



**DHBW**

Duale Hochschule  
Baden-Württemberg  
**Ravensburg**

## **Multi-Agent Architecture Design**

Workflow Automation in Compliance  
with Operational Excellence

### **Bachelor's Thesis**

Wirtschaftsinformatik—Business Engineering  
Duale Hochschule Baden-Württemberg  
Ravensburg

Francisco Rodriguez Müller, on September 16, 2025  
Mat. Nr.: 2775857, Course: RV-WWIBE122  
Supervisor: Prof. Dr. Paul Kirchberg

# Declaration of Authenticity

I hereby declare that I have written the present bachelor's thesis independently and that I have not used any sources or aids other than those indicated. All passages that were taken from published or unpublished works are clearly marked as such. This thesis has not been submitted to any other examination authority in the same or a similar form.

Thesis Title:

**Multi-Agent Architecture Design**

Workflow Automation in Compliance with Operational Excellence

Ravensburg, September 16, 2025

---

*(Place and Date)*

---

*(Signature)*

## Acknowledgments

I would like to thank the *All for One Group SE* for providing the opportunity and resources to write this thesis. I also extend my gratitude to my academic supervisor, *Prof. Dr. Paul Kirchberg* for his guidance throughout the process; my professional advisor, *Emily Celen* for helping me with the scope and realistic expectations for this work; and my direct supervisor and friend *Ralph Thomaßen* for his wise advice and relentless support throughout my academic journey.

## Abstract

[Placeholder, still a work in progress] The literature suggests three design-relevant insights: (1) OpEx increasingly hinges on adaptability, decision quality, and compliance—not only efficiency; (2) automation is moving from deterministic scripts (RPA) to adaptive, tool-using agents; and (3) MAS provide a natural organizing structure for such agents in complex enterprises but necessitate robust governance, monitoring, and integration. These insights inform the requirements (e.g., traceability, policy constraints, observability) and the architecture choices (agent roles, interaction patterns, orchestration) developed later in this thesis.

# Contents

<b>Acknowledgments</b>	<b>ii</b>
<b>Abstract</b>	<b>iii</b>
<b>1 Introduction</b>	<b>1</b>
<b>2 Methodology</b>	<b>2</b>
2.1 Qualitative Content Analysis . . . . .	3
2.2 Requirements Engineering & Systems Analysis . . . . .	4
2.3 Information Systems Design . . . . .	5
<b>3 Literature Review</b>	<b>6</b>
3.1 Operational Excellence . . . . .	6
3.2 Workflow Automation . . . . .	8
3.3 Agentic Artificial Intelligence . . . . .	9
<b>4 Modeling the Requirements</b>	<b>10</b>
4.1 Requirements Clustering & Consolidation . . . . .	10
4.2 Requirement Reformulation & Classification . . . . .	12
4.3 Model-Based Requirements Representation . . . . .	15
<b>5 Modeling the Architecture</b>	<b>15</b>
5.1 [placeholder]Modeling the Agents . . . . .	15
5.2 [placeholder]Modeling the Architecture . . . . .	15
5.3 [placeholder]Modeling the Interactions . . . . .	15
<b>6 Discussion: Applicability Criteria</b>	<b>15</b>
<b>7 Conclusion</b>	<b>15</b>
<b>Appendix A: Full SysML Models</b>	<b>16</b>

---

## Abbreviations

DSR .....	Design Science Research
GaaS .....	Governance as a Service
GenAI .....	Generative Artificial Intelligence
IPA .....	Intelligent Process Automation
ISD .....	Information Science Design
LLM .....	Large Language Model
MAS .....	Multi-Agent System
MBSE .....	Model-Based Software Engineering
OpEx .....	Operational Excellence
QCA .....	Qualitative Content Analysis
RE .....	Requirements Engineering
ReAct .....	Reason & Action
RPA .....	Robotic Process Automation
SysML .....	Systems Modeling Language
WfMs .....	Workflow Management Systems

# 1 Introduction

Organizations across industries continue to face persistent challenges in achieving operational excellence (OpEx). Fragmented processes, manual interventions, and inconsistent data quality undermine efficiency and decision-making. Legacy workflows and siloed systems exacerbate these inefficiencies, while traditional automation approaches often lack the adaptability needed in dynamic business environments. For companies, this translates into slower response times, higher compliance risks, and limited scalability—issues that directly threaten competitiveness.

