purpose of use:**

  GPT 5.1 was used to "brainstorm" competing ideas on how to conduct research that answers the research questions at hand, what kind of theoretical approach to take and what kind of methodology supports the goals of the research. This was done to leverage the AI tool's capability to effortlessly present varying ideas to support the decision making of the researcher and saving the researcher's time. No personal data or any other sensitive information was processed with this AI tool.

  Example prompt:

  "[A verbal description of the research setting, research questions, available resources and limitations] Give me competing ideas of how to conduct a study like this. Provide pros, cons and any important considerations for each idea."
  (Prompted in November 2025)

  GPT 5.4 was used to rephrase text made by the researcher and correct grammatical errors in the researcher's text. Trying to adhere to the constraints of using practitioner and academic language structures and terminology in a non-native language can lead to complex sentence

structures with bad readability. An “outsider perspective” from an AI tool was used occasionally to improve the presentational clarity of the text and remove grammatical errors.

> *“[A section of text written by the researcher] Give me examples of how I can improve the clarity of this text without significant alterations to the informational content.”*
>
> *(Prompted in February 2026)*

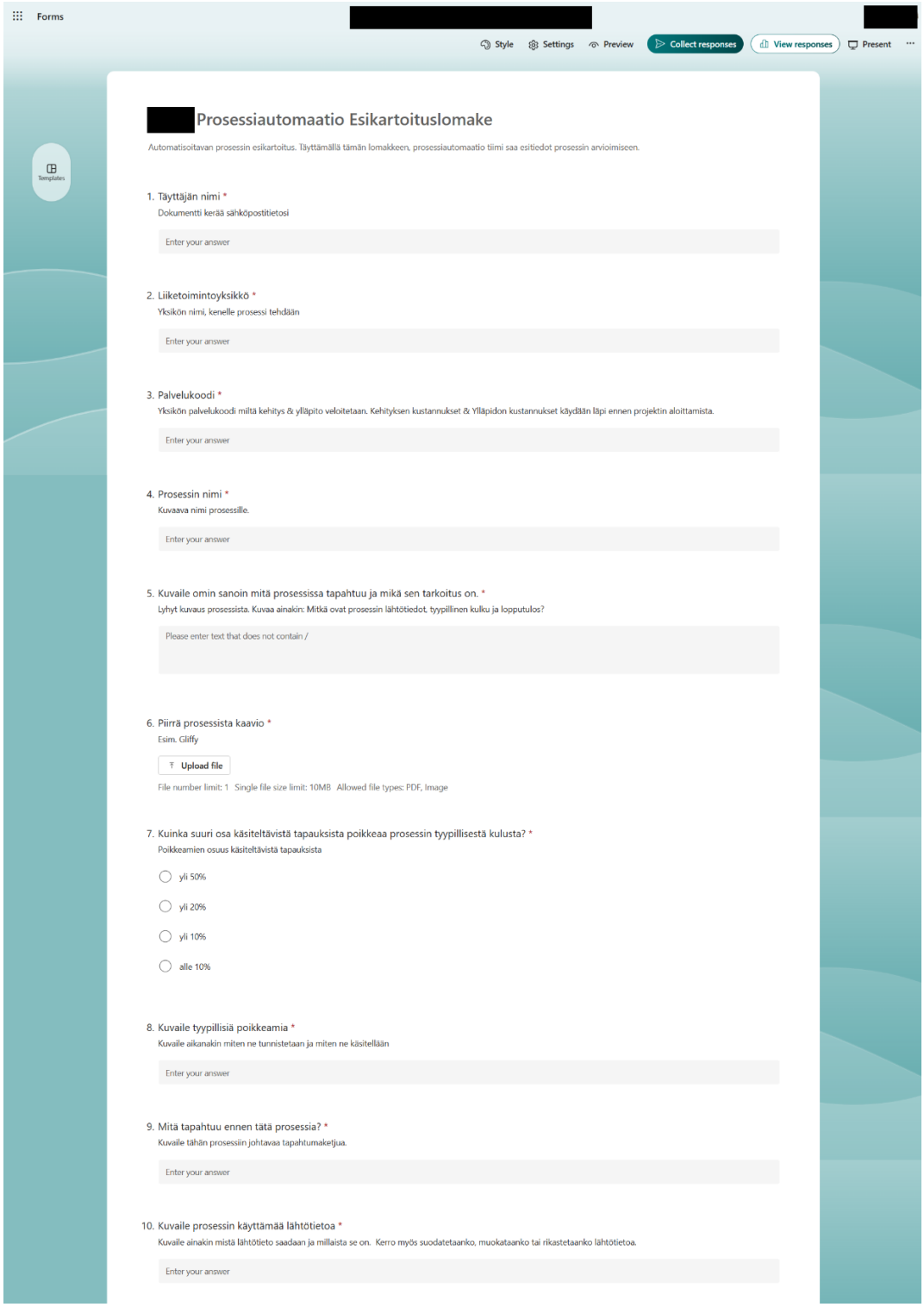
No personal data or any other sensitive information was processed with this AI tool.

