SAIL – A Strategic AI Lifecycle Framework

Coordinated AI Adoption in Software Organizations

Master Thesis

Submitted to
IMC University of Applied Sciences Krems

Masters Programme Digital Business Innovation and Transformation

by

David Bobek

for the award of academic degree
Master of Arts in Business (MA)

under the supervision of
[Roger Hage]

Submitted on March 2026

Table of Contents

Declaration of honour

I declare on my word of honour that I have written this Master Thesis on my own and that I have not used any sources or resources other than stated and that I have marked those passages and/or ideas that were either verbally or textually extracted from sources. This also applies to drawings, sketches, graphic representations as well as to sources from the internet. The Master Thesis has not been submitted in this or similar form for assessment at any other domestic or foreign post-secondary educational institution and has not been published elsewhere. The present Master Thesis complies with the version submitted electronically.

David Bobek
March 2026

ACKNOWLEDGEMENTS

[Your acknowledgements here]

ABSTRACT

The rapid adoption of Artificial Intelligence (AI) technologies is transforming the way software organizations operate. While AI offers significant potential for innovation, efficiency, and competitive advantage, many organizations struggle to integrate AI solutions effectively into their existing workflows. Challenges include aligning AI initiatives with strategic objectives, managing the AI lifecycle, ensuring governance, and fostering organizational readiness.

This thesis introduces **SAIL**, the *Strategic AI Lifecycle* framework, designed to guide software organizations in the coordinated adoption of AI. SAIL provides a structured approach to managing AI initiatives, from strategic planning to deployment and continuous monitoring, ensuring that AI adoption is both effective and sustainable. By leveraging this framework, organizations can systematically address technical, operational, and strategic challenges associated with AI, enabling better decision-making and innovation.

The research develops a comprehensive framework for AI adoption in software organizations, provides practical guidelines for managing the AI lifecycle, and highlights best practices and lessons learned for coordinated AI implementation.

Keywords: AI Adoption, Enterprise Architecture, Digital Transformation, Innovation Diffusion Theory, Dynamic Capabilities, MLOps, AI Governance

Chapter 1
INTRODUCTION

The rapid adoption of Artificial Intelligence (AI) technologies is transforming the way software organizations operate. While AI offers significant potential for innovation, efficiency, and competitive advantage (McKinsey & Company, 2025; Dzreke, 2024), many organizations struggle to integrate AI solutions effectively into their existing workflows (Deloitte, 2025; Haefner et al., 2023). Challenges include aligning AI initiatives with strategic objectives, managing the AI lifecycle, ensuring governance, and fostering organizational readiness (IBM, 2024; Ransbotham et al., 2020). Organizations face significant barriers in scaling AI beyond initial pilots, with issues ranging from technical complexity to organizational resistance (Makarius et al., 2020).

This thesis introduces **SAIL**, the *Strategic AI Lifecycle* framework, designed to guide software organizations in the coordinated adoption of AI. SAIL provides a structured approach to managing AI initiatives, from strategic planning to deployment and continuous monitoring (Stone et al., 2025), ensuring that AI adoption is both effective and sustainable. By leveraging this framework, organizations can systematically address technical, operational, and strategic challenges associated with AI, enabling better decision-making and innovation.

1.1 Research Objectives

The primary objectives of this thesis are to:

- Develop a comprehensive framework for AI adoption in software organizations.
- Provide practical guidelines for managing the AI lifecycle.
- Evaluate the effectiveness of the SAIL framework through case studies or simulations.

- Highlight best practices and lessons learned for coordinated AI implementation.

Despite the growing body of research done on air maturity models, digital transformation and enterprise architecture, the. Already existing approaches in the current state of the industry largely fail to provide an orchestrated, holistic approach on AI adoption and full lifecycle management. The current. Frameworks are not sufficient to guide software organizations from early AI experimentation to scalable organization wide. AI adoption. Most of the already existing models focus simply on analysis and assessment rather than a full execution. They often lack mechanisms for reflecting on processes and fail to encompass iterative approach.

1.2 Research Questions

Will be guided by the following research questions:

RQ1: How can AI adoption in software organizations be structured across the full lifecycle from opportunity identification to scalable, AI-native integration?

RQ2: What organizational, architectural, and governance mechanisms are required to enable coordinated and scalable AI adoption?

RQ3: How can a lifecycle-oriented framework support decision-making, reuse, and alignment across multiple AI initiatives?

Chapter 2
LITERATURE REVIEW

In order to ground the SAIL framework in established theory, this chapter will focus on the current state of research across five key theoretical lenses relevant to AI adoption in software organizations. The concept of AI adoption is multifaceted, involving technological, organizational, and strategic dimensions (Dwivedi et al., 2021). The main challange is the lack of an integrated framework which is addressing the concept of “Full Lifecycle” adoption of AI in a coordinated and scalable manner (Nogare and Silveira, 2025). The aspect of AI being a very new technology, hinders the existence of established frameworks and supportive literature (Benbya et al., 2020). Organizations require comprehensive guidance that addresses both technical and managerial aspects of AI adoption (Berente et al., 2021). Therefore, this chapter will rather review individual theories and models that are going to be synthesized later throughou the theoretical framework chapter.

2.1 Enterprise Architecture and AI Integration

In order to ground the SAIL framework in established theory the focus on Enterprise Architecture (EA) is crucial. Enterprise Architecture is providing a holistic approach to the problem of aligning IT with business strategy and goals. The principle of EA and its utilization in organizations is well documented in literature. However the not so well pointed out aspect is the type of company to which EA is applied. Different types of companies have different needs and requirements and technological readiness. Software organizations are typically more agile and introduction of major changes is easier to implement and does not create a lot of resistance (Kaddoumi and Watfa, 2022). In contrast to this, traditional industries such as manufacturing or banking are more rigid and changes are harder to implement, thus EA frameworks in these industries are more rigid and bureaucratic (Canat et al., 2018).

EA does not work as a silver bullet and it is not solving all problems by itself. There are several frameworks and methodologies which are used in practice. The most well known and implemented in companies are The Open Group Architecture Framework (TOGAF) (The Open Group, 2022) and the Zachman Framework (Zachman, 1987). The mentioned frameworks are the current state of the art in EA and are widely used in practice. Each of the frameworks has its strengths and weaknesses and the choice of the framework depends on the context of the organization. TOGAF being more dominant in the cases where the organization is more agile and dynamic, while Zachman Framework is more suitable for traditional and rigid organizations.

In the current Digital Transformation era, TOGAF is more suitable as more and more organizations are becoming agile and dynamic and the usage of AI can be implemented in a way that is more aligned with TOGAF principles while still keeping the structure and rigor of EA (Fitriani et al., 2023). The usage and value extraction of AI in organizations is not a trivial task and it requires a structured approach. EA can provide the necessary structure and rigor to ensure that AI initiatives are aligned with business strategy and goals (Ern et al., 2025).

EA is therefore crucial for AI integration and scaling :

- It alligns AI initiatives with business strategy, ensuring that isolated pilots are connected to broader organizational goals and prevents existance of "siloed" AI projects (McKinsey & Company, 2025).
- It provides a systemic view of the organization while having established governance mechanisms, which is crucial for managing the complexity of AI adoption across multiple teams and departments.
- It facilitates reuse of AI assets (models, data, knowledge) across teams by establishing common standards and repositories, thus preventing duplication of efforts and promoting efficiency (Lakarasu, 2022).

On contrast the framework of Zachman is more rigid and bureaucratic and it is not so suitable for the dynamic nature of AI adoption. The Zachman points out that the architecture is a schema for organizing architectural artifacts (for example, design documents, specifications, and models) that takes into account both who the artifact targets (for example, business owner and builder) and what particular issue (for example, data and functionality) is being addressed (Robertson-Dunn, 2012). The Zachman framework is more suitable for traditional and rigid organizations where changes are harder to implement. The usage of AI in such organizations is more challenging as the adoption of AI requires a more agile and dynamic approach.

Both of the frameworks have their strengths and weaknesses and the choice of the framework depends on the context of the organization. However, for the purpose of this thesis, TOGAF is more suitable as it is more aligned with the principles of Digital Transformation and AI adoption. Both of them were developed in the times of “pre-AI” era and therefore do not address the specific challenges and requirements of AI adoption. However, the principles and concepts of EA can be adapted and extended to address the specific needs of AI adoption in software organizations. The SAIL framework will therefore build upon the principles of EA and adapt them to the specific context of AI adoption in software organizations.

2.2 Digital Transformation and Organizational Readiness

The concept of Digital Transformation is closely related to the adoption of AI in organizations. Digital Transformation refers to the integration of digital technologies into all areas of a business, fundamentally changing how the organization operates and delivers value to customers (Westerman et al., 2019). Year by year, more and more organizations are undergoing Digital Transformation in order to stay competitive in the market and the prevalence of AI is only amplifying this successful digital revolution. The proof of the fact that Digital Transformation is in fact a successful concept can be seen by comparing the performance of companies which have undergone Digital Transformation in the last decade. Digital leaders achieve earnings growth that is 1.8 times higher than digital laggards, and more than double the growth in total enterprise value (Forth et al., 2020). The companies which have undergone Digital Transformation have seen a significant increase in revenue and market share, while the companies which have not undergone Digital Transformation have seen a decline in revenue and market share.

The success of Digital Transformation can be observed in the performance of companies that have succesfully embraced it compared to those that have not. Firms that have implemented Digital Transformation strategies report significantly stronger revenue growth and improved market share (Deloitte, 2023). Research shows that digital leaders achieve substantially higher earnings growth and enterprise value compared to their less digitally mature peers (Forth et al., 2020). However, the path to success is challenging: 70% of digital transformations fall short of their objectives, with only 30% achieving their targets and creating sustainable change (Forth et al., 2020).

These results show that Digital Transformation is not just a theory; it is a real way to gain a competitive edge. As AI technologies become more and more a part of Digital Transformation, they are expected to help businesses make more money, run more smoothly, and stay strong over time.

Proof of Digital Transformation success:

- Digital leaders achieve earnings growth that is 1.8 times higher than digital laggards, and more than double the growth in total enterprise value (Forth et al., 2020). Organizations must carefully select appropriate metrics to measure digital transformation progress, as wrong KPIs can lead to misaligned efforts and failed outcomes (Ahmad et al., 2021; Schrage et al., 2022).

- Successful digital transformations create, on average, 66% more value and improve corporate capabilities by 82% compared to failed transformations (Forth et al., 2020).

- Only 30% of digital transformations meet or exceed their target value and result in sustainable change, while 70% fall short of their objectives (Forth et al., 2020).

- Organizations that adopt AI as part of their digital transformation strategy gain significant competitive advantages in innovation, efficiency, and market positioning (McKinsey & Company, 2024).

Regarding AI adoption, Digital Transformation provides several lessons: - Organizational readiness is crucial. Successful Digital Transformation requires not just technology but also changes in culture, skills, and processes (Jöhnk et al., 2021). Similarly, AI adoption needs organizational readiness to manage change effectively (Hradecky et al., 2022; Holmström, 2022). Building AI-powered organizations requires systematic transformation of leadership, technology, data, and organizational capabilities (Fountaine et al., 2019). - Maturity models can guide progression. Digital Transformation maturity models outline stages from initial experimentation to full integration. AI adoption can benefit from similar staged approaches to manage complexity and scale effectively (Enholm et al., 2022).

The implementation of AI in organizations very much depends on the readiness of the organization to embrace change (Shrestha et al., 2019). This change comes in different forms, such as changes in culture, skills, and processes but also in the current technological infrastructure of the company and its ability to integrate new technologies. Challanges such as lack of permissions, lack of skills, lack of understanding of the technology and its potential, lack of resources and budget are all common barriers to the adoption of AI in organizations (Bain & Company, 2024). These barriers can be overcome by following a structured approach to AI adoption, which is the main purpose of the SAIL framework. This would allow companies to overcome the barriers and successfully implement AI in their organizations and show them the standardized way of doing so and thus increase the chances of success while outlining which fields of the organization need to be changed in order to successfully implement AI.

The broad term of organizational readiness can be measured through different maturity models (Al-Ali and Marks, 2022). The most well known and widely used maturity model is the Digital Maturity Model (DMM) developed by Deloitte (Deloitte, 2023). The DMM outlines five stages of digital maturity: 1. Initial: Ad hoc and uncoordinated digital initiatives. 2. Developing: Some digital initiatives, but still siloed and uncoordinated. 3. Defined: Digital initiatives are defined and coordinated across the organization. 4. Managed: Digital initiatives are managed and measured for impact. 5. Optimized: Digital is fully integrated into business strategy and operations.