Agentic AI, building on the advances of generative artificial intelligence (GenAI), opens new possibilities to extend automation beyond deterministic scripts. While GenAI provides the cognitive and generative capabilities, agentic AI leverages these to create adaptive, tool-using agents that can plan, act, and coordinate—thereby supporting governance, decision quality, and organizational agility. Despite this potential, both practice and academic literature lack structured strategies and conceptual frameworks for embedding such agentic capabilities into operational workflows in a scalable and value-driven way. This gap motivates the present research.

In this context, multi-agent systems (MAS) can serve as a reference architecture for integrating GenAI-enabled agentic AI into enterprise workflow automation. The central research question is:

*How can a MAS architecture be designed to integrate GenAI capabilities into workflow automation, in order to enhance agility, compliance, & decision quality to achieve OpEx?*

To answer this question, the study addresses the following sub-questions:

- *Which design requirements are necessary to align a multi-agent architecture with the goals of OpEx?*
- *How should a MAS be architected to fulfill these requirements?*
- *Under which conditions is deploying a generative multi-agent architecture justified over traditional automation approaches?*

Methodologically, the thesis applies Design Science Research (DSR) to develop a conceptual reference architecture. The approach synthesizes requirements from academic literature and OpEx principles, models agent roles and interactions, and derives applicability conditions for real-world deployment. Although the architecture is designed to remain industry-agnostic, a use case from the financial services

sector is introduced to illustrate how the conceptual model can be instantiated in a regulated, legacy-intensive environment.

The core contribution of this work is a conceptual design of a MAS that leverages GenAI to support OpEx in enterprise workflows. Specifically, it delivers:

- *A structured synthesis of system requirements derived from academic literature and OpEx principles.*
- *A conceptual architecture detailing agent roles, interactions, and integration points.*
- *A set of applicability conditions and design considerations to guide future deployment and evaluation of generative multi-agent architectures in practice.*

The scope is limited to conceptual design; formal evaluation and technical implementation are proposed as future work. Although the architecture is designed to remain industry-agnostic, a use case from the financial services sector is introduced to illustrate how the conceptual model can be instantiated in a regulated, legacy-intensive environment.

After this introduction in Section 1, the thesis is structured as follows: Section 2 outlines the research methodology, including the use of Design Science Research (DSR) and supporting methods. Section 3 presents a literature review on operational excellence, workflow automation, and agentic AI. Section 4 details the synthesis and modeling of requirements. Section 5 develops the conceptual multi-agent architecture. Section 6 discusses the applicability of MAS in workflow automation use cases, and Section 7 concludes with reflections and directions for future research.

## 2 Methodology

This thesis applies DSR methodology to create a conceptual artifact—a multi-agent architecture for workflow automation. Practically, the approach unfolded in three steps: (1) *reviewing the literature* on OpEx, workflow automation, and agentic AI; (2) *deriving and structuring requirements* from literature and case material into a requirements model; and (3) *designing a conceptual system architecture* using Systems Modeling Language (SysML).

Supporting methods included Mayring-style qualitative content analysis (QCA) for the review, requirements engineering (RE) and systems analysis for the requirements model, and information systems design (ISD) to structure the architecture and



ensure requirement-to-design traceability, supported by SysML modeling practices from Model-Based Systems Engineering (MBSE). Within DSR, the work focuses on problem identification, objective definition, and conceptual design, while instantiation/demonstration and formal evaluation are out of scope given the bachelor-thesis format and resource constraints. This scoping maintains methodological rigor while keeping the contribution focused: a well-argued reference architecture ready for subsequent implementation and empirical evaluation.

## 2.1 Qualitative Content Analysis

To ensure a systematic and structured literature review, this thesis employed QCA following the principles of Mayring and Fenzl (2022). As a rule-based method for synthesizing insights from textual sources, QCA was used within the DSR framework to support the problem identification and objective definition phases (Hevner et al. 2004; Peffers et al. 2007). In this thesis, it was applied in a literature-focused manner to structure the review and provide a traceable basis for subsequent RE.