- **Verification of results:**

  Any ideas on how to conduct the research provided by the AI tool were at no point considered reliable or taken at face value. They were used as preliminary ideas that needed to be verified from primary sources and evaluated by the researcher before being implemented in the study.

  All text rephrasing done with an AI tool was used on existing text of the researcher to avoid AI-made “ideas” from entering the study. All rephrasing was also validated by the researcher and very rarely taken “as is” from the AI tools response without the researcher's editing.

Appendix 4 Pre-assessment form



11. Mitä tapahtuu tämän prosessin jälkeen? *

Kuvaile tätä prosessia seuraavaa tapahtumaketjua.

Enter your answer

12. Onko tämä jo olemassa oleva manuaalinen prosessi vai kokonaan uusi? *

Voit kirjoittaa (Kyllä / Ei / En tiedä) + mahdolliset lisätiedot omin sanoin.

Enter your answer

13. Onko prosessista olemassa työojetta? *

Jos sellainen löytyy, lataa tiedosto tähän. Työohje helpottaa prosessin määrittelyä huomattavasti.

Upload file

File number limit: 1 Single file size limit: 100MB Allowed file types: Word, Excel, PPT, PDF, Image, Video, Audio

14. Prosessin omistaja *

Prosessin omistaja, kuka siunaa mm. kustannukset

Enter your answer

15. Kerro prosessin asiantuntijat *

Henkilöt jotka tuntevat prosessin / tekevät sitä manuaalisesti / ovat muuten tekemisissä prosessin kanssa. Tarvitsemme näitä henkilöitä prosessin määrittelyyn ja myöhemmin automaation validointiin.