In order to successfully implement AI in organizations, the assumption of high level of digital maturity is crucial. The higher the level of digital maturity, the higher the chances of success in implementing AI in organizations. The SAIL framework will therefore build upon the principles of Digital Transformation and adapt them to the specific context of AI adoption in software organizations.

2.3 Innovation Diffusion Theory

Innovation Diffusion Theory (IDT), developed by Everett Rogers, provides a valuable lens for understanding how new technologies, such as AI, are adopted within organizations (Rogers, 2003). This theory is particularly relevant in the context of AI adoption in software organizations, where the pace of rapid industry change and technological advancement creates both opportunities and challenges for adoption.

IDT become relevant in the context of technology adoption in organizations. The theory outlines how innovations are communicated over time among the members of a social system. The key elements of IDT include the innovation itself, communication channels, time, and the social system (Dearing, 2009). The rise of usage of IDT is closely related to the rise of technology adoption in organizations and the recognition that successful adoption of new technologies requires more than just the technology itself; it also requires effective communication and social dynamics within the organization (Almaiah, 2022).

Perfect example where IDT was successfully applied is in the Israeli tech sector, where early adopters in startups and tech companies drove the initial adoption of AI technologies, leading to broader acceptance across the industry. This diffusion was facilitated by strong networks and knowledge sharing among innovators and early adopters, which helped to build trust and demonstrate the value of AI. Thus creating a country which swiftly became the "Startup Nation" and a global leader in AI innovation while having an insignificant population of only 9 million people and the amount of AI startups per capita being the country with the highest number worldwide (Startup Genome, 2024; Start-Up Nation Central, 2024).

The key components of IDT include: - Innovation: The acceptance of AI technologies as a new innovation within the organization. - Communication Channels: The methods through which information about AI is shared within the organization (e.g., meetings, workshops, internal communications). - Time: The duration over which AI adoption occurs, including the stages of adoption. - Social System: The organizational culture and structure that influence how AI is perceived and adopted.

These key components were laid out in Rogers' seminal work "Diffusion of Innovations" (2003) more than two decades ago, but they remain highly relevant in today's context of rapid technological change and AI adoption. The ability to understand and leverage these components is crucial for successful AI adoption in software organizations (Schneider et al., 2023). Each of these components plays a critical role in shaping the adoption process and determining the success of AI initiatives. By following the principles of IDT, organizations are able to effectively manage the adoption process, address resistance, and build momentum for AI initiatives, while still being aware of the risks and challenges associated with AI adoption. The SAIL framework will therefore incorporate insights from IDT to guide the scaling strategies for AI adoption in software organizations. The effective usage of IDT in organisations can be implemented through the following steps:

1. Start with innovators and early adopters: Identify and engage individuals or teams within the organization who show a strong interest in AI and are willing to experiment with new technologies. This will naturally create champions for AI adoption. The most important aspect is to identify the right people who are willing to experiment and take risks and give the people the freedom to do so. This will create a culture of innovation and experimentation within the organization.

2. Leverage communication channels: The usage of effective communication channels is crucial for spreading awareness. The spark that was created by the innovators and early adopters needs to be spread across the organization and let the fire of innovation spread. The most effective ways to do so are through company wide presentantions, demos, workshops and upper management support. The communication needs to be clear and concise and the benefits of AI adoption need to be highlighted.

3. Address concerns and resistance: It is natural for people to be resistant to change, the natural human behavior is to resist change and stick to the status quo. Therefore the emphasize on addressing concerns and resistance is crucial. This is done by creating a level of trust and transparency. The concerns of employees need to be addressed and the benefits of AI adoption need to be highlighted. For the successful adoption of AI the value proposition needs to be clear and the benefits need to outweigh the risks (Patnaik, 2024).

4. Build a supportive social system: Organizational culture and structure play a crucial role in shaping how AI is perceived and adopted. This means that the culture of the organization needs to be supportive of innovation and experimentation. By fostering a culture that is exploratory and innovation driven we can create an environment where AI adoption can thrive. The kernel of the culture needs to allign with the long term mission and vision of the organization which at the end needs to be AI-Native.

By following the principles of IDT, organizations have a clear way to manage the adoption process, address resistance, and create a momentum where AI is embraced and incorporated into the culture which the SAIL framework will build upon.

2.4 Dynamic Capabilities Theory

Dynamic Capabilities Theory (DCT), introduced by David Teece (Teece et al., 1997), provides a valuable framework for understanding how organizations can adapt and thrive in rapidly changing environments such as the current AI landscape which is mostly characterized by extremely fast technological advancements and changes. DCT as a theory is particularly emphasizing the importance of an organization's ability to sense, seize, and transform in order to maintain a competitive advantage. The winner is not the strongest or the most intelligent, but the one who is most adaptable to change while still achieving a sustained competitive advantage. DCT was origanlly developed in the context of strategic management and organizational theory, but its wide adoption in the tech and software industry is closely related to the rapid pace of technological change and the need for organizations to become more agile, adaptive and innovative in order to stay competitive.

DCT consists of three main components: 1. Sensing: The ability to identify and assess opportunities and threats in the external environment. 2. Seizing: The ability to mobilize disposable resources to capture value from opportunities. 3. Transforming: The ability to continuously renew and realign organizational processes and structures (Liu, 2024) to adapt to changing environments while still maintaining operational efficiency.

The winners in the modern digital economy are those who can quickly sense new opportunities, seize them effectively, and transform their organizations to stay ahead of the competition. These 3 simple steps might seem trivial, but they are the key to success. This type of mental allignment will be extremely relevant for the successful adoption of AI in organizations. By not following these principles, organizations risks moving and operating outside of their core competencies and thus deviating from their core mission and vision. Having a clear 3 step approach helps the organization to stay focused and aligned while still being able to reflect and adapt to the changing environment.

DCT can be used in both small and large organizations, but the implementation might differ. In small organizations, the decision-making process is typically more centralized, allowing for quicker sensing and seizing of opportunities. On the other hand the lack of available and disposable resources might hinder the ability to seize opportunities effectively and thus leading to a more visionary mindset instead of an operational one. In large organizations, the decision-making process is typically more decentralized and complex, which in most cases leads to slower sensing and seizing of opportunities. The advantage of large organizations is the availability of resources which allows them to seize opportunities more effectively and while not being as visionary as small organizations, they are more operationally efficient and the ability to transform and capitalize on opportunities is higher.

The usage of DCT can be perfectly showcased on an rather succesful startup from the heart of Tel Aviv called "Waze". Waze was founded in 2006 and was acquired by Google in 2013 for $1.1 billion. The success of 3 Jewish founders Ehud Shabtai, Amir Shinar, and Uri

Levine can be attributed to their ability to sense the opportunity of real-time traffic information, seize the opportunity by developing a mobile app that leveraged user-generated data, and transform the organization by continuously updating and improving the app based on very crucial user feedback. Organizations developing AI capabilities must similarly develop dynamic capabilities that enable them to sense opportunities, seize them through AI implementation, and transform their operations to maintain competitive advantage (Mikalef and Gupta, 2021; Wamba-Taguimdje et al., 2021).

2.5 Related AI Adoption Models

The AI adoption models that are currently existing in literature and practice are mostly focused on the maturity of AI adoption in organizations. These models are typically structured around different levels of maturity and their intended use is to help organizations assess their current state of AI adoption and identify areas for improvement. This is a very useful approach, which can help organizations where to start and what are the prerequisites for successful AI adoption. However, these models are mostly focused on the maturity of AI adoption and do not provide a structured approach to the actual adoption process itself. This is the research and market gap where the SAIL framework is trying to fill in.

Several AI maturity models have been developed by leading consulting firms and research organizations to help organizations assess their AI readiness and capabilities. The most prominent example is Deloitte's AI Maturity Model, which outlines five levels of AI maturity, from "Ad Hoc" to "Optimized", focusing on areas such as strategy, governance, data management, and talent (Deloitte, 2023). Similar frameworks have been proposed by other major consulting firms, typically structured around progressive maturity stages that organizations can use for self-assessment. Academic research has contributed additional maturity models, including frameworks that progress from isolated ignorance to systematic intelligence (Lichtenthaler, 2020) and organizational readiness assessment frameworks that examine the structural prerequisites for AI adoption (Pumplun et al., 2019). However, a common limitation across these models is that while they are useful for assessing the current state of AI adoption, they generally do not provide a detailed, structured approach to the adoption process itself. Academic research on digital maturity in specific sectors, such as education, has also contributed frameworks that can be adapted to AI adoption contexts (Al-Ali and Marks, 2022).

These models are certainly useful for organizations in order to approach the topic of AI adoption and understand where they currently stand. Every company and business operating in the market needs to start somewhere and these models provide a good starting point. However, the main question is still opened and that is "What next?" and "How to actually implement AI in a structured way?". This question is still not answered by the existing models and is the most crucial one as the actual conversion of the theoretical knowledge into practical implementation is the most challenging part, but is the only part which actually creates value. Staying too long undercover with reasearch gets you only as far as the size of the company's budget allows you to go. The actual implementation and value creation is the only thing which will make the company successful in the long run. The purpose of the SAIL framework is to convert the theoretical knowledge into practical implementation and provide a structured approach to the adoption process itself. While focusing on mostly internal aspects of the organization and efficiency, resulting in reduction of overhead costs and increase of productivity.

2.6 Already Existing Solutions

This section this section is going to review all of the existing AI adoption in terms of their scaling solution, how the research is implemented, their frameworks across various sectors. The aim is to provide the current state of the industry and documented approaches.

2.6.1 Enterprise Infrastructure and Deployment Solutions

The research conducted over the last few years has placed AI deployment and integration as a full life cycle problem (Hechler et al., 2020; Sharma, 2024; Ali and Nicola, 2018). It encompasses the processes of design, DevOps, and governance pillars. It outlines that there are significant gaps when transitioning proof-of-concepts into production environments and therefore recommends that these gaps shall be addressed systematically.

Among the largest gaps is the underestimation of infrastructure requirements (Sharma, 2024; Hechler et al., 2020; Sundaramurthy and Ravichandran, 2022). The latest research states that the underestimation stems from computational demands in terms of the processing power required by AI models, the way AI models are served in terms of infrastructure setup and deployment expectations, and bottlenecks related to resource allocation when serving AI models to customers.

The principle of cloud-native architecture with AI has been explored by several studies (Shaban and Zeebaree, 2025; Ali and Nicola, 2018; Bhatia and Sandiri, 2025). These studies often link easily accessible cloud-native architectures with fast deployment through APIs. By having a cloud-native architecture, the deployment cycles are shortened and therefore the AI services often reach better scalability and accessibility.

When talking about regulated environments, a layered architecture is required (Bhatia and Sandiri, 2025; Sundaramurthy and Ravichandran, 2022). This type of architecture is required due to the need for compliance and easier auditing capabilities. By having a layered architecture, it is easier to not only track changes but also provide customer excellence through better service levels.

The principles of resilience and security are often mentioned due to model attack vectors (Sundaramurthy and Ravichandran, 2022; Sharma, 2024; Hechler et al., 2020). These studies often focus on the core principle of the SAIL framework being the scalability of AI systems. The key findings mention the importance of having graceful degradation frameworks as they are crucial for maintaining service levels in case of failures or attacks.

In order to ensure successful architecture and lotng-term sustainability, it is important to tailor AI architecture to the system as a whole (Anny, 2024; Ali and Nicola, 2018; Gupta et al., 2018). These studies often outline their own frameworks, all agreeing on the principle of deeply embedding AI into the core of their architecture. This is opposed to the bolt-on approach as it reduces technical debt over a longer period of time.

The need for a holistic orchestration framework has been outlined and thoroughly discussed in several studies (Ali and Nicola, 2018; Hechler et al., 2020; Shaban and Zeebaree, 2025). Studies often emphasize the importance of successful orchestration of various aspects, often noting DevOps principles, DataOps, and most importantly the cloud infrastructure. The evolution toward MLOps practices represents a critical maturation of AI deployment capabilities, providing structured approaches to operationalize machine learning at scale (AWS, 2024; John, 2025).

