

Proposal

Master Thesis Research Proposal – A Simple One-pager Abstract

IMC University of Applied Sciences, Krems

Author – David Bobek

Title – SAIL: A Scalable AI Lifecycle Framework for Coordinated AI Adoption in Software Organizations

Purpose – The purpose of this paper is to develop and evaluate SAIL, a structured framework that enables software

organizations to adopt AI in a coordinated, scalable, and reusable manner. The expected outcome is a plug-and

play AI adoption model with clear guidelines for use case mapping, staged implementation (Awareness → Pilot

→ Scale → AI-Native), governance, and cross-team reuse—providing practical value for CTOs, engineering leads, and innovation managers.

Research gap – While literature addresses enterprise architecture (Bernard, 2012), digital transformation (Warner

& Wäger, 2019), innovation diffusion (Rogers, 2003), dynamic capabilities (Teece, 2007), and resource-based

view (Barney, 1991), there is no integrated, use-case-driven framework designed specifically for scalable AI

adoption across software teams. Existing models focus on isolated AI pilots or generic transformation strategies,

lacking mechanisms for coordination, reuse, and progressive scaling. This study fills that gap by proposing and

conceptually validating SAIL as a practical, theory-informed framework tailored for software organizations.

Research question – How should AI adoption in software organizations be structured to ensure scalability,

coordination, and reusability across diverse use cases?

Theoretical lens – The study is grounded in five key theoretical foundations: (1) Enterprise Architecture (EA)

(Bernard, 2012) for systemic integration; (2) Digital Transformation (Matt et al., 2015) for organizational readiness; (3) Innovation Diffusion Theory (Rogers, 2003) to understand adoption dynamics; (4)

Dynamic

Capabilities (Teece, 2007) for strategic agility in sensing, seizing, and transforming AI opportunities; and (5)

Resource-Based View (RBV) (Barney, 1991) to treat AI assets (models, data, knowledge) as valuable, rare, and

hard-to-imitate resources. These lenses collectively inform SAIL's design and evaluation.

Design/methodology/approach – This research follows a conceptual, theory-driven methodology using design

based research (Hevner et al., 2004). The approach includes: (1) a comprehensive literature review synthesizing

theories on AI adoption and organizational change; (2) framework development defining SAIL's pillars (use case identification, adoption staging, governance, implementation roles, evaluation & reuse); and (3) conceptual evaluation through theoretical application to realistic AI use cases (e.g., code generation, documentation automation, sprint planning). For each use case, I will outline step-by-step how SAIL guides: identification → prioritization → pilot design → scaling path → governance → reuse. This demonstrates SAIL's internal coherence, scalability, and practical utility without primary data collection.

Time plan – Submit proposal: End of this week

- Finish literature review and start empirical part: By the end of September
- Submit your master thesis: By the End of the year

Expected challenges & limitations – As a conceptual study, the main limitation is the absence of real-world validation. However, the use case walkthroughs are designed to simulate realistic organizational contexts, enhancing practical plausibility. A key challenge is ensuring the framework remains both theoretically rigorous and accessible to practitioners, which will be addressed through iterative refinement and alignment with industry relevant examples.

Master Thesis

Title: SAIL – A Scalable AI Lifecycle Framework for Coordinated AI Adoption in Software Organizations

Author: [Your Name]

Supervisor: Roger Hage

Institution: IMC University of Applied Sciences, Krems

Date: [Month, Year]

PROF

Abstract

- Purpose and research problem.
- Research gap in AI adoption frameworks.
- Theoretical lenses applied.
- Methodology (Design Science Research).
- Key contributions (framework + evaluation).
- Expected impact for academia and practice.

Table of Contents

1. Introduction	
2. Literature Review	
3. Theoretical Framework	
4. Research Methodology	
5. Framework Development (SAIL)	
6. Use Case Walkthroughs	
7. Discussion	
8. Conclusion & Future Work	
References	
Appendices	

1. Introduction

1.1 Background

- Role of AI in modern software organizations.
- Industry applications (development, project management, decision-making).
- Drivers of adoption (efficiency, cost reduction, innovation).
- Pressures: regulatory, competitive, customer expectations.

1.2 Problem Statement

- Fragmented AI initiatives; isolated pilot projects.
- Lack of scalability and systematic governance.
- Coordination issues across departments/teams.
- Risks: inefficiency, duplication, lack of standardization.

1.3 Research Gap

- Existing frameworks address transformation broadly but not AI adoption specifically.
- AI maturity models focus on levels, not lifecycle coordination.
- Limited guidance on reuse and knowledge transfer across teams.
- Need for an integrated, lifecycle-based adoption framework.

1.4 Research Question and Objectives

- *RQ: How should AI adoption in software organizations be structured to ensure scalability, coordination, and reusability across diverse use cases?*
- Objectives:
 - Develop SAIL framework.
 - Integrate multiple theoretical lenses.
 - Apply framework to use case walkthroughs.
 - Highlight theoretical and practical contributions.

1.5 Structure of the Thesis

- Overview of chapters and flow of arguments.
-

2. Literature Review

2.1 Enterprise Architecture and AI Integration [REF]

- EA as systemic integration mechanism.
- EA's role in aligning IT with strategy.
- Potential for supporting AI alignment and scaling.

2.2 Digital Transformation and Organizational Readiness [REF]

- Definition and scope of digital transformation.
- Barriers to readiness (skills, culture, infrastructure).
- Stages of transformation maturity.
- Lessons for AI adoption.