Enter your answer

16. Käytetyt järjestelmät *

erota järjestelmät puolipilkulla

Enter your answer

17. Onko kohdejärjestelmissä testipuolta sekä testidataa saatavilla? *

Testimateriaali ja testijärjestelmät helpottavat kehitystyötä huomattavasti

- ◯ Kyllä
- ◯ Osassa järjestelmiä
- ◯ Ei
- ◯ En osaa sanoa

18. Kuka vastaa testitapausten luomisesta *

sähköpostiosoite keneen ottaa yhteyttä testitapausten luomiseen liittyen

Enter your answer

19. Kuka luvittaa järjestelmiin käyttöoikeudet? *

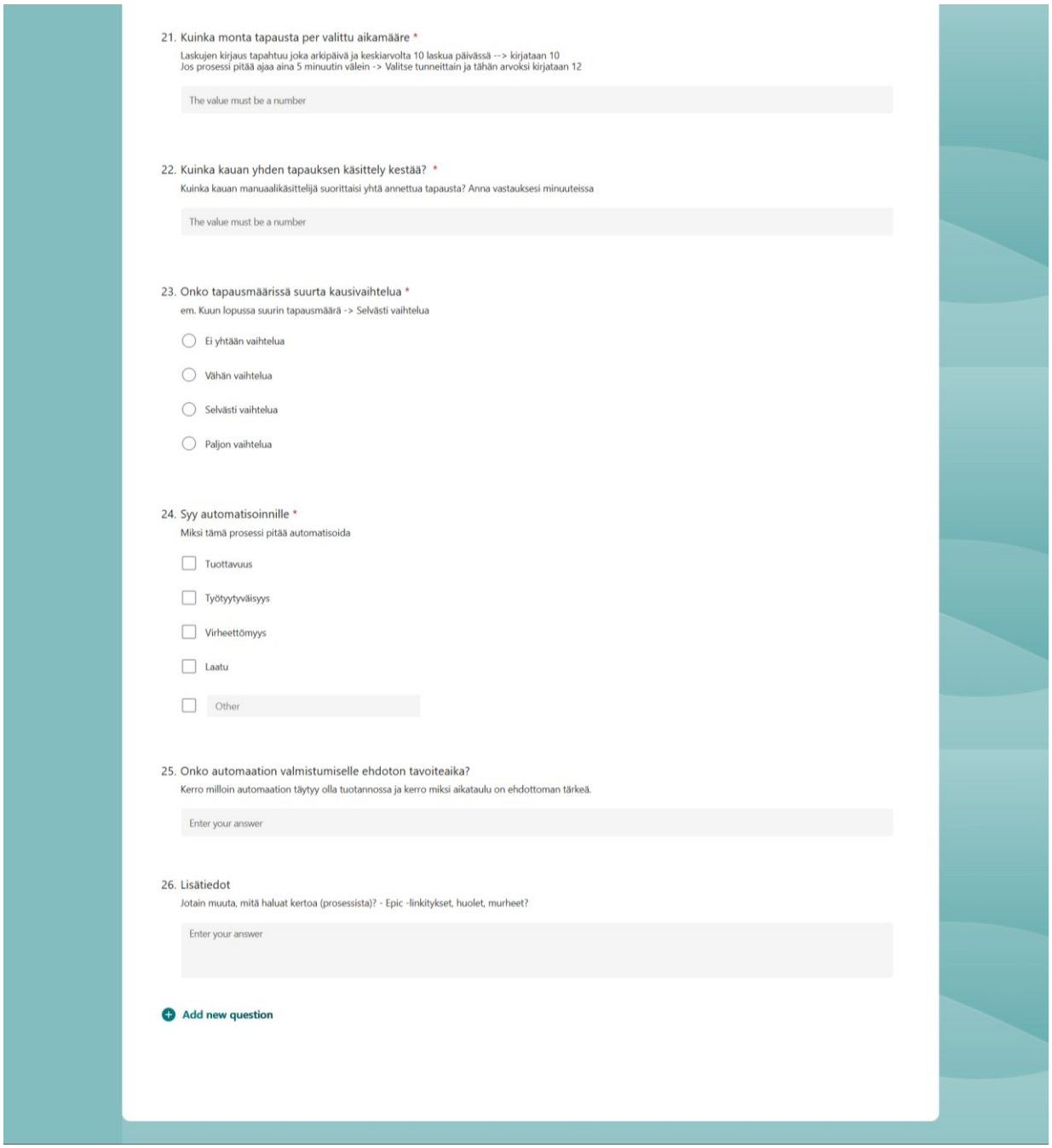
Taho / tahot keneltä saa apua käyttöoikeuksien luvituksiin

Enter your answer

20. Kuinka usein prosessin tehtävät pitää suorittaa *

anna Frekvenssi, kuinka usein prosessin pitää lähteä liikkeelle

- ◯ Tunneittain
- ◯ Päivittäin
- ◯ Joka arkipäivä
- ◯ Viikoittain
- ◯ Kuukausittain
- ◯ Vuosittain