2.6.2 Human-Centric Organizational Transformation

Another key part of the AI framework challenge is the human centric organizational transformation. There is a certain paradigm shift from technology centric to human centric approaches (Fenwick et al., 2024; Wilson and Daugherty, 2023; Rinta-Kahila et al., 2022). The recent publications are outlining that the human aspect has been heavily neglected due to the increased focus on technical aspects such as model accuracy, infrastructure, and data quality (Fenwick et al., 2024; Wilson and Daugherty, 2023; Rinta-Kahila et al., 2022). As majority of the research states, the largest reason for resistance in terms of AI, adoption are aspects of job security, the lack of skill and the cultural shift of the ever changing work environment. In order for a successful AI adoption the HR has to take step in place. The hour in the sense of Not only leading the entire change management process, but also creating a structured path for the engineers. This path has to not only focus on reskilling, but creating a culture of curiosity. and continuous learning

The term collaborative intelligence comes into play when humans interact with AI Systems to augment their capabilities (Wilson and Daugherty, 2023). This principle is emphasizing on the complementary nature and the easy to use AI tools to elevate and enhance human capabilities rather than replacing them. Understanding the dynamics and determinants of technology adoption, as demonstrated in studies of blockchain and other emerging technologies, provides valuable insights that can be applied to AI implementation strategies (Fosso Wamba et al., 2021).

Several studies have tried to outline the basic principles of how to overcome the resistance to change when it comes to AI (Rinta-Kahila et al., 2022; Chowdhury et al., 2022; Jarrahi et al., 2023). These studies came to a joint common finding, which is mentioning the importance of involving employees in the early process by setting clear goals and expectations and starting the process together as a company instead of being mandated to use certain AI tools. The company is able to win early trust and buy-in from the employees. What is often mentioned as a great way how to gain trust is to provide several pilot programs desmonstrating real value. The studies are clearly showing that having a structured change management process always beats a strict mandated usage (Rinta-Kahila et al., 2022; Chowdhury et al., 2022; Jarrahi et al., 2023).

Another important aspect to mention is the dilemma in between automation and augmentation (Raisch and Krakowski, 2021; Brynjolfsson et al., 2023; Davenport and Ronanki, 2019). This dilemma is outlining. The principle of AI adoption where some tasks are fully automated, while others are augmented. These studies all agree on the establishment of a healthy balance in between the two.

2.6.3 Strategic Frameworks and Organizational Readiness

A very important aspect of any framework related to AI adoption must mention organizational readiness. Several frameworks to tackle this have already been established (Kurup and Gupta, 2022; Dasgupta and Wendler, 2019; Radhakrishnan et al., 2022; Madanchian and Taherdoost, 2025). Some of the key findings from these frameworks and studies include the intersection of having a clear process for transformation along with ethical AI adoption. They emphasize that focusing on long-term sustainability can only be achieved when an organization is ready. These frameworks operate on the principle that a company needs to be properly evaluated, processes need to be established and measured in terms of the capacity of the company, and only then a certain change can be implemented.

Certain frameworks are outlining the process of nonlinear adoption and flexible framework principles based on experimentation, validation, and scaling. These are valid for the rapidly changing environment of tech (Dasgupta and Wendler, 2019; Radhakrishnan et al., 2022; Madanchian and Taherdoost, 2025). This type of approach is heavily emphasizing the strong importance of proof-of-concept validation. By being able to iterate, test, validate, and distribute to customers, the organization learns and becomes ready over time. The intersection between these studies is the core idea that a proper governance structure and a strong executive sponsorship lead to a higher rate of adoption.

Some studies are also outlining the principle of having a strong ethical integration (Madanchian and Taherdoost, 2025; Chowdhury et al., 2022; Jarrahi et al., 2023; Pumplun et al., 2019). The ethical integration shall not be misled with the moral values of individuals. The principle of ethical integration in these studies is outlining that bias will be reduced with more data and more progress a company does. These studies are showing that with more data collected, bias is able to be reduced, and the rate of hallucination is correlated with hallucination in terms of the expectation from management.

As companies are moving towards more organizational readiness, a proper framework focused on risk management aligning with emerging regulatory requirements is needed (Parlov et al., 2025). This framework focuses on responsible AI deployment in the landscape of various regulatory challenges, such as the EU AI Act and ISO standards. The need to be compliant with the regulatory requirements in a given country is more important than ever before, with highly rising fees and penalties coming from governing bodies.

The last principle of organizational readiness comes with cost. Certain organizations try to push for radical change at times when the company is not ready for it, thus leading to a failed process of change management, wasted resources, and a decrease of trust from employees. As a company operates on the principle of a balance sheet, it is important to note that AI adoption is also working on the principle of cost as its main driver.

Several studies are noting that successful AI adoption can only yield a net positive outcome when the organization is ready for it (Agrawal et al., 2022; Coombs et al., 2020; Berente et al., 2021; Makarius et al., 2020). These studies are reporting on high productivity boosts and cost reduction with the use of AI automation augmentation, but mainly reskilling procedures.

2.6.4 Software Sector and Technology-Native Organizations

The software sector is the most relevant for the SAIL framework as it targets software organizations. Therefore, this section reviews already existing relevant frameworks that focus on AI adoption in software organizations and technology-native companies. A maturity model by (Garousi et al., 2025) is a revolutionary model based on the principle of assessing maturity for a cohesive effect between individual and organization. This model focuses solely on the principle of AI maturity rather than just assessing the technical capabilities.

The ability to combine both AI technical capabilities with organizational readiness showcases that it is possible to combine these two in order to create a cohesive system. It is very important to note that the model recognizes that software organizations require a tailored approach in terms of the assessment criteria as they often face unique challenges of integrating AI into their current processes. As each industry is different, it is harder to generalize throughout them. This framework specifically addresses software development companies from technical approaches such as machine learning models, automated testing, and pull requests. By highly focusing on the domain of technology, this model is very relevant.

In terms of generative AI adoption in software companies, a multiple case study was conducted (Kemell et al., 2025). This study focused on the same principle of moving from individual developer efforts into large organization orchestration. Throughout their research, it has been revealed that while many generative AI tools are providing significant value, it is necessary to approach this problem with a holistic framework encompassing reusable AI components, platforms, and managing technical debt, which often arises in AI-native environments. The conclusion of this research is that technology capabilities developed by individuals are often not sufficient to meet the expectations of management and can only be addressed when a strong framework comes into place and solves the question of how to develop AI software efficiently and scale it throughout the whole company. One of the key findings in this research is that AI product managers are becoming more and more relevant as they oversee the coordination between the data science teams, other management, and engineers.

The question of scaling was also thoroughly examined (Nwashili, 2025). This research focused on developing a product management-focused framework which addressed the challenges related to scaling AI systems. As this study was conducted in a larger company, it is more relevant to companies with higher human capital. This framework provided actionable steps to ensure that multiple product teams have efficient collaboration. This research focused heavily on the principle of platform thinking and allowing reusable AI components and platforms. The core of this framework was an AI platform which allowed multiple teams to develop on top of it at a single time. This framework becomes very relevant especially due to its key findings, which once again highlighted the need to have efficient AI product managers. The most important result from this research is the need to deploy with a systematic approach while focusing on not duplicating efforts.

Another AI-driven Enterprise Maturity Model has been developed and focused on the principle of utilizing maturity with progression (Ern et al., 2025). This maturity framework, specifically the AI-Driven Platform Enterprise Maturity Model (AIMM), once again mentions the platform principle (Yablonsky, 2021). It outlines that businesses should leverage AI as a core competitive advantage which would guide them on coordination of AI processes in a structured manner. Their framework focused on developing several platforms and entering these platforms once the teams become ready based on the internal evaluation metrics. This framework, however, emphasized deployment of real-time solutions due to the nature of the solution they were implementing. Therefore, the need to have appropriate maturity levels is very much justified. The key components of this research focused on AI automated testing and monitoring to ensure that the DevOps pipelines remain stable and the product meets the highest standards. This framework also mentions the Governance Board, which is a very important key factor in order to ensure that AI meets the compliance standards and does not violate any regulations and ethical principles.

The last proposed framework, called the Multidimensional Hybrid Intelligence Framework, focuses on creating a comprehensive development environment which addresses the core problem of integration of humans and artificial intelligence (Sherson et al., 2023). The unique aspect of this paper is that it applies this framework to several key standards in terms of the human-AI pattern. The purpose of this framework was to solve the question of creating hybrid intelligent systems which are structured in order to effectively combine the interaction between humans and AI to solve and automate software systems. Throughout this framework, they chose a multidimensional approach which encompassed both technical architecture for human-AI collaboration and an interface so that there is seamless interaction between the two entities. This model emphasizes quality assurance. These quality assurance mechanisms focus on improving the output generated by AI via continuous improvements and distributed tasks between humans and AI systems. The key result from this framework is that it successfully recognizes that software organizations must move beyond the typical viewing of AI as a replacement of humans and rather create a multi-environment hybrid space allowing humans to interact with AI in a cohesive manner. The ability to complement and not compete is the core principle of this framework.

2.7 AI Implementation and Scaling Frameworks: A Comprehensive Review

While the previous section examined general AI maturity models, this section focuses specifically on frameworks designed to guide the implementation and scaling of AI initiatives in organizations. Recent research has increasingly recognized that successful AI adoption requires more than maturity assessment—it demands structured approaches to navigate the complex journey from proof-of-concept to organization-wide deployment. This body of work directly informs the development of SAIL by highlighting critical success factors, common challenges, and proven strategies for AI implementation.

2.7.1 Phases of AI Implementation

The review conducted by (Haefner et al., 2023) have came to to the conclusion that the successful implementation of AI requires a structured, phase-based approach that addresses both technical and organizational dimensions. Their conclusion of a three-phase model consists of firstly proving the concept, followed by productionizing, and finally platformizing AI capabilities. Their findings are highlighting the socio-technical nature of AI adoption and emphasizing on the fact that both technical and organizational factors must be addressed simultaneously for an AI adoption to be successful. Another very valuable insight from their research is the fact that competition in AI adoption is fierce and organizations that fail to implement structured frameworks risk being outcompeted by firms which are more innovative and focus on integrating AI into their operations with a holistc approach. This insight will be very valuable for the SAIL framework as the vision of it is to be based on the socio-technical approach as well as it not only involves technical capabilities of the engineers, but also the aspect of change management and the natural resistance to change, which every organization is and will be facing due to human nature.

In terms of the proving phase, the research focuses on demonstrating the technical feasibility throughout and actual controlled experiments. This process suggests that organizations will need to validate any of their AI-based solutions before comitting to any of them majorly in order for them to not waste valuable resources which could potentially drain the company. The strategic deployment of AI requires understanding both the technological possibilities and business value creation mechanisms (Iansiti and Lakhani, 2020; Chui et al., 2018). Organizations must identify high-impact use cases and systematically evaluate their potential before scaling (Davenport et al., 2020).

2.7.2 Healthcare and High-Stakes AI Implementation

Industry where AI is currently being used heavily and has a very positive impact on is healthcare. This industry involves strong regulatory requirements and very high stakes for patients lives. The study from (Gama et al., 2022) focused on implementation of an AI framework into healthcare environment which very much differs from the software industry, but still provides valuable insights. Research has identified numerous barriers and facilitators specific to healthcare AI adoption, including technical infrastructure, clinical workflow integration, and regulatory compliance (Hassan et al., 2024). Additionally, marketplace dynamics and demand-side factors play crucial roles in healthcare AI adoption, requiring specialized frameworks that address the unique characteristics of healthcare ecosystems (Singh et al., 2025). The study outlines that trust with change adoption and major technological breakthrough tend to be mutually exclusive. Interesting insights from this study was the NASSS (Non-adoption, Abandonment, Scale-up, Spread, and Sustainability) which emerged and tries to dig deeper into failing of AI technologies in the medical industry.

As already mentioned that the software and healthcare very much differ the human capital stays the same. The responsible management of efficiently converting value to time stays the same. The study was mainly focusing on sustainability and scalability but not directly providing solutions. This will be the goal of the SAIL framework to show how framework can be still sustainable while scalable.