The analysis was scoped along three dimensions. The *analysis unit* was defined as the overall body of literature addressing operational excellence, workflow automation, and agentic AI. The *context unit* consisted of individual publications (books, peer-reviewed articles, industry reports, standards). The *coding unit* was defined as discrete statements or conceptual claims relevant to the intersection of OpEx, automation paradigms, and AI-based multi-agent systems.

A mixed deductive-inductive approach was used. Deductive categories were derived from established theory, including OpEx dimensions such as adaptability, compliance, and decision quality, as well as prior automation frameworks (e.g., Robotic Process Automation, RPA; Intelligent Process Automation, IPA). Inductive categories emerged from the material itself, capturing issues highlighted repeatedly in the sources, such as observability, traceability, and governance in agentic AI. This balance ensured that both established and novel concerns were systematically reflected.

The outcome of this categorization was not a formal codebook, but a set of thematic clusters that guided the narrative structure of Section 3. Each subsection of the literature review is organized around these categories, which in turn serve as the input for the elicitation lists presented at the beginning of Section 4. In this way, QCA provides both a conceptual ordering of the literature and a direct bridge

into the requirements engineering process.

## 2.2 Requirements Engineering & Systems Analysis

Following the IEEE (1990) definition, a requirement is a *condition or capability needed by a user to solve a problem or achieve an objective*. RE provides the systematic means to derive such objectives. In this thesis, RE was applied in the early phases to ensure that the conceptual architecture rests on precise, validated needs rather than general aspirations.

The synthesis of requirements followed a structured but literature-driven process. Recurring design concerns were identified from the results of the qualitative content analysis (QCA), documented in *elicitation lists*, and then consolidated into a unified set of requirement candidates. Consistent with Glinz et al. (2020), each candidate was reformulated into an atomic, unambiguous, and verifiable “shall” statement and classified into *functional requirements* (system behaviors), *quality requirements* (non-functional attributes such as performance or compliance), or *constraints* (technological or regulatory limits).

While this procedure draws on the phases described by Herrmann (2022) (elicitation, documentation, analysis, management), it was adapted to the scope of this thesis: instead of stakeholder workshops, the primary elicitation source was the systematically coded literature. To situate the requirements, a complementary systems analysis defined the system boundary, identified stakeholders and external actors, and clarified interface obligations—helping prevent scope creep and omissions.

Requirements were then represented in SysML requirement diagrams. Trace links connect each documented requirement to the respective architecture elements, enabling full requirement-to-design traceability via *«satisfy»* and *«verify»* relationships. This model-based approach ensures that design decisions can always be traced back to validated needs and that no requirement was overlooked.

In summary, the integration of RE and systems analysis provided a structured, traceable, and quality-assured requirement set. This foundation anchors the subsequent conceptual architecture in rigorously defined objectives, ensuring consistency with both DSR methodology and operational excellence goals.

## 2.3 Information Systems Design

In line with DSR, this thesis applies principles of information systems design to structure the artifact. The architecture was modeled in SysML, making use of MBSE practices to ensure requirement-to-design traceability. While originating in MBSE originates in systems engineering, its modeling discipline is transferable to information systems contexts and supports the systematic development of conceptual architectures.

A conceptual architecture was systematically developed based on the previously synthesized requirements using a MBSE approach. MBSE provides a formalized way to transform requirements into a rigorous system model and to validate the design at the conceptual level. In fact, MBSE is defined as the formal application of modeling to support system requirements, design, and analysis, along with verification and validation, beginning in the conceptual design phase. Adopting MBSE thus ensured that even at this early stage, the architecture could be checked against stakeholder needs and constraints. The methodology enabled a unified representation of the entire system that made it possible to study interactions among components and agents before implementation. Each requirement was traced to corresponding elements in the model (e.g. agent roles, interactions, or policies), guaranteeing requirements-to-design traceability and consistency throughout the design process.