2.3 Innovation Diffusion Theory [REF]

- Rogers' model: innovators, early adopters, majority, laggards.
- Adoption curve dynamics.
- Relevance to AI technologies in software organizations.
- How diffusion insights guide scaling strategies.

2.4 Dynamic Capabilities [REF]

- Sensing opportunities (AI trends, competitor actions).
- Seizing opportunities (investment, pilot design).
- Transforming organizations (structures, processes).
- Application to fast-moving AI landscape.

PROF

2.5 Resource-Based View (RBV) and AI Assets [REF]

- VRIN framework: data, algorithms, expertise as resources.
- Building sustained competitive advantage through AI.
- Risks of resource imitation or commoditization.

2.6 Related AI Adoption Models [REF]

- Overview of AI maturity models.
- Digital transformation frameworks.
- Strengths and weaknesses.
- Missing elements (coordination, reuse, lifecycle orientation).

2.7 Synthesis and Identified Gaps

- Comparative analysis across theories and models.

- Explicit statement of gaps.
 - Justification for creating SAIL.
-

3. Theoretical Framework

3.1 Conceptual Foundations of SAIL

- Purpose of a lifecycle adoption framework.
- Positioning relative to digital transformation and EA.
- Addressing fragmentation through coordination and reuse.

3.2 Integration of Theoretical Lenses

- EA → systemic integration.
- Digital Transformation → readiness and strategic alignment.
- Innovation Diffusion → adoption dynamics across teams.
- Dynamic Capabilities → agility in scaling and transformation.
- RBV → AI as strategic assets.

3.3 Principles Derived for Framework Design

- Scalability as a design principle.
 - Coordination across organizational units.
 - Knowledge reuse and cross-team learning.
 - Governance and accountability.
-

4. Research Methodology

4.1 Research Design – Design Science Research

- Why conceptual framework development fits this research.
- Link to design science cycles (relevance, rigor, design).

4.2 Literature Review Approach

- Search strategy, databases, keywords.
- Inclusion/exclusion criteria.
- Process of synthesizing theory.

4.3 Framework Development Process

- Iterative design and refinement.
- Mapping theories to practical framework elements.

4.4 Conceptual Evaluation via Use Cases

- Rationale for conceptual evaluation (vs. empirical).
- Selection of realistic use cases (code gen, documentation, planning).
- Evaluation criteria: coherence, scalability, reusability.

4.5 Limitations

- Absence of empirical validation.
- Risk of context-specific assumptions.
- Boundaries of applicability.

5. Framework Development (SAIL)

5.1 Overview of SAIL Framework

- Core design logic.
- Lifecycle perspective.
- Visual diagram.

5.2 Adoption Stages (Awareness → Pilot → Scale → AI-Native)

- Awareness: building knowledge, initial exploration.
- Pilot: testing in limited scope, measuring outcomes.
- Scale: expanding across units, refining governance.
- AI-Native: embedding AI into strategy, culture, and processes.
- Success factors and risks for each stage.

5.3 Governance and Roles

- Role of CTOs, innovation managers, engineering leads.
- Governance mechanisms: policies, ethics, compliance.
- Decision-making structures.
- Risk and accountability management.

5.4 Use Case Mapping & Prioritization

- Criteria: feasibility, impact, alignment with strategy.
- Prioritization frameworks (e.g., impact-effort matrices).
- Portfolio approach to managing multiple use cases.

5.5 Evaluation and Reuse Mechanisms

- Capturing lessons learned.
 - Creating knowledge repositories.
 - Facilitating cross-team reuse.
 - Feedback loops for continuous improvement.
-

6. Use Case Walkthroughs

6.1 Code Generation

- Problem definition.
- Mapping to SAIL stages.
- Governance considerations.
- Expected outcomes and reuse potential.

6.2 Documentation Automation

- Step-by-step application of SAIL.
- Benefits: efficiency, knowledge standardization.
- Risks: accuracy, oversight.

6.3 Sprint Planning

- AI-supported resource allocation and scheduling.
- Coordination across agile teams.
- Governance and trust issues.

6.4 Cross-Case Insights

- Common success factors.
- Differences in adoption dynamics.
- Evidence of reusability and scalability.

7. Discussion

7.1 Theoretical Contributions

- How SAIL extends existing frameworks.
- Integration of multiple theoretical lenses into one model.

7.2 Practical Implications for Software Organizations

- Playbook for CTOs and managers.
- Guidelines for avoiding fragmented AI adoption.

7.3 Comparison with Existing Models

- Side-by-side strengths and weaknesses.
- Unique value of SAIL (lifecycle + coordination + reuse).

7.4 Challenges and Limitations

- Conceptual boundaries.
- Risks of overgeneralization.
- Directions for empirical follow-up.

8. Conclusion & Future Work

8.1 Summary of Findings

- Restate research question and main outcomes.

8.2 Contributions to Theory and Practice

- Academic relevance.
- Practical usability for organizations.

8.3 Future Research Directions

- Empirical testing of SAIL.
- Expansion to other industries beyond software.

8.4 Final Reflection

- Broader perspective on AI adoption journey.
- Closing thought on the role of frameworks in digital transformation.

References

[Use [REF] placeholders during drafting; insert proper citations later.]

Appendices

- Extended diagrams of SAIL.
- Comparison tables of frameworks.
- Additional case walkthrough details.