2.7.3 Organizational Adoption Barriers and Enablers

Several studies in the last few years have tried to examine the barriers which companies have to overcome in order to scale AI beyond the initial pilots. The latest exhibit being the study of (Praveen et al., 2024) which came to the conclusion that many companies fail to convert pilot to solutions due to issues of integration complexity, scalability issues and once again the resistance to change. Research on AI adoption in public sector organizations reveals similar challenges, with implementation success depending heavily on organizational culture, leadership support, and change management strategies (Chen and Gascó-Hernandez, 2024). This piece of work is very much emphasizing on the need to establish proper governance framework which fosters innovation and does not hurt motivation. The authors have found that successful AI transformation depends on creating an environment where experimentation is encouraged. This means that the failures are being treated as learning experiences rather than mistakes.

A case study from a very high-tech firm (Tamburri and Tonnarelli, 2022) published before the rise of AI shows us that having a structured roadmap helps them convert the resistance to curiousity. They are also very much advocating for active flexibility due to the sudden changes in the industry and clear guidance from the side of the framework need. This study also focuses on having a strong feedback mechanism. One of the key pillars of SAIL framework and its structure which will be outlined in future chapters. The deployment and operation of machine learning systems present unique challenges that require careful planning and systematic approaches (Baier et al., 2019).

2.7.4 Evidence-Based Approaches to Scalable Adoption

An evidence-based framework specifically focused on bridging the gap between AI ambition and scaled impact addresses a critical problem: many organizations successfully complete AI pilots but struggle to systematically embed AI into operations at scale (Pandiri, 2024). The author identifies three key gaps that prevent successful scaling: the strategy gap (misalignment between AI initiatives and business objectives), the execution gap (lack of operational capabilities to deploy AI at scale), and the culture gap (organizational resistance and insufficient change management). SAIL directly addresses these gaps through its Strategic Alignment stage, Scale Deployment processes, and emphasis on change management throughout the lifecycle.

The evidence-based approach emphasizes the importance of clear success metrics, structured decision gates, and mechanisms for knowledge transfer across teams. Organizations that successfully scale AI typically establish centers of excellence, develop reusable components, and create communities of practice that facilitate learning and collaboration. These practices are incorporated into SAIL's governance structure and reuse mechanisms.

2.7.5 Finance Sector and Domain-Specific Implementation

The financial services sector presents unique challenges for AI adoption due to regulatory constraints, risk management requirements, and the need for explainable AI in high-stakes decision-making. A modeling framework specifically for scaling AI adoption in finance has been developed, utilizing multi-agent orchestration approaches (Sepanosian et al., 2024). Their implementation study demonstrated that domain-specific considerations must be integrated into AI frameworks rather than treated as afterthoughts. Financial institutions require robust audit trails, model validation processes, and mechanisms to ensure fairness and prevent bias—requirements that extend to other highly regulated industries.

While SAIL is designed as a general framework for software organizations, the finance sector research underscores the importance of flexibility and adaptability. Organizations must be able to customize the framework to their specific regulatory context, risk tolerance, and business requirements while maintaining the core structure that ensures coordinated and scalable adoption.

2.7.6 Simplified Adoption Frameworks and Success Factors

Recognizing that overly complex frameworks can themselves become barriers to adoption, research has worked toward a simplified AI adoption framework that identifies essential success factors for implementing AI-based information systems (Kučević et al., 2024). Their design science research identified critical elements including executive sponsorship, cross-functional teams, iterative development processes, and clear value propositions. The authors argue that frameworks should provide sufficient structure to guide decision-making without imposing unnecessary bureaucracy that slows innovation.

This tension between structure and agility is central to SAIL's design philosophy. SAIL provides clear stages and decision gates while allowing organizations to iterate within stages, adjust timelines based on learning, and scale at a pace appropriate to their context. The framework is prescriptive enough to prevent common pitfalls but flexible enough to accommodate diverse organizational contexts and use cases.

2.7.7 Guiding Frameworks for AI Readiness and Piloting

A guiding framework focused on enabling successful AI adoption through careful assessment of organizational readiness has been developed (Amling, 2024). The framework emphasizes that readiness assessment is paramount before launching multiple AI initiatives simultaneously. Organizations must evaluate their data infrastructure, technical capabilities, talent availability, and cultural preparedness for AI adoption. The framework provides decision criteria for prioritizing AI use cases, determining optimal pilot scope, and establishing success metrics that balance technical performance with business impact.

This readiness-focused approach complements SAIL's Opportunity Scouting and Technical Feasibility stages, where organizations assess both the potential value of AI initiatives and their capacity to execute them successfully. SAIL extends this by providing explicit guidance on how to proceed when readiness gaps are identified—whether through capability building, external partnerships, or pilot scope adjustment.

2.7.8 The Influence of AI on Firm Scaling

Research exploring the relationship between AI implementation and firm scaling investigated how AI adoption influences organizational growth and scaling processes (Mecca, 2024). This research found that AI can both enable and constrain scaling, depending on how it is implemented. Organizations that successfully integrate AI into core business processes experience accelerated growth, improved efficiency, and enhanced competitive positioning. However, poorly planned AI implementations can create technical debt, organizational friction, and resource drains that actually impede scaling efforts.

This finding reinforces SAIL's emphasis on strategic alignment and architectural thinking from the outset. AI initiatives must be designed with scaling in mind from the beginning, rather than attempting to retrofit scalability after initial success. This requires considering factors like data architecture, model deployment infrastructure, and organizational change management as integral parts of the AI adoption process rather than secondary concerns.

2.7.9 Synthesis: Implications for SAIL

This review of AI implementation and scaling frameworks reveals several consistent themes that directly inform SAIL's design. First, successful AI adoption requires a phase-based approach that recognizes the distinct challenges of proving concepts, productionizing solutions, and platformizing capabilities. Second, socio-technical considerations are as important as technical ones—frameworks must address organizational culture, change management, and stakeholder engagement alongside technical architecture and data infrastructure. Third, governance mechanisms and clear decision gates are essential for managing risk and ensuring alignment with business objectives. Fourth, reuse and knowledge sharing across teams prevent duplication of effort and accelerate adoption. Finally, frameworks must balance structure with flexibility, providing clear guidance while allowing adaptation to organizational context.

SAIL integrates these insights into a comprehensive lifecycle framework specifically designed for software organizations. By combining the phase-based structure identified by Haefner et al., the readiness assessment approach of Amling, the barrier-focused perspective of Praveen et al., and the scaling emphasis of Pandiri and Mecca, SAIL offers a holistic approach to AI adoption that addresses both the technical and organizational dimensions of this complex challenge. The framework's eight-stage lifecycle, quality gates, and governance mechanisms provide the structure needed to coordinate AI initiatives across teams while maintaining the agility required in the fast-moving AI landscape.

2.8 Synthesis and Identified Gaps

In order to properly synthesize the different theoretical lenses and existing model, the strategy to look at the gaps which are currently existing will need to be applied. No model or theory is perfect and in certain instances the existing models are simply not covering the needs and requirements of the modern software organizations and their solutions. The new wave of AI technologies and their adoption simply falls beyond the scope of the existing models and theories. There are currently several gaps in the literature and practice regarding AI adoption in software organizations and approaches which the software organizations are currently tackling.

In the next section we will outline the 5 most crucial gaps which are not only in the existing literature but also in the current practice of software organizations.

- Lack of lifecycle orientation. This results in the fact the current models which are existing do not provide a concrete approach which is covering the entire system lifecycle and thus not outlining the necessary steps from the early initial awareness until the late stages of full integration and being AI-Native.

- Insufficient focus on coordination across teams and departments, leading to fragmented AI initiatives: In order for a model to be succesfully implemented and provide long term sustainable value, the coordination of AI initiatives across different teams and the knowledge transfer between them is beyong crucial as individual isolated efforts are not creating the desired value. The distributed nature of software organizations requires a structured approach to coordination. It is in fact natural for team to operate individually and focus on their own tasks, however from the perspective of achieving a common goal, this approach is not effective. Organizations must also address ethical implications and governance challenges throughout the AI lifecycle (Vidgen et al., 2020; Felzmann et al., 2019).

- Neglection of “Reuse Mechanisms”: Very important aspect of AI adoption and efficiency on the way towards it is being able to develope AI assets (models, data, knowledge) which are reusable across various teams and projects. This type of initiative is able to unleash an enormous amount of efficiency and time savings as individual temas are not required to reinvent the wheel, but simply reuse existing assets and alter them to their specific needs. It is in fact expected that each team will be requiring a certain level of customization, but by ivnesting into a modular design and reuse mechanisms, the need to reinvent simply becomes obsolete.

- Holistic integration of theoretical lenses: There are certain individual theories as already mentioned in the previous chapters which are providing valuable insights and are well adopted in practice. However, the holistic approach of integrating these insights into a coherent solidified framework is missing. The theoretical lenses outlined are proven to work and are widely adopted in practice, thus the integration of these theories into a single framework is expected to provide a significant contribution to both theory and practice.

- Scalability challenges: Scalability is a crucial aspect of any software solution. It is extremely important to consider the question of "Does it scale?" when designing and implementing any type of solution which is meant to be use by wide audience. This aspect is beyong important when dealing with AI adoption as we are not only dealing with resistance of change but also with the complexity of AI technologies (Keding, 2021). If the solution implemented will not be able to scale and the first impression will be negative the chance to further invest into its development and reclaiming the trust of the organization will be extremely low. Therefore, scalability needs to be considered from the very beginning and the solution needs to be designed in a way which is able to scale across different teams and departments.

By addressing these gaps a synthesis strategy will be applied to look at the key components of each theory and model, looking at how will it overlap with the SAIL framework, where will it diverge in terms of approach towards AI adoption and finally which gaps will be addressed by SAIL in order to provide a clear rationale for its development

Theoretical Lens / Model	**Key Components**	**Overlaps with SAIL**	**Divergences from SAIL**	**Identified Gaps Addressed by SAIL**
Enterprise Architecture (EA)	IT and business strategy alignment Systemic integration Governance TOGAF and Zachman frameworks	Blueprint for connecting AI to strategy Governance mechanisms promote alignment	Originates in pre-AI era Lacks explicit lifecycle orientation No reuse mechanisms	Lifecycle focus AI-specific governance Asset reuse across projects
Digital Transformation (DT)	Digital-first orientation Cultural and organizational readiness Maturity models (Deloitte DMM)	Phased maturity models Emphasizes readiness as critical success factor	Broad digital-change lens Not specifically focused on AI challenges	Aligns maturity thinking to AI lifecycle Granular adoption stages beyond general digital readiness
Innovation Diffusion Theory (IDT)	Adoption curve (innovators → laggards) Communication channels Social systems	Lens on AI adoption dynamics Highlights roles of innovators and early adopters	Focused on diffusion rather than organizational coordination or reuse	Mechanisms for scaling Coordination Knowledge reuse across teams

Theoretical Lens / Model	Key Components	Overlaps with SAIL	Divergences from SAIL	Identified Gaps Addressed by SAIL
Dynamic Capabilities Theory (DCT)	Sensing, seizing, transforming Adaptability in volatile environments	Agility lens for adopting and scaling AI	No structured procedure for AI lifecycle	Integrates sensing–seizing–transforming into AI lifecycle stages Defined governance and processes
AI Maturity Models (Deloitte, PwC, Gartner, Forrester)	Step-by-step assessment of AI readiness across: Strategy Governance Data Talent	Strong diagnostic tools Starting points for AI adoption	Static assessment focus Lack lifecycle depth No reuse strategies	Actionable lifecycle roadmap Supports adoption beyond maturity assessment into implementation

This comparative analysis clearly demonstrates the unique value proposition of the SAIL framework. By integrating insights from multiple theoretical lenses and addressing the specific gaps identified in existing models, which SAIL aims to fill.

The gap of not having the structured approach to the entire AI adoption lifecycle, from initial awareness to full integration is addressed by SAIL through its clearly defined stages of adoption (Awareness $\rightarrow$ Pilot $\rightarrow$ Scale $\rightarrow$ AI-Native).

Chapter 3
THEORETICAL FRAMEWORK

The following chapter will outline the theoretical framework of SAIL. Answering the questions like “Why does SAIL exist?”, “What theoretical foundations is it built upon?” and “How do these theories inform the design of the framework?”. The goal of this chapter is to provide a clear understanding of the theoretical underpinnings of SAIL and how are these theories being integrated into a cohesive framework.