The conceptual architecture was captured as a SysML v2 model (the latest OMG Systems Modeling Language standard for systems modeling) leveraging a plain-text modeling workflow in Eclipse Systems Mode. This setup allowed the use of SysMLv2's textual notation to define the system's structure and behavior in a tool-agnostic, version-controlled manner. The choice of SysMLv2 (over SysMLv1 or informal diagrams) is justified by its improved expressiveness and alignment with current MBSE best practices; as an OMG-developed language it provides robust semantics for specifying complex interactions in MAS while remaining an emerging industry standard. In summary, the information system's design was conducted in a model-driven fashion: the SysML v2 conceptual model served as an executable blueprint of the architecture, facilitating early validation of design decisions against the requirements and providing a solid foundation for subsequent development steps (cf. Madni 2023; OMG 2023).

### 3 Literature Review

The literature review is organized into three subsections that together frame the problem context of this thesis. It begins with operational excellence, which defines the strategic objectives—adaptability, compliance, decision quality—that guide enterprise transformation efforts. The second subsection addresses workflow automation, as the established technological approach for operationalizing these objectives in practice. The third subsection turns to agentic AI, a rapidly emerging paradigm that extends automation beyond deterministic scripts toward adaptive, tool-using agents. This sequence—objectives, established solutions, emerging solutions—provides a logical progression from strategic goals to current practice and then to prospective innovations. It ensures that the requirements synthesized in Section 4 are grounded in both enduring management principles and the latest technological developments relevant to workflow design.

#### 3.1 Operational Excellence

OpEx originated as a management philosophy in the manufacturing sector, particularly in the automotive industry to optimize quality and efficiency. In this classical context, OpEx focused on minimizing defects, eliminating waste, and embedding continuous improvement practices into organizational routines (Juran and Godfrey 1999; Womack and Jones 2013). While these roots remain important, they provide only a partial foundation for understanding OpEx in today’s IT-driven enterprises, which operate in volatile environments shaped by rapid technological change, regulatory complexity, and global competition.

One central dimension of OpEx is ADAPTABILITY AND AGILITY. In IT-driven firms, where OpEx is defined less by physical production flows and more by the ability to execute strategies effectively while maintaining innovation, adaptability refers to the capacity of processes to be reconfigured in response to volatility, ensuring resilience in dynamic environments. Agility emphasizes change-readiness and rapid adaptation, qualities increasingly recognized as indicators of organizational excellence in globalized and unpredictable markets. While adaptability enhances resilience, it can reduce efficiency if frequent changes disrupt standardization; conversely, a strong focus on efficiency can make processes rigid and less responsive to unexpected events. In practice, organizations must reconcile continuous improvement with agility, balancing stability and flexibility in their operational routines

(Carvalho et al. 2023).

Another recurring theme in the literature is COMPLIANCE AND RISK MANAGEMENT. OpEx in regulated industries demands that workflows embed mechanisms for ensuring transparency, auditability, and regulatory adherence (Owoade et al. 2024). Compliance safeguards minimize operational risk but can introduce bureaucratic overhead and slow decision-making. The key challenge is balancing strict rule enforcement with the flexibility needed to respond to novel business conditions, a tension especially visible in digital service environments with evolving legal frameworks (Juran and Godfrey 1999).

A further category is DECISION QUALITY. A core aim of OpEx is not only to accelerate decision-making but to improve its reliability and evidential grounding (Owoade et al. 2024). Automation can help by aggregating relevant data and reducing errors, but it also risks opacity and overconfidence when human oversight is limited. The central trade-off is speed versus quality: excessive automation may produce faster but less accountable outcomes, while excessive oversight slows operations. For sustainable excellence, systems must therefore support evidence-based choices and make decision pathways transparent.

Another key dimension is EFFICIENCY AND CONTINUOUS IMPROVEMENT. Rooted in Lean and TQM traditions, efficiency emphasizes minimizing waste and reducing manual effort, while continuous improvement institutionalizes iterative refinements in processes (Juran and Godfrey 1999; Womack and Jones 2013). Together, these principles enhance operational reliability and cost-effectiveness, yet they can conflict with the need for flexibility in volatile environments. The challenge is to design workflows that are optimized for today while remaining adaptable for tomorrow.