21. Kuinka monta tapausta per valittu aikamääre *

Laskujen kirjaus tapahtuu joka arkipäivä ja keskiarvolta 10 laskua päivässä --> kirjataan 10
Jos prosessi pitää ajaa aina 5 minuutin välein -> Valitse tunneittain ja tähän arvoksi kirjataan 12

The value must be a number

22. Kuinka kauan yhden tapauksen käsittely kestää? *

Kuinka kauan manuaalikäsittelijä suorittaisi yhtä annettua tapausta? Anna vastauksesi minuuteissa

The value must be a number

23. Onko tapausmäärissä suurta kausivaihtelua *

em. Kuun lopussa suurin tapausmäärä -> Selvästi vaihtelua

- ○ Ei yhtään vaihtelua
- ○ Vähän vaihtelua
- ○ Selvästi vaihtelua
- ○ Paljon vaihtelua

24. Syy automatisoinnille *

Miksi tämä prosessi pitää automatisoida

- ☐ Tuottavuus
- ☐ Työtyytyväisyys
- ☐ Virheettömyys
- ☐ Laatu
- ☐ Other

25. Onko automaation valmistumiselle ehdoton tavoiteaika?

Kerro milloin automaation täytyy olla tuotannossa ja kerro miksi aikataulu on ehdottoman tärkeä.

Enter your answer

26. Lisätiedot

Jotain muuta, mitä haluat kertoa (prosessista)? - Epic -linkitykset, huolet, murheet?

Enter your answer

⊕ Add new question

Appendix 5 Shared specification document

Blank Process specification template

Links to relevant resources:

Business unit guide to process automation development cycle

Dictionary of proprietary terminology

General information

Key personnel from process automation team

Process automation team fills this table with developers, analysts and any other resources appointed to this project.

Name	Email	Role	Area of expertise	Core responsibility in the project	Approval to proceed to the next development cycle phase (joint specification meetings)/(development)/(production)
[redacted]	[redacted]	Developer	Technical implementation	Developing the automation according to process specification	
[redacted]	[redacted]	Analyst	Methodological expertise in process specification	Methodological support for defining the process and finalizing the process specification	

Key personnel from business unit(s)

Lead process expert fills this table with all relevant subject matter experts, decision makers and other stakeholders, whose involvement benefits this automation project.

Name	Email	Role	Area of expertise	Core responsibility in the project	Approval to proceed to the next development cycle phase (joint specification meetings)/(development)/(production)
[redacted]	[redacted]	Process owner	Business unit strategy, business unit OKR's & budget	Provide approval for the automation initiative	
[redacted]	[redacted]	Lead process expert	Subject matter of the process	Responsible for defining the process, validating process specification and validating the automation.	
[redacted]	[redacted]	Process expert	Subject matter in a specific area of process	Responsible for defining their area of expertise	
[redacted]	[redacted]	Process expert	Subject matter in another specific area of process	Responsible for defining their area of expertise	
[redacted]	[redacted]	Target system expert	Target enterprise system	Access rights, test environment, test data	

High-level description of the process

{Insert answer 5 from pre-assessment form}

{Insert workflow chart from answer 6 of pre-assessment form}

Known uncertainties in the process description

Is there something about the process that is unclear at this point and needs to be determined later?

Are there uncertainties of the process that need to be clarified in practice as automation development progresses?

Process maturity and automation suitability

Is this process already executed in a manual form or is this a brand new process?

{Insert answer 12 from pre-assessment form}

{Insert answer 13 file from pre-assessment form}

How long has the manual process been in place?

Has the process workflow changed during that time?

Is the manual process executed by more than one person?

Does the process workflow vary based on the preference of who executes it?

If the process is brand new, what created the need to automate it now?