3.1 Conceptual Foundations of SAIL

In order to properly understand the conceptual foundations of SAIL, it is extremely crucial to understand the purpose of a lifecycle adoption framework. The main purpose of SAIL and any lifecycle adoption framework is to provide a holistic and structured approach for a set of activities, in this case the adoption of AI in a coordinated and scalable manner. This aspect of lifecycle orientation is crucial as it is not narrowly focused on a single aspect of the adoption process, but looks on the landspace from an overarching perspective. The lifecycle orientation can be understood as a series of stages that an organization goes through in order to successfully adopt and integrate AI into its operations while approaching the topic from a comprehensive perspective.

3.1.1 Purpose of Scalable Artificial Intelligence Lifecycle (SAIL)

The purpose of Scalable Artificial Intelligence Lifecycle (SAIL) is to create a structured and systematic approach for software organizations to adopt, scale, and integrate AI technologies effectively. SAIL is aiming to serve as a roadmap that guides organizations through the complex journey of AI adoption and integration, ensuring that AI initiatives are aligned with business objectives, coordinated across teams, and capable of delivering sustained value.

SAIL framework will not serve as a rigid prescription, but as a practical playbook which will provide actionable steps for enterprises to follow. It will not be a theoretical model. It will not be a true "one-size-fits-all" solution. It will not be an assesment model. It will not be a maturity model. It will not be a diagnostic tool.

It will be a lifecycle-based roadmap outlining the stages of AI adoption, from initial awareness to full integration and being AI-Native, while staying true to solving the identified gaps and use cases in the operational context of the organization.

Enterprises are currently standing in front of a major technological shift, where AI is swiftly becoming a core component of business strategy and operations. The ability to effectively exploit and leverage this technology is becoming a key determinant of competitive advantage.

Efficiency is not secondary and innovation is not a luxury, but a necessity for survival in the modern digital economy. Being second is equivalent to being last.

3.1.2 Positioning Relative to Digital Transformation and Enterprise Architecture

While SAIL is conceived as an independent framework, it lies at the confluence of Digital Transformation and Enterprise Architecture (EA). To ensure that AI initiatives remain consistent with corporate strategies and organizational objectives, SAIL must rest on the principles of both Digital Transformation and EA.

Whereas companies will not transform themselves to fit the SAIL framework, the framework itself must adapt to the organizational context and structural realities. By applying EA principles, SAIL guarantees that AI initiatives are embedded within the larger enterprise architecture rather than developed in isolated silos. Since such global alignment is indispensable, it provides the foundation for scalability and acts as the enabler of coordination across organizational teams.

3.1.3 Addressing Fragmentation through Coordination and Reuse

Even when organizations produce strong ideas or embrace advanced technologies, such efforts inevitably fail without a clear coordination mechanism. Because coordination is frequently overlooked prior to innovation, promising concepts often remain unrealized, become fragmented, or duplicate existing efforts, thereby undermining potential synergies. By embedding coordination at the very outset of AI adoption, organizations establish the conditions for effective integration and sustainable scaling of AI initiatives.

Although AI adoption often appears to originate at the top, in reality it begins at the operational level, where individuals experiment with technology and emerge as its earliest champions. Only when senior management creates the necessary environment and allocates appropriate resources can these bottom-up innovations thrive. Since coordination and information exchange serve as the glue of the adoption process, they ensure alignment, integration, and synergy across organizational activities.

The conceptual foundations of SAIL will be built upon theoretical lenses which will be outlined in the following chapter. They will each provide a beneficial and crucial part of the puzzle and will be carefully integrated into the design of the framework.

3.2 Integration of Theoretical Lenses

Integration as a business process is the act of coupling different systems and processes together in order to seize value and synthesize a final form of the product. This type of synthesis involves cherry-picking the best parts of each individual component and linking them together in order to compliment each other and create a more valuable system.

SAIL framework will be built upon the integration of the theories analysed in the chapter of Literature Review, where the analysis of the existing literature and a brief introduction of the concepts was conducted. The answer which needs to be provided is how these theories support the design of the framework.

EA and SAIL

Enterprise Architecture (EA) will be the backbone of the creation of the SAIL framework. EA can be understood as the practice of analyzing, designing, planning, and implementing enterprise analysis in order to execute on business strategies. As SAIL framework is aiming to provide a structured approach to the adoption of AI in organizations, the similiarity of the outlined set of processe can not be overlooked. This approach in enterprise architecture is crucial for not only setting up initial organisation when going through the adoption process, but also whenever enterprises are scaling or implementing new technologies or processes. Regarding EA as already mentioned SAIL will focus more on the model of TOGAF as it is more flexible and adaptable to the current technological landscape and SAIL is expected to be more implementation oriented and practical especially in the context of digital business and companies which have started their digital transformation journey and are lookging to implement AI in their operations. SAIL will need to be aligned with the existing enterprise architecture of the organization and most importantly will need to complement it. The principles of a succesful EA will be the guiding principles for the design of SAIL and will need to truly ensure that AI will not act as a disruptive force, but rather as a true enabler of efficiency and innovation.

Digital Transformation and SAIL

Digital Transformation consists of the integration of digital technologies into all areas of a business, while fundamentally changing how the business operates and delivers value to customers. Digital Transformation is a major step for businesses which are not yet connected to the digital world and are still operating in a traditional way. Not every single business is required to undergo a digital transformation, but for those who do, it is a major step which requires a lot of resources and commitment. The root principle of Digital Transformation is the ability to not only adapt and embrace new technologies, but also to change the work culture in which the business and the teams whithin the business operate. This might be a challenging task, as it requires a lot of change management and the ability to overcome resistance. The SAIL as a framework will have the presumption that the organization is already digitally mature and has already undergone a digital transformation. By targeting digitally mature organizations, SAIL will be able to focus on the actual adoption of AI and not on the prerequisites which are required for a successful adoption. This will lead to efficient and streamlined internal process rollout and not wasting resources on the prerequisites. This also rules out the risk of failure due to lack of readiness and thus increases the chances of success. The principles of Digital Transformation are still going to hand in hand with SAIL, as SAIL will not be acting as a digital transformation framework, but rather as an AI transformation framework which is complementing the existing digital transformation efforts of the organization.

Innovation Diffusion Theory and SAIL

Innovation Diffusion Theory (IDT) is a theoretical framework which explains how, why, and at what rate new ideas and technology spread through cultures. The ideology underlying IDT and SAIL is innovation and the adoption of new technologies. The principles of IDT will be crucial for the design of SAIL, as it will provide a clear understanding of how AI adoption is happening in organizations and what are the key factors which influence the adoption process. From early adopters to laggards, the principles of IDT will be taken into consideration when discussing and implementing the collaboration and coordination mechanisms of SAIL. The principles of communication channels, social systems, and time will be crucial for the design of SAIL and its stages of adoption will be inspired by the IDT framework. SAIL will encompass the principles of IDT regarding reuse and knowledge transfer between teams and will provide a clear roadmap for the adoption process. IDT supports SAIL as SAIL is based on innovation and is one of its core principles. The innovation aspect in SAIL is the use case driven approach, where the adoption is driven by the actual use cases and their value proposition. This is a crucial aspect as it ensures that the adoption is not happening in a vacuum, but is driven by the actual needs of the organization.

Dynamic Capabilities Theory and SAIL

Dynamic Capabilities Theory (DCT) is a theoretical framework which is explaining how organizations can adapt and thrive in rapidly changing environments. DCT and SAIL are closely related as SAIL will integrate the 3 main components of DCT into its design and those being the sensing, seizing, and transforming. SAIL will be starting with the sensing of opportunities, where the organization will need to perform a short analysis and audit on the gaps which they will be need to fill in order to successfully adopt AI. This will be followed by the seizing of opportunities and prepare a framework which will serve as a benchmark for the actual adoption process. The final step will be the transforming of the organization, where efficient communication and implementation of the framework will be crucial for the success. The principles of DCT are in fact very crucial and relevant for the SAIL framework as they will inspire the design of the framework and SAIL will build upon the principles of DCT in order to provide a structured approach to the adoption process. The ability to sense, seize, and transform will be crucial for the success of the adoption process and will be the guiding principles for the design of SAIL.

In order to properly summarize the process of integration of theoretical lenses and their impact on the design of SAIL, the following table will outline each of the theories, their key concepts, design implications, and how they are being implemented in SAIL.

3.3 Principles Derived for Framework Design

The outcome of the synthesis of the theoretical lenses and the conceptual foundations of SAIL will be the derivation of the key principles which will guide the design of the framework. These principles will be crucial for the success of the adoption process and will ensure that the framework is aligned with the theoretical underpinnings. The key principles which will guide the design of SAIL are:

- Scalability: Scalability acts as a principle of the framework, as it will ensure that the adoption process is not happening in a vacuum and will be able to outgrow the initial adopters working on the AI solution during their pilot phase. The ability to scale the adoption process across different teams and departments will be crucial for the success of the adoption process and will ensure that the AI initiatives are able to deliver sustained value. Scalability will be achieved through the use of coordination mechanisms and reuse mechanisms which will be outlined in the following principles.

- Coordination: Coordination as an actor in the process has been already outlined in the previous chapter and will be crucial for the success of the adoption process as without a clear communication no message will be able to be delivered and the adoption process will fail with certainity. This polemical message is crucial for the

Table 3.1: Integration of Theoretical Lenses into the SAIL Framework

Theory	**Key Concept**	**Design Implication**	**Implementation in SAIL**
Enterprise Architecture (EA)	Business–IT alignment, governance, systemic integration	AI initiatives must align with organizational strategy and architectural principles	Strategic alignment checks at each SAIL stage; defined governance roles and architectural consistency across AI initiatives
Digital Transformation (DT)	Organizational readiness, digital maturity, cultural change	AI adoption requires phased organizational enablement beyond technology alone	Lifecycle progression with readiness considerations embedded in early SAIL stages
Innovation Diffusion Theory (IDT)	Adoption stages, role of innovators and early adopters	AI adoption progresses unevenly across teams and requires coordination mechanisms	Lifecycle stages supporting experimentation, validation, and structured scaling across organizational units
Dynamic Capabilities Theory (DCT)	Sensing, seizing, transforming	AI adoption must remain adaptive in dynamic and uncertain environments	Iterative feedback loops and decision gates enabling reassessment and transformation across SAIL stages
AI Maturity Models	Diagnostic assessment of AI readiness	Readiness assessments must translate into actionable implementation guidance	SAIL extends maturity diagnostics into an operational, end-to-end AI lifecycle

design of SAIL and insisting on the importance of coordination will be the one of the most detrimental aspects of the framework.

- Value-driven use case prioritization: The adoption process will be driven by the actual use cases and their value proposition. This will be achieved by the initial DCT analysis which will be performed at the beginning of the adoption process by a small team of experts. The use cases will be prioritized based on their feasibility, impact, and alignment with the overall business strategy. This will ensure that the adoption process is not happening in a vacuum and is driven by the actual needs of the organization. The early adopters acting as the champions of the adoption process will be crucial for the success of the adoption process and will ensure that the AI initiatives are able to deliver sustained value.

- Mechanisms for capturing and reusing AI assets: The ability to capture and reuse AI assets (models, data, knowledge) remains as an important factor of the framework and the ability to repurpose AI initiatives and being able to branch from the initial use case will be very beneficial for the success of the adoption process as the teams will be able to save time and solve multiple usecase withe the same AI asset. This will ensure that the adoption process is not happening in a vacuum and the AI asset will be able to deliver value accros multiple divisions and departments. The reuse mechanisms will be crucial for the success of the adoption process and will ensure that the AI initiatives are able to deliver sustained value.

Chapter 4
RESEARCH METHODOLOGY

The upcoming chapter will focus on the research methodology which will be used for the development of the SAIL framework. This chapter will outline the research design, literature review approach, framework development process, and conceptual evaluation via use cases. Research methodology is a crucial aspect of any research project, as it is going to define and establish the credibility and validity of the research findings. The chosen methodology will be aligned with the research question and objectives, ensuring that the research is able to provide a solid an comprehensive answer to the research question of “How can software organizations effectively adopt and scale AI technologies through a structured lifecycle framework?”.

4.1 Research Design – Design Science Research

This thesis is using the model called Design Science Research (DSR) as the research methodology. DSR is a research paradigm which is focused on the creation and evaluation of artifacts (models, methods, frameworks) in order to solve real-world problems. This thesis is focused on the creation of a conceptual framework (SAIL) that solves a real-world problem (AI adoption in software organizations). The DSR methodology is particularly well-suited for this research as it provides a structured approach for the development and evaluation of the framework. The target of this thesis is not literature review, nor is it a metanalysis of existing frameworks. The primary and sole goal of this thesis is the resugence of a new framework which is able to solve the identified gaps and use cases in the operational context of the organization.