Customer-facing outcomes are captured by CUSTOMER-CENTRICITY. OpEx emphasizes that processes must be aligned with user needs and service-level commitments (Womack and Jones 2013; Juran and Godfrey 1999), ensuring reliability, responsiveness, and satisfaction. A strong customer focus can create pressure to customize and accelerate processes, which may undermine efficiency or compliance. The literature stresses that sustainable excellence requires balancing external demands with internal consistency, often formalized through service-level agreements (SLAs).

The category of USER EMPOWERMENT AND CULTURE recognizes that operational excellence is not solely technical but also organizational. Effective improvement requires systems that support transparency, collaboration, and employee en-

agement (Womack and Jones 2013). Empowerment fosters ownership and participation, but decentralizing authority can introduce inconsistency and conflict with standardization goals. Culture therefore functions as both an enabler and a constraint for process excellence, shaping how automation is accepted and leveraged by human actors (Juran and Godfrey 1999).

Finally, TECHNOLOGY INTEGRATION AND SCALABILITY reflects the increasing role of digital platforms in enabling OpEx. Modern enterprises rely on architectures that integrate automation, AI, and cloud services to achieve scalable and resilient operations (Owoade et al. 2024). Integration enables end-to-end process coverage and agility, but also increases dependency on heterogeneous systems and external vendors. Scalability promises growth and innovation, yet without careful governance it can amplify complexity and risk.

### 3.2 Workflow Automation

One foundational aspect of workflow automation is PROCESS ORCHESTRATION. Orchestration denotes the centralized coordination of tasks and activities according to a defined process logic. In a typical WfMS, a workflow engine enacts the process model, dispatching tasks to the right resources (human or machine) in the correct sequence and enforcing the business rules at each step (Basu and Kumar 2002). This engine-driven coordination brings predictability and repeatability to workflows: tasks are executed in a fixed, optimized order with minimal ad-hoc variation. By systematically controlling task flow, early workflow systems could eliminate many manual hand-offs and delays, thereby boosting efficiency and consistency in outcomes (Stohr and Zhao 2001).

Another central category is INTEGRATION AND INTEROPERABILITY. Workflow automation must seamlessly connect heterogeneous applications, data sources, and organizational boundaries. Integration reduces fragmentation and enables end-to-end process visibility, but it also increases complexity when systems evolve independently. Research emphasizes that successful workflow deployment hinges on robust interfaces and interoperability standards (Stohr and Zhao 2001).

The concept of MODULARITY AND REUSABILITY emphasizes that workflows should be constructed from building blocks that can be reused and recombined. Modularity allows subprocesses to be adapted to different contexts with minimal effort, reducing engineering overhead. Reusability also supports organizational learn-

ing, since well-tested workflow fragments can be applied across multiple processes. Literature highlights this as a key factor in making workflow automation scalable and maintainable (Stohr and Zhao 2001).

A further dimension is EXCEPTION HANDLING AND FLEXIBILITY. Real-world processes inevitably deviate from their modeled paths, whether due to missing information, unexpected events, or external disruptions. A workflow system must detect such deviations and either resolve them automatically or escalate them appropriately (Basu and Kumar 2002). Flexibility ensures resilience, but excessive exception handling can increase system complexity and reduce predictability.

Finally, WORKFLOW GOVERNANCE concerns the mechanisms for ensuring accountability, compliance, and control in automated processes. Monitoring, audit trails, and role-based access controls allow organizations to supervise workflows and detect irregularities. Governance strengthens trust and regulatory compliance, but it can also add bureaucratic layers that reduce agility (Owoade et al. 2024).

### 3.3 Agentic Artificial Intelligence

A defining property of agentic AI is AUTONOMY IN DECISION-MAKING. Agents can operate independently within clearly scoped authority, making local decisions without constant human supervision. Autonomy increases scalability and agility but introduces risks of misalignment or unintended behavior if oversight is weak (**shapiroFoundations2023**).

Another core capability is TOOL USE AND INTEGRATION. Modern agents increasingly rely on external tools, APIs, and services to extend their functionality. This requires robust interfaces and safeguards to prevent errors or malicious use. Research on tool-augmented AI demonstrates both the promise and the complexity of integrating agents into real enterprise environments (**schickToolformer2023**).