Process context

Describe the workflow that leads up to the execution of this process?

{Insert answer 9 from pre-assessment form}

Describe the workflow that follows the execution of this process?

{Insert answer 11 from pre-assessment form}

Benefits of automation

Why process should be automated? Costs, quality, etc?

{Insert answer 24 from pre-assessment form}

Target Systems

System name	System description	System special requirements	System preparations /-setting	Requires installation
[redacted]	Financial ERP system	<Developer / Analyst fills this in>	<Developer / Analyst fills this in>	<Developer / Analyst fills this in>
			<Developer / Analyst fills this in>	

Detailed Process Definition

<Detailed description of the process. Step by step guide with pictures>

Producer

Consumer

Business exceptions

	Exception reason	Exception definition	Robot Actions	Business actions required
Invoice already accepted	Invoice already accepted. Will not be handled by robot	Adds to exception report?	No actions	
Invoice status is rejected	Invoice is already rejected	Exception reason, notify business if exception raised	Check invoice status manually	

Business rules

<Business rules are maintained by business. Business rules are used to control robot.>

Rule	Where the rules are maintained	Column control rule (Sarakkeen ohjaussääntö)	Additional information
<Companies to skip>	Path: [redacted] 000-ohjausexcel.xlsx> Sheet: <ohitettavat>	• <A-Column: Company ID/all> • <B-Column: Company name/all> • <C-Column: Company y-tunnus/all>	If single company → Skip If "all" → Skip all

Process scheduling & Reporting requirements

Business requirements

<When, how, how often & business SLA>

Question	Answer	Additional information
When	<Every day, every weekday>	<Not on holidays>
How often	<Hourly,Daily, Weekly,Monthly>	<System X has service break daily between 01-03>
Latest run time	<Process should be ran at latest: 15:00>	

Process scheduling in Control Room

<Example of schedule & restriction of schedule etc. Describe here how schedule should be done, so that it works autonomously. >

Process reporting requirements

<what to report, how often, whom?>

Question	Answer	Additional information
Location of the report (Where is report saved)	<path>	
To whom report is sent?	<Emails>	<ei sähköposti-ilmoituksia, pelkkä tiedoston tallennus>
How often?	<daily, once a week>	
Who is responsible of the business exceptions?	[redacted]	Business exception needs to be checked manually. Check the business exception list: **<add link to business exceptions>**
Special requirements	• <Process termination should be informed immediately> • <Exceptions should be informed immediately > • <Report only business exceptions>	

Example report/email

Item Key	Last updated	Status	Tags	Exception Reason
12345	1.1.2020 00:30:00	Completed	Tag 3	
98765	1.1.2020 01:00:00	Business Exception	Tag 1, Tag 2	Invoice already handled

Process description

Workspaces

Workspace name	Additional info
XXX	
XXX	

Tasks

Task name	GitLab repository	Additional information	Requires rdp	AD for login / runnergroup	List of needed ENV_VARIABLES
pxxx-robotname-producer			Yes/no	[redacted]	
pxxx-robotname-consumer					

Work Items

Key	Value

Process dependencies

<Describe process dependencies here (Gliffy diagram or verbal). Which process has dependencies with other processes etc. Describe complex structures verbally if needed.>

Process vaults

<Define here which vaults robot is using>

<Add Vault name Here>

Key	Value
[redacted]	[redacted]
[redacted]	[redacted]

Process assets

<Define here if robot is using assets>

Asset Name	Description

Process integrations

<Define here if robot is using process specific Control Room API keys or webhooks>

Environment variables

Variable purpose	Variable name	Value in Prod	Valuen in test
Robot busines rules document path	ENV - RPA-000 - Säännöstö	[redacted]	[redacted]
Robot reporting path	ENV - RPA-000 - Raportti	[redacted]	[redacted]
Aineistosijainti	ENV - RPA-000 - Aineisto	[redacted]	[redacted]