The DSR methodology consists of three main cycles: relevance cycle, rigor cycle, and design cycle. The relevance cycle is focused on the identification of the problem and the context in which the problem exists. The rigor cycle is primarily focused on the review of existing literature and theories in order to provide a theoretical foundation for the framework that is being developed. The last stage of the DSR methodology is the design cycle, which is focused on the actual development and evaluation of the framework. The last stage will be the most important one, as the actual process of pitching and outlining the framework will take place. The evaluation of the framework will be done through the use of realistic use cases, which will be outlined in the following chapter. The use cases will be used to evaluate the framework against multiple criteria, such as coherence, scalability, and reusability. The use cases will be selected based on their relevance to the research question and their ability to provide a comprehensive evaluation of the framework. The use cases will be based on realistic scenarios that software organizations might face when adopting and scaling AI technologies.

To adhere to the DSR methodology, the research will be conducted in a structured manner, following the three cycles outlined above.

4.1.1 Relevance Cycle

The relevance cycle will be focused on the identification of the problem and the context in which the problem exists. The problem of AI adoption in software organizations has been identified as a real-world problem that needs to be addressed. The context in which the problem exists is the software industry, which is rapidly evolving and adopting new technologies. The relevance cycle will also involve the identification of the stakeholders who are affected by the problem, such as CTOs, innovation managers, and engineering leads. The relevance cycle will provide a clear understanding of the problem and its context, which will inform the development of the framework. Relavance as a cycle will be outlined in the SAIL framework as the first stage of the adoption process, where the organization will need to perform a short analysis and audit on the gaps which they will be need to fill in order to successfully adopt AI. This will prove whether the organization is ready to adopt AI and will provide a clear understanding of the problem and its context, which will inform the development of the framework and therefore the relevance cycle is being integrated into the design of SAIL.

Connection of Relevance cycle to SAIL: - SAIL is addressing fragmentation through coordination and reuse, which is a real-world problem that needs to be addressed. - The Relevance cycle emphasizes on the importance of understanding the context in which the problem exists, and on the practical needs - Use cases will act as a bridge between theory and practice, ensuring that the framework is relevant to the needs of software organizations.

The Relevance Cycle guided SAIL's development, by being able to identify the aoption challanges while ensuring that the framework is addressing the practical needs of software organizations.

4.1.2 Rigor Cycle

The goal of the rigor cycle is to provide a solid theoretical foundation for the framework that is being developed. The rigor cycle is involving the review of existing literature which was done in the chapter of theoretical frameworks where a thorough analysis of the existing literature and theories was conducted and its connection was established and the gaps were identified. By being able to connect the existing theories to the design of the framework, the rigor cycle increases on the validty as the framework is not being developed in a vacuum, but is based on solid theoretical foundations. Rigor cycle basis its foundational principles on the integration of multiple theoretical scopes and lenses into one meta-model which serves as the backbone of the framework. The principles derived from the synthesis of the theoretical lenses will be crucial for the design of SAIL, as they will provide a clear understanding of how to leverage the internal resources of the organization in order to successfully adopt AI and create value.

Connection of Rigor cycle to SAIL: - Chapter of theoretical frameworks provided a solid theoretical foundation for the framework that is being developed. - The listed theories of EA, Digital Transformation, IDT and DCT serve as information sources for the design of SAIL. - Ensures that SAIL is not arbitrary but is grounded in established knowledge.

The Rigor Cycle ensured that SAIL is grounded in established knowledge, by providing a solid theoretical foundation for the framework that is being developed.

4.1.3 Design Cycle

The design cycle or otherwise known as the build and evaluate cycle will be the cycle which lays its scope on the foundational principles of the SAIL framework. The design cycle is focused on the actual development and evaluation of the framework. The primary focus of this thesis is the development of the framework, which will be done in a structured manner, in the upcoming chapter. The evaluation of the framework will be done through the use of realistic use cases where each use case will be evaluated against multiple criteria and will be specified on how to overcome it in a structured manner. The use cases will be selected based on their relevance and the possibility of them occuring in a real-world scenario in the ambit of software organizations. The foundation of the model will be based on the principles derived from the synthesis of the theoretical lenses, however as it is a new framework, it would be unprofessional to limit the level of innovation and creativity by simple observing the existing theories. SAIL as a framework will be innovative and creative in its design, while still being grounded in established knowledge. While the theoretical lenses provide the foundation, the novel contributions of SAIL are a product of independent thought and design. This balance between theory and innovation is crucial for the success of the framework, as it ensures that the framework is both relevant and practical. By pushing the boundaries of existing knowledge, SAIL aims to provide a new and breakthrough approach to AI adoption in software organizations.

Connection of Design cycle to SAIL: - The design cycle is focused on the actual development and evaluation of the SAIL framework which is the primary focus of this thesis. - Use cases will be used in the evaluation process of the framework, ensuring that the framework is practical and relevant. - While grounded in theory, SAIL incorporates innovative elements that extend beyond existing models.

The Design Cycle facilitated the creation of SAIL. It is inevitable that the design of SAIL will be innovative and creative, while still being grounded in established knowledge and the process of outlining followed by the evaluation of the framework will be done through the use of realistic use cases.

4.2 Literature Review Approach

The question of why each of the theories was chosen and how they are connected to the design of SAIL was already answered in the chapter of theoretical frameworks, however in order to provide a clear understanding of the literature review approach, the following segment will elaborate on the search strategy, inclusion/exclusion criteria, and the process of synthesizing theory.

4.2.1 Search Strategy, Databases, Keywords

The search strategy for the literature review was focused on identifying relevant articles, models, and frameworks that are addressing the topic of AI adoption in software organizations, which is the primary focus of this thesis. The search was done over multiple databases, including Google Scholar, IEEE Xplore, ACM Digital Library. These databases were helpful in providing a wide range of articles and models from both academic and industry sources. The search was not only limited to academic articles, but also included industry reports and whitepapers from leading consulting firms such as Deloitte, PwC, Gartner, and Forrester.

The problematic aspect of AI adoption as already mentioned is a very recent topic and therefore the most relevant and up-to-date information can be found either by direct reports of large consulting firms or in conference papers and articles or journals which are focusing on the topic of AI adoption and its challenges.

The keywords used in the search included combinations of terms such as “AI adoption”, “Artificial Intelligence implementation”, “Digital Transformation” and other outlined frameworks. Regarding the research on the already existing frameworks, the process was more straightforward as the frameworks are already established and therefore the search was more focused on finding the most relevant and up-to-date information about the frameworks and their principles.

4.2.2 Inclusion/Exclusion Criteria

The inclusion criteria for selecting articles and models were based on their relevance to the research question, publication date (preferably within the last 10 years to ensure contemporary relevance), however this was not a strict rule as some of the theories are older but still relevant such as the IDT. Inclusion of the articles was also based on their citation count, with a preference for highly cited works which indicates their influence and acceptance not only in the academic community but also in the industry.

4.2.3 Process of Synthesizing Theory

The process of synthesizing theory through the literature review involved a thorough analysis of the selected articles and models. The main motivation behind the synthesis was to identify the key principles and concepts that are relevant which are relevant in the context of AI adoption in software organizations. The main contribution of the synthesis was the identification of the gaps and the ideation process started with the cardinal question of “Where do existing frameworks fall short in addressing the challenges of AI adoption in software organizations?” This question was crucial for the design of SAIL, and the frameworks which are crucial and are the cornerstone of the design of SAIL. There were many more frameworks which were analysed, but the ones which were chosen are the ones which are most relevant and are truly the depiction of the vision of the SAIL framework. Frameworks which were not chosen were either too generic and did not provide any specific insights into the topic of AI adoption or were too specific and did not provide a holistic view of the adoption process. In the end the list of the theories which were chosen are the ones which are most relevant and are truly the foundation of the design of SAIL, both from the perspective of theory and practice.

4.3 Framework Development Process

The framework development process will be based on an iterative design and refinement approach. Multiple iterations of this framework were done in order to ensure that the framework is practical and relevant and the final version of the framework is the one which is able to solve the identified gaps and use cases in the operational context of the organization, which is the one that will be presented in the upcoming chapter. This process of the development and the refinement of the framework was done in a structured manner, following the principles derived from the synthesis of the theoretical lenses. Each of the theoretical lenses defined in the chapter 3 will need to be connected to a realisitc part of th SAIL framework, otherwise the point of the chapter 3 is simply lost. The following segment will outline the mapping of the theories to practical framework elements.

- Enterprise Architecture (EA) → systemic integration: EA principles are guiding the design of SAIL to ensure that AI initiatives are embedded within the larger enterprise architecture rather than developed in isolated silos. The improtance of this theoretical framework will play part on effective integration and sustainable scaling of AI initiatives.

- Digital Transformation → readiness and strategic alignment: SAIL presumes that the organization which shall adopt the framework is already digitally mature and has already undergone a digital transformation. This leads to efficient and streamlined internal process rollout and not wasting resources on the prerequisites, such as data infrastructure and digital culture.

- Innovation Diffusion Theory (IDT) → adoption dynamics across teams: IDT does not only serve as the amplifier of the innovation aspect of SAIL and the use case driven approach, but still gives the framework a clear understanding of how AI adoption is happening.

- Dynamic Capabilities Theory (DCT) → agility in scaling and transformation: The principles of sensing, seizing, and transforming are guiding the design of SAIL and are integrated into the stages of adoption. The ability to sense, seize, and transform will be crucial for the success of the adoption process and will be the guiding principles for the design of SAIL.

Each of these theories is a unique piece of the puzzle and when combined together they are able to create a ruleset and outline the scope of the SAIL and give an estimate on the ability of the framework to solve the identified gaps and use cases in the operational context of the organization.

4.4 Conceptual Evaluation via Use Cases

Extremely crucial part of the DSR methodology is the evaluation of the developed framework. Great idea is only as great as the strongest mechanism which is validating it. The larger and harsher the validation mechanism is, the more credible and valid the idea becomes. By testing the framework against realistic use cases which are not only focused on extreme scenarios, but also on the more common ones, the framework is able to prove its versatility and applicability in a wide range of scenarios. The framework will most likely not be able to solve every single use case which is thrown at it due to extremely large amount of unknown and unpredictable factors which are influencing the adoption process. However, by trying to solve a wide range of use cases, which occur at the companies will give a clear understanding of the strengths and weaknesses of the framework and will provide a solid foundation for future research and development of the framework. The interpretation of the results will be crucial for the success of the framework, as it will provide a clear understanding of the strengths and weaknesses of the framework and will inform future research and development of the framework. The outcomes will need to be interpreted in a structured manner with the ability to reduce a bias and provide a clear understanding of the strengths and weaknesses of the framework. By having a great suite of testing scenarios, the framework is able to prove its versatility and applicability in a wide range of scenarios. The choice of usecases might be even more crucial than the actual design of the framework, as the use cases will be the ones which will validate the framework and will provide a clear understanding of its strengths and weaknesses. Seeing from which aspect the framework is outperforming and from which aspect it is underperforming will be crucial for the future development of the framework and will provide a solid foundation for future research and development of the framework and its adoption in the real-world scenarios.

Chapter 5

Framework Development (SAIL)

5.1 Overview of SAIL Framework

The Structured AI Lifecycle (SAIL) framework is a nexus between the theoretical foundations of AI adoption and the practical realities faced by software organizations. SAIL is designed to provide a structured approach of company-wide AI adoption, ensuring that business value is maximized while being focused on the operational aspect of the business. By building on the theoretical foundations outlined in chapter 3 and methodological rigor from chapter 4, SAIL operates in a manner of defining clear design principles and practical steps such as scalability, coordination, value-driven use case prioritization, reuse mechanisms, and governance structures.

SAIL is structured into eight-step lifecycle stages, each addressing a critical phase of AI adoption. The AI adoption journey encompasses stages from early initial opportunity identification to full AI-native integration and optiomization. Each stage is designed to solve a specific set of challenges that organizations typically face during their AI adoption journey. None of these stages exist in isolation; rather, they are interconnected and their combined effect is greater than the sum of their parts. The butterfly effect of each stage is crucial as individual decisions on stage level amplify the overall success of the AI adoption journey.