Multi-agent systems further enable COORDINATION AND SPECIALIZATION. By dividing labor into roles, agents can focus on specialized tasks while coordinating outcomes through explicit communication protocols. Coordination increases overall efficiency but raises new challenges of synchronization, conflict resolution, and overhead in negotiation (**wooldridgeIntroduction2009**).

The category of OBSERVABILITY AND TRANSPARENCY addresses the need for explainable and auditable agent behavior. All agent actions and decisions should be logged, and reasoning pathways must be accessible for debugging and compliance

checks. Transparency fosters trust but can conflict with performance and efficiency if excessive instrumentation is required (Owoade et al. 2024).

Finally, GOVERNANCE AND COMPLIANCE are essential to align agent behavior with organizational and ethical standards. Oversight mechanisms, including policy enforcement layers and human-in-the-loop checkpoints, provide guardrails against undesirable outcomes. Such governance ensures accountability, though it may reduce the autonomy and speed that make agentic AI attractive (Owoade et al. 2024).

## 4 Modeling the Requirements

The literature review identified recurring design concerns across operational excellence, workflow automation, and agentic AI. Synthesizing these insights yields *elicitation lists* which represent the initial outcome of requirements documentation based on systematically coded sources. A subsequent *clustering* step consolidated overlapping items into a unified set of requirement candidates.

Each candidate was then reformulated into an atomic, unambiguous, and verifiable “shall” statement, following the best-practice formulation rules of Glinz et al. (2020). The final requirements were organized into *functional requirements*, *quality requirements*, and *constraints*, in line with the Glinz taxonomy.

Requirements were then represented in SysML v2 as dedicated requirement elements, with their textual statements captured in the description field. Trace links connect each requirement to its source in the elicitation lists and to the architecture elements that *«satisfy»* it. This model-based representation ensures that design decisions remain traceable to validated needs and that requirement coverage can later be verified systematically.

### 4.1 Requirements Clustering & Consolidation

#### O — OPERATIONAL EXCELLENCE

1. ADAPTABILITY AND AGILITY — processes must remain reconfigurable in response to volatile conditions, ensuring resilience in dynamic environments.
2. COMPLIANCE AND RISK MANAGEMENT — regulatory adherence and transparency must be embedded into workflows to minimize compliance risks.



3. **DECISION QUALITY** — automation should enable data-driven, timely, and well-informed decisions rather than simply increasing speed.
4. **EFFICIENCY AND CONTINUOUS IMPROVEMENT** — workflows should reduce manual effort, eliminate waste, and institutionalize iterative refinements.
5. **CUSTOMER-CENTRICITY** — operations must align with user needs and service-level commitments to sustain value delivery.
6. **USER EMPOWERMENT AND CULTURE** — systems should support collaboration, transparency, and employee engagement in improvement processes.
7. **TECHNOLOGY INTEGRATION AND SCALABILITY** — architectures must accommodate automation, AI, and cloud services to enable sustainable innovation.

#### W — WORKFLOW AUTOMATION

1. **PROCESS ORCHESTRATION** — workflow engines must enforce task sequences and business rules to guarantee reliable execution.
2. **INTEGRATION AND INTEROPERABILITY** — automation must seamlessly connect heterogeneous applications, data sources, and organizational boundaries.
3. **MODULARITY AND REUSABILITY** — workflows should be composed of modular tasks or subprocesses that can be reused and reconfigured with minimal effort.
4. **EXCEPTION HANDLING AND FLEXIBILITY** — systems must detect, manage, and escalate deviations rather than failing in unforeseen scenarios.
5. **WORKFLOW GOVERNANCE** — monitoring, audit trails, and role-based controls must ensure accountability and compliance throughout automated processes.

#### A — AGENTIC AI

1. **AUTONOMY IN DECISION-MAKING** — agents should operate independently within clearly scoped authority to enhance agility while managing risks.
2. **TOOL USE AND INTEGRATION** — agents must invoke external tools, APIs, or services reliably, requiring robust interfaces and safeguards.
3. **COORDINATION AND SPECIALIZATION** — multi-agent systems should divide labor through explicit roles and structured coordination mechanisms.
4. **OBSERVABILITY AND TRANSPARENCY** — all agent actions and decisions must be logged and explainable to support trust, debugging, and compliance.
5. **GOVERNANCE AND COMPLIANCE** — oversight mechanisms, including policy enforcement layers and human-in-the-loop checkpoints, are essential to align agent behavior with organizational and ethical standards.