The north star of SAIL is to ensure that AI initiatives are not only successfully launched but are also scalable, coherent, and reusable across the organization. Furthermore SAIL will be presented in a visual diagram which will outline the different stages of the framework and their interconnections. The visual representation will provide a clear understanding of the framework and will serve as a reference point for the implementation of the framework in real-world scenarios. The SAIL framework is iterative and flexible, allowing organizations to adapt it to their specific contexts while adhering to the core principles of effective AI adoption and therefore the need to have edge cases and deviations from the outlined path is inevitable.

SAIL framework will be evaluated through the use of 10 realistic use which are going to be outlined in the upcoming chapter. This will serve as the empirical part of the DSR methodology and will provide the necessary validation for the framework.

This chapter will outline the SAIL framework in detail, describing each stage, its objectives, key activities, roles involved, decision points, and expected outputs. This chapter will not only propose the framework but will also dig into the practicalities of its implementation, governance and knowledge reuse mechanisms. The goal is to provide a comprehensive guide that organizations can follow to navigate the complexities of AI adoption effectively.

5.2 Lifecycle at a Glance

- Use a diagram here (8-step lifecycle with feedback loop).

At a high level, the SAIL framework consists of the following eight stages: 1. **Opportunity Scouting:** Identifies and prioritizes high-impact AI opportunities where the impact of AI can be maximized while the effort to implement is minimized. This stage involves planning, idea collection, initial value estimation and effort estimation. Effort vs impact matrix will be used to prioritize the use cases. 2. **Strategic Alignment Review:** Ensures that selected opportunities align with business strategy and have stakeholder buy-in. This stage involves stakeholder review, alignment workshops, and strategic fit analysis, which are crucial for the success of the adoption process. 3. **Feasibility & Risk Assessment:** Assesses technical debt, organizational readiness, and risks which are inherent to the adoption process. The objective of this stage will be to identife whether the use case is feasible and whether the risks are manageable. 4. **Pilot Design:** Plans the pilot implementation, defining scope, success metrics, and resource allocation. The aim of this stage is to ensure that the pilot is well-defined and has clear success criteria. Throughout this stage there is not a single pilot being designed. In ordert to be as efficient as possible, multiple pilots are being designed solving different use cases. This will ensure that the organisation is able to diversify its AI portfolio and is not putting all its eggs in one basket. 5. **Pilot Execution:** Builds and tests the AI solution in a controlled environment, collecting user feedback and iterating as needed. The purpose here is that the pilot is able to solve and deliver painpoints outlined in the initial stages. The pilots will need to be evaluated against the success criteria outlined in the planning stage. Focus on quick wins and fast iterations to build momentum is the key to success. 6. **Evaluation & Scaling Decision:** Evaluates pilot outcomes against success criteria to decide on scaling, pivoting, or terminating the initiative. This stage serves as a make it or break it point for the adoption process. Throughout this phase the decision of whether to scale the pilot, iterate on it, or abandon it will be made. The decision shall not be taken lightly, and the emotions shall be taken out of the equation. The only thing that matters is the value delivered and the potential for future value. 7. **Scale Deployment:** Rolls out the solution organization-wide, managing change, training, and integration with existing systems. The objective of this stage is to ensure that the solution is successfully scaled across the organization and is able to deliver sustained value. Change management and training will be crucial for the success of this stage. 8. **AI Native Optimization:** Integrates, optimizes, and continuously improves the AI solution, embedding it into business processes and culture. The final stage of the adoption process will be the optimization and continuous improvement of the AI solution. The objective of this stage is to ensure that the AI solution is able to deliver sustained value and is integrated into the business processes and culture of the organization.

The outlined set of stages is designed to be flexible and adaptable, allowing organizations to tailor the framework to their specific contexts while adhering to core principles of effective AI adoption. The driver force of rapid innovation and value-based delivery principle enhances the long-term success of AI initiatives. In order to understand the framework in a more structured manner, a diagram will be presented to illustrate the stages, their interconnections, and the overarching governance structure.

[Insert Diagram Here: 8-step lifecycle with feedback loop]

5.3 Governance Layer Across Stages

Governance as a concept is inevitable for the success of the adoption process and will be embedded across all SAIL stages, ensuring oversight, alignment, and conflict resolution. Governance is a term often misunderstood and misused, but in the context of SAIL, governance is not about bureaucracy or red tape. It is about ensuring that the adoption process is aligned with the overall business strategy and that the key decision points are made in a structured and transparent manner. If the governance mechanisms are not in place, the adoption process has an increased chance of failure due to the simple principle of not following a structured approach. The successful governance mechanisms encompasses several key stakeholders and roles with each having a clear understanding of their responsibilities and decision-making authority. The key roles involved in the governance of SAIL include: - **Sponsor:** Champions the AI adoption initiative. Their responsibilities include securing resources, advocating for the initiative at the executive level, and ensuring alignment with strategic objectives. These sponspors are unique individuals who are able to see the big picture and are able to drive the adoption process forward. Their crucial role in the process is the ability to secure resources and advocate for the initiative at the executive level. Their work is mostly seen in the initial stages of the adoption process, but their influence is felt throughout the entire lifecycle. They are usually the individuals who are well spoken, value oriented and are able to present and sell the idea of AI adoption in a structured manner. - **Governance Board:** A cross-functional team, being responsible for approving/rejecting use cases. Often might be seen as a bottleneck, but their role is crucial for the success of the adoption process. Their key responsibilities include reviewing use cases for strategic fit, resolving conflicts (e.g., overlapping pilots), and ensuring that the adoption process adheres to organizational policies and standards. This board shall be composed of senior analysts and engineers together with business stakeholders who are able to provide a holistic view of the adoption process and value delievery. Senior analysits providing the analytical perspective and potential bottlenecks, opportunities and directions for the AI solution while the business stakeholders are able to provide the business perspective and roadmap allignment with the overall business strategy. - **Adoption Manager:** Oversees the day-to-day execution of the SAIL framework. Their responsibilities include coordinating activities across stages, managing timelines, and reporting progress not only to the sponsor but also to the governance board and other stakeholders such as team leads and managers. The adoption manager is the glue holding the entire adoption process together, while ensuring that deadlines are met and that the communication is clear and transparent. The connection between sponsor and adoption manager will be a holding point as they are both focused on the success of the adoption process, but from different perspectives. The sponsor is focused on the strategic aspect while the adoption manager is focused on the operational aspect. - **Teams/Pilot Teams:** Execute tasks within each stage, providing feedback and insights. Their primary responsibility is the implementation of the AI solution outlined. They are not only the key to the value creation but also the key to the success of the innovation and idea generation. Teams implement the ideas throughout their day to day work and are the ones creating the PoCs. The very important aspect of the teams will also be in the idea generation process. In a software companies engineers are the majority of the workforce and vale creation is happening through their work. By letting engineers draft and propose usecases which are optimally company wide painpoints, the AI adoption process is able to solve painpoints which are actually existing and are not just theoretical and have an impact on the day to day work of all engineers. Engineers are the ones who are able to see the painpoints and are able to propose solutions which are actually solving internal bottlenecks and by letting the pilot team try to solve these painpoints with AI. The overall company is able to yield the benefits of the AI adoption process and is able to deliver same value in shorter timeframes allowing for more innovation and creativity. - **Team Leads/Managers:** Facilitate team involvement, ensure resource availability, and support change management during scaling. The Team Leads and Managers play a very important role in the adoption process and their integration in it is pivotal for the success of the adoption process. The Sponsor is negotiating team capacity for the pilot teams and their engineers as the engineers are not expected to work on AI initiatives full time, but rather as a side project. The Team Leads and Managers are the ones who need to adhere and ensure that the capacity is available and that the engineers are able to dedicate time to the AI initiatives. Without the support of the Team Leads and Managers, the adoption process will fail as engineers will be forces to work on the AI initiatives outside of their working scope and the overall quality of the work will be diminished. The Team Leads and Managers are the ones who are able to provide the necessary support and resources for the success of the adoption process.

5.4 Key Decision Checkpoints

As already mentioned earlier, the SAIL framework incorporates key decision checkpoints and works with dynamic events and processes which are influencing the adoption process. There is a set of key decision checkpoints which are crucial for adoption process and will serve to determine whether the initiative is viable and should proceed to the next stage. After each of the key decision checkpoints, there is a possibility to either proceed to the next stage, iterate on the current stage, or abandon the initiative altogether. Without successful governance mechanisms, these decision points can become bottlenecks or sources of conflict, hindering progress and diluting value. Every single one of the stages has its own decision points and shall only be proceeded to the next stage if the decision points are successfully passed. By having clear decision points, the teams and the organisation are able to reduce the risk of failure, track the progress are receive feedback on the progress of the adoption process not only from customer's persective but also from the internal stakeholders. As already very briefly mentioned, each of the stages is sequential but iterative, meaning that the stages are not linear and there is a possibility to go back to previous stages if the need arises. Before entering the next stage, the decision points need to be successfully passed and a question of whether to proceed, iterate, or abandon needs to be answered. The 3 possible outcomes of each decision point are: - **Proceed:** This initiative meets the strategic, technical, and operational criteria in order for the team to move to the next stage. The proceed option can only be chosen if the initiative is able to deliver value and is aligned with the overall business strategy. The proceed option is the most desirable outcome as it indicates that the initiative is on track and is able to deliver value. - **Iterate:** The initiative shows promise but requires further refinement. Either in the form of additional research, pilot adjustments, or stakeholder engagement. The iterate option is chosen when the initiative is not yet ready for it to be proceeded into the next stage. The ability to iterate is crucial as it involves dynamic requirements and does not accept mediocrity. The option to iterate involes a 2 man's game where the manager and the team are able to work together in order to refine the initiative and make it ready for the next stage. - **Abandon:** The initiative does not meet the necessary criteria and should be discontinued to avoid further resource expenditure. This stage does not always indicate that the initiative is a failure, but rather that the initiative is not aligned with the overall business strategy or is not able to deliver value. This means that the implementation of the initiative is not feasible or might not be needed anymore. In the best case abonded initiatives are the ones which are able to provide learnings and insights which can be used in future initiatives.

In order to properly decide whether to move to the next stage, the correct questions need to be asked and answered. The following segment will outline the key decision points for each of the stages:

Stage	Key Decision Checkpoint	Possible Outcomes	Governance / Roles	Notes / Edge Cases
1. Opportunity Scouting	Is the use case strategically relevant and feasible?	Proceed Iterate Reject	AI Sponsor Business Analyst EA Lead	Multiple overlapping opportunities → consolidate portfolio Unclear ROI → iterate
2. Strategic Alignment	Does the initiative align with corporate strategy and transformation goals?	Proceed Iterate Reject	Steering Committee CTO	Conflicting priorities → defer or adjust scope
3. Technical Feasibility & Data Assessment	Are risks manageable? Is the use case feasible?	Proceed Iterate Reject	Risk Officer IT Department Adoption Manager	Legacy systems or missing data → iterate Severe gaps → reject
4. Pilot Design	Is the pilot implementation plan feasible and properly scoped?	Proceed Iterate Reject	Pilot Manager Team Leads Governance Board	Dependencies on other pilots → integrate Lack of resources → iterate
5. Pilot Execution	Does the pilot meet adoption and performance criteria?	Proceed Iterate Reject	Dev Team Data Scientists Adoption Managers	Low adoption → iterate with change management Technical failure → iterate or reject
6. Evaluation & Scaling Decision	Should the initiative be scaled across the organization?	Proceed Iterate Reject	Steering Committee Governance Board	Partial success → phased scaling Multiple pilots succeed → portfolio prioritization

Stage	Key Decision Checkpoint	Possible Outcomes	Governance / Roles	Notes / Edge Cases
7. Scale Deployment	Is full deployment technically and operationally feasible?	Proceed Iterate Reject	Adoption Manager IT Ops Department Leads	Bottlenecks or workflow conflicts → iterate Systemic risks → reject
8. AI-Native Optimization	Is continuous improvement and adoption sustainable?	Proceed Iterate Reject	AI Governance Board Process Owners	New use cases emerge → loop back to Step 1 Tech upgrades → iterate

Each of the questions which are outlined per step were consciously chosen in order to target the underlying challenges which are inherent to each of the stages. The questions per stage try to target the main risk and challange connected with the stage and thus by having a clear understanding of the challenges, the teams are able to mitigate the risks and are able to make informed decisions. The possible outcomes of each decision point are designed to provide flexibility and adaptability, allowing organizations to respond to new insights or challenges as they arise. By incorporating these decision checkpoints, SAIL ensures that AI initiatives are continuously evaluated for their strategic fit, technical feasibility, and operational readiness, thereby enhancing the likelihood of successful adoption and sustained value delivery. In the table each question and step has an inherent connection to the relevant stakeholder and roles which are involved in the decision-making process. By having clear roles and responsibilities, the decision-making process is able to be streamlined and is able to reduce the risk of conflicts and bottlenecks. This also allows clear communication and transparency throughout the entire adoption process as each stakeholder knows where do they stand in the decision and accountability process. The table is also enhanced by a column of notes and edge cases which are outlining the potential edge cases and scenarios which might occur during the decision-making process. These are serving as a potential and expected bottleneck which the company might be facing during the adoption process. The edge cases such as overlapping pilots, conflicting priorities, legacy systems or missing data are very common in the adoption process and by being aware of these potential bottlenecks, the teams are able to plan ahead and reduce the risk of failure and increase the probability of successfully passing the decision points. Each 8 step decision cycle presents a unique set of challenges and opportunities, and by having clear decision points, roles, and potential edge cases, the SAIL framework is able to provide a structured and transparent approach to AI adoption in software organizations. This means that after each cycle the ability to loop and start from the beginning is possible and is even encouraged as the company might gather more Use Cases, knowledge and identify new fleet of opportunities which are able to be solved with AI. The dynamic decision making framework allows the company to be agile and responsive to the ever-changing landscape of AI technology and its applications in the software domain while ensuring the process of governance and structured approach is not lost. The strategic placement of the decision points is crucial as they are placed in the most logical and impactful stages of the adoption process. The decision points are placed after individual stages where the most critical decisions need to be made, ensuring that the adoption process is continuously evaluated and aligned with organizational goals.