Together, these elicitation lists represent the distilled outcome of the literature review. In RE terminology, they correspond to the *documentation* of raw requirements prior to formal specification. In the next step, a *clustering* analysis was conducted to identify similarities and resolve redundancy across the three lists (e.g., governance appears in both workflow automation and agentic AI). This step corresponds to the *analysis phase* in Herrmann (2022) RE cycle, where elicited items are consolidated and normalized before specification.

## 4.2 Requirement Reformulation & Classification

Following Glinz et al. (2020), each requirement is formulated as a single, unambiguous “shall” statement that is necessary, atomic, verifiable, and consistent. Requirements are classified into functional, quality, and constraint types. In SysML v2, requirements are modeled as dedicated requirement elements with their textual content captured in the description field, and their relationships (e.g. «satisfy», «verify») expressed through model links. Rationale and verification considerations are documented narratively in the requirements analysis and traceability sections, while the requirements themselves remain concise textual statements embedded in the SysML v2 model.

Functional requirements specify externally visible system behaviors, separated from quality attributes and constraints. They were elicited from the categorized insights of the literature review, documented in “shall” form, and prepared for trace links in the SysML requirements model.

### FR-01 PROCESS ORCHESTRATION ENGINE

*Statement*— The system shall execute workflow models by dispatching tasks to human or software actors according to model control flow and business rules.

*Rationale*— Centralized orchestration ensures reliable, repeatable execution.

### FR-02 TASK ASSIGNMENT & ROLE ROUTING

*Statement*— The system shall route tasks based on roles, skills, and authorization.

*Rationale*— Role- and skill-aware routing aligns work with organizational responsibilities.

**FR-03 TASK REASSIGNMENT AND ESCALATION**

*Statement*— The system shall ensure task continuity by supporting reassignment to eligible actors and enforcing time-based escalation policies.

*Rationale*— Continuity mechanisms prevent stalls and maintain service levels.

**FR-04 MODEL INGESTION**

*Statement*— The system shall ingest workflow or process definitions in a machine-readable format and make them available for execution and versioning.

*Rationale*— Importable models enable governance, repeatability, and controlled change.

**FR-05 ENTERPRISE INTEGRATION**

*Statement*— The system shall provide connectors to interact with external applications, data sources, and services as workflow steps.

*Rationale*— Interoperability enables end-to-end automation across heterogeneous systems.

**FR-06 AGENT TOOL USE**

*Statement*— The system shall allow agent components to invoke approved external tools or APIs with controlled inputs/outputs and capture results for downstream steps.

*Rationale*— Tool use turns agents into capable actors while containing risk.

**FR-07 INTER-AGENT COORDINATION**

*Statement*— The system shall support coordination patterns (e.g., hierarchical, peer-to-peer, brokered) among multiple agents executing tasks.

*Rationale*— Structured coordination prevents conflicts and enables distributed problem solving.

**FR-08 EXCEPTION HANDLING**

*Statement*— The system shall detect execution errors and deviations, support compensating actions and escalation, and enable resumable recovery.

*Rationale*— Robust exception handling preserves correctness and availability.

**FR-09 WORKFLOW GOVERNANCE**

*Statement*— The system shall enforce access control, maintain audit trails, and ensure compliance with organizational policies during workflow execution.

*Rationale*— Governance guarantees accountability and regulatory conformity.

#### FR-10 AUTONOMY IN DECISION-MAKING

*Statement*— The system shall enable agents to make decisions within explicitly scoped authority without requiring constant human input.

*Rationale*— Scoped autonomy improves responsiveness while containing risk.

#### FR-11 OBSERVABILITY & TRANSPARENCY

*Statement*— The system shall log agent decisions, actions, and tool invocations in a human-readable and queryable format.

*Rationale*— Transparent logs support trust, auditing, and debugging.

#### FR-12 HUMAN-IN-THE-LOOP CONTROL & RUNTIME VIEWS

*Statement*— The system shall provide configurable human approval/override points and runtime execution views, including replay capability.

*Rationale*— Human oversight and operational views enable intervention and diagnosis.

#### FR-13 POLICY & COMPLIANCE ENFORCEMENT

*Statement*— The system shall validate agent actions against declarative policies and block or redirect disallowed behaviors.

*Rationale*— Policy enforcement prevents rule violations and unsafe behavior.

#### FR-14 RISK-BASED ROUTING

*Statement*— The system shall evaluate risk signals (e.g., anomalies, policy matches) and prefer safer modeled paths without overriding explicit policy decisions.

*Rationale*— Risk-based routing provides adaptive safeguards while preserving compliance.

#### FR-15 CONFIGURATION, SCALABILITY & EXPLAINABILITY

*Statement*— The system shall maintain versions of process models, policies, agent roles, and connectors; support controlled rollout and rollback of configurations; distribute workflows and agents across execution environments; and provide

justifications for agent decisions with references to inputs, rules, and tools used.

*Rationale*— Versioning, scalability, and explainability together ensure safe evolution, resilience, and trust in system operation.

*Verification approach.* Each functional requirement is verifiable by inspection (model presence), analysis (policy/model correctness), or test (execution against acceptance scenarios). The SysML model (Appendix) will provide trace links from FR-IDs to architecture elements (agents, gateways, connectors) and validation scenarios.

### 4.3 Model-Based Requirements Representation

## 5 Modeling the Architecture

**Listing 1:** Excerpt of the Requirements model

### 5.1 [placeholder]Modeling the Agents

### 5.2 [placeholder]Modeling the Architecture

### 5.3 [placeholder]Modeling the Interactions

## 6 Discussion: Applicability Criteria

## 7 Conclusion

Future work should extend this conceptual design into practical evaluation and implementation. In particular, empirical validation of the architecture in industry settings, tool-supported instantiation in SysML, and comparative studies against traditional workflow automation would provide valuable evidence of its applicability and impact. Further, integrating additional agentic AI capabilities such as autonomous negotiation or explainability could enhance both usability and compliance assurance.

## References

- Basu, Amit and Akhil Kumar (2002). “Research Commentary: Workflow Management Issues in e-Business”. *Information Systems Research* 13, pp. 1–14.
- Carvalho, André M. et al. (2023). “Operational excellence, organizational culture, and agility: bridging the gap between quality and adaptability”. *Total Quality Management & Business Excellence* 34, pp. 1598–1628.
- Glinz, Martin et al. (2020). *Handbook for the CPRE Foundation Level according to the IREB Standard*. International Requirements Engineering Board.
- Herrmann, Andrea (2022). *Grundlagen der Anforderungsanalyse: Standardkonforme Requirements Engineering*. Wiesbaden Heidelberg: Springer Vieweg.
- Hevner, Alan R. et al. (2004). “Design Science in Information Systems Research”. *MIS Quarterly* 28, pp. 75–105.
- IEEE (1990). *Standard Glossary of Software Engineering Terminology*. Corrected ed. New York: Institute of Electrical and Electronics Engineers.
- Juran, Joseph Moses and A. Blanton Godfrey (1999). *Juran’s quality handbook*. 5th ed. McGraw Hill.
- Mayring, Philipp and Thomas Fenzl (2022). “Qualitative Inhaltsanalyse”. *Handbuch Methoden der empirischen Sozialforschung*. Springer Fachmedien Wiesbaden, pp. 691–706.
- Owoade, Samuel et al. (2024). “Systematic Review of Strategic Business Administration Practices for Driving Operational Excellence in IT-Driven Firms”. *International Journal of Scientific Research in Science and Technology* 11, pp. 680–700.
- Peppers, Ken et al. (2007). “A design science research methodology for information systems research”. *Journal of Management Information Systems* 24, pp. 45–77.
- Stohr, Edward A. and J. Leon Zhao (2001). “Workflow Automation: Overview and Research Issues”. *Information Systems Frontiers* 3, pp. 281–296.
- Womack, James P. and Daniel T. Jones (2013). *Lean Thinking: Ballast abwerfen, Unternehmensgewinn steigern*. 3. Auflage. Weinheim: Campus Verlag.

## **Appendix A: Full SysML Models**

**Requirements Model**

**Architecture Model**

**Interactions Model**