5.5 Detailed Adoption Stages

In the following subsections, each of the eight stages of the SAIL framework will be outlined from the perspective of objectives, key activities, roles involved, decision points, and expected outputs. This detailed breakdown will provide a comprehensive understanding of the necessesity of each stage and how they interconnect to form a cohesive AI adoption lifecycle. By granularly defining each stage and breaking it down into its core components and the 5 primary elements, the analysis will have a cohesive and structured approach to understanding the underpinning principles of each stage and how they contribute to the overall success of the AI adoption journey. This detailed breakdown will serve as a practical guide for organizations looking to implement the SAIL framework, providing clarity on what to expect and how to navigate each phase effectively.

5.5.1 Step 1: Opportunity Scouting

Opportunity scouting is the initial stage of the SAIL framework, where the focus is on identifying and prioritizing high-impact AI opportunities within the organization. This stage will be setting the foundation of the AI revolution in the company. It is the stage where the ideas of the future are being born and where the potential for value creation is being identified. The objective of this stage is to identify the initial set of painpoints which the company is dealing with at the current day and age. The idea generation will be led in a bottom-up manner, where the engineers and the teams are able to propose use cases which they are facing in their day to day work. By giving the engineers a say and voice the company is receiving direct feedback from the people who are actually doing the work and are able to identify the painpoints and bottlenecks which are hindering their productivity. The idea generation will be complemented by a top-down approach, where the management and the business stakeholders are able to propose use cases which are aligned with the overall business strategy and goals. By creating a hybrid approach where idea generation comes from engineers and is evaluated by upper management overseeing the vision, mission and roadmap of the company, the company is able to create a balanced portfolio of AI initiatives which are able to deliver value in the short, medium and long term. The idea generation will be complemented by a structured approach of evaluating the ideas based on their potential impact and effort required to implement them. The ideas will be evaluated based on an effort vs impact matrix, where the ideas with the highest impact and lowest effort will be prioritized for further evaluation. This structured approach will ensure that the company is able to focus its resources on the most promising opportunities and is able to deliver value in a timely manner. The idea generation process shall be led by the Adoption Manager and the Sponsor, who will be responsible for collecting and evaluating the ideas and ensuring that they are aligned with the overall business strategy and goals. The preferred outcome of this stage will be a shortlist of candidate use cases which are able to deliver value and are aligned with the overall business strategy and goals. This process shall be done in an open manner where engineers have the freedom to propose ideas and are not limited by the current technological capabilities or the current state of the organization. In order to ensure that the idea generation process is as efficient as possible, the Adoption Manager and the Sponsor shall prepare a set of categories and themes which are aligned with the overall business strategy and goals. Categories such as (e.g., customer support, internal efficiency, product development, workflow automation) will be used to guide the idea generation process and give engineers in the ideation round a spark and direction in which they see potential for AI to be implemented. The idea generation process shall empower the engineers and the teams to think outside the box and propose ideas which are not only solving current painpoints but are also pushing the boundaries of what is possible with AI.

- **Objective:** Identify and prioritize high-impact AI opportunities.

  The objective of this stage is to identify and prioritize high-impact AI opportunities inside of the organisation which are solving a real painpoint and are able to deliver value in a timely manner. he focus on value in terms of either efficiency, cost reduction or revenue generation is crucial as the AI initiatives are not only a technological experiment but are also a business initiative which is able to deliver value to the organisation. The objective of this stage will be the process of strategically placing the engineers in a closed environment where the idea generation process will lead them throughout the sessions with the key focus on innovation and value delivery which helps the company in not only short term but also long term manner

- **Key Activities:** Environmental scanning, idea collection, initial value estimation.

  The key activities during the opportunity scouting stage will be focused on environmental scanning, idea collection and initial value estimation. The need to create a structured approach to idea collection is crucial as it helps engineers think in a direction where the ideas are aligned with the overall business strategy and goals. After the initial idea collection, the ideas will be evaluated based on their potential impact and effort required to implement them. The ideas will be evaluated based on an effort vs impact matrix, where the ideas with the highest impact and lowest effort will be prioritized for further evaluation. By having outlined the set of key activities the teams and adoption manager will be able to approach this stage with a greater easy and ensuring the people that the framework they are following is structure and will prevent chaos and confusion.

- **Roles:** Sponsor, Adoption Manager, Teams.

  The key roles involved in the opportunity scouting stage will be the Sponsor, Adoption Manager and the Teams. In this stage the sponsor and the adoption manager will be responsible for leading the idea generation process and ensuring that the ideas are aligned the company's business strategy and goals. The teams and especially the engineers will be responsible for proposing ideas and providing feedback on the ideas which are being proposed. The engineers are the pinpoint of this stage as all the ideas are coming from them and they are the ones who are able to identify the painpoints and bottlenecks which are hindering their productivity, the role of Sponsor and Adoption Manager is to only efficiently channel the ideas and ensure that they are aligned with the overall business strategy and goals.

- **Decision Points:** Is the use case strategically relevant and feasible?

  The decision point whether to proceed, iterate or abandon the opportunity scouting stage will be based on the question "Is this use case strategically relevant and feasible?". The question shall be asked for every single one of the potentially identified and outlined usecases as it determines winners from losers. This question however dos not focus on strategy nor business alignment, it is rather a feasibility check to the round of engineers, whether this usecase occurs regularly or it does not need AI as a solution.

- **Output:** Shortlist of candidate use cases.

  The expected output of this stage will be a set of potential use cases which are not analysed but rather a good starting point for the next stage of the adoption process. The shortlisted use cases are not the final use cases as they were not properly analysed, nor alligned with the overall business strategy and goals and were not risk assessed, but shall rather serve as a state of mind of the engineers dealing with pain points and bottlenecks in their day to day work.

5.5.2 Step 2: Strategic Alignment Review

The process of strategic alignment review is the second stage of the SAIL framework in which the focus comes on the list of use cases drafted by the engineers. This provess envolves the governance board and the sponsor who are able to evaluate the use cases based on their strategic fit and alignment with the overall business strategy and goals. The objective in this case will be the process of ensuring that the use cases are aligned and are relevant for the business. Throughout this stage each of the use cases will be evaluated by the responsible teams and the initial analysis will be done. This analysis will be done by both teams and the sponsor together with the governance board. This analysis will be done in a dual format due to the need to find realistic expections.

The initial analysis done shall follow this structure:

What is your idea? **The Problem we are solving:** Describe the specific problem or pain point this project addresses.

The Solution: Explain your proposed solution and how it works.

Who will use it? Identify the target users and stakeholders.

Why is this valuable? **Business Benefits:** How does this help the business? What value does it create?

Time Savings: How much time will this save? Per day/week/month?

Daily Work Impact: How will this make people's daily work better/easier/faster?

How will you build it? **Implementation Plan:**

What do you need? What tools, systems, or resources are required?

How long will it take? Rough estimate for development and implementation.

What could go wrong? Main risks or challenges you're worried about.

How will you know it's successful? **Success Measures:** List specific, measurable criteria that will indicate success.

Current Status & Next Steps **Where are we heading?** Describe current status and immediate next steps.

By being able to answer this set of questions, the teams and the governance board are able to have a dicussion and are able to evaluate the use cases based on their strategic fit and alignment with the overall business strategy and goals. This analysis gives the teams a sense of ownership and responsibility which improves the quality of the use cases and gives a new perspective on the use case to the governance board and the sponsor. The important questions such as “Why is this valuable?” and “How will you know it’s successful?” are crucial for the success of the adoption process as they help the teams and the governance board to focus on value delivery and measurable outcomes. The questions of “Why is this valuable?” and the 3 subquestions are crucial as they help the teams and the governance board to focus on value delivery and measurable outcomes. The key of them is not a precise estimation but rather a guidance for the governance board to understand the impact of the use case and the ability to capitalize from successful execution.

- **Objective:** Ensure selected opportunities align with business strategy.

  The key objective of this stage is to ensure that the use cases which were drafted in the previous stage are aligned with the overall business strategy and goals. This being achieved by a dual analysis of the use cases by both the teams and the governance board. The objective of this stage will be to ensure that the drafted use cases are prioritized by the governance board and the sponsor based on their strategic fit will be considered when the teams are walking down the line of the SAIL framework.

- **Key Activities:** Stakeholder review, alignment workshops.

  The key activities throughout the strategic alignment review stage will be stakeholder review and alignment workshops, which represent the indepth use cases analysis. The activity of analysis the use case picked by the team creates ownership which is desired by engineers as the feeling of belonging and responsibility serves as driving force for the success of the adoption process. The alignment workshops will be led by the sponsor and the governance board, will serve as a discusson with review, essentially leading to cherry picking the set of use cases which are able to either proceed to the next stage, need to be iterated on or need to be abandoned.

- **Roles:** Governance Board, Sponsor, Teams.

  The roles in the strategic alignment review stage are the Governance Board, Sponsor and the Teams. As already mentioned previously the Governance Board and the Sponsor are the ones working together to creating one side of the analysis coming from the perspective of the business and strategy while the teams are the ones providing the technical perspective and the feasibility of the use case. The dual analysis concept comes to the closing stages once the final discussion is held and the decision of whether to proceed, iterate or abandon the use case is made. The ability to yield both perspectives is crucial as it helps the teams and the governance board to have a holistic view of the use case and its potential impact on the organization.

- **Decision Points:** Does the initiative align with corporate strategy and transformation goals?

The decision point whether to proceed, iterate or abandon the strategic alignment review stage will be based on the question “Does this initiative align with corporate strategy and transformation goals?”. The question needs to be asked with the mind of not only the business value and strategy but also the technical feasibility and the actual impact which the engineers estimated. After the analysis is done, the concrete numbers of efficiency gains, time savings and daily work impact will be used to evaluate the use case and the impact leaving on the organisation and its internal processes

- **Output:** Approved use cases for assessment.

  The output of this stage will be a set of approved use cases which are able to proceed to the next stage of the adoption process. This will be a set of use cases which are critical for the business, create value, improve either efficiency or reduce costs and are aligned with the overall business strategy and goals. This approved set of use cases will be the backbone of the adoption process and will be the use cases which will be evaluated in the upcoming stage of feasibility and risk assessment. By having a great understanding of the usecases and a deep down analysis done by both business side and technical side the probability of success is greatly increased and the use cases are able to be measured by concrete goals.

5.5.3 Step 3: Feasibility & Risk Assessment

The next stage in the framework is the stage of feasibility and risk assessment, where the focus comes on the technical feasibility and thorough risk assessment process. This stage is the first stage where the technical perspective is coming into play and is the stage where dreams meet reality and might often turn into dust as the idea is faced with the current IT challanges and obstackles which have to be either resolved or the idea will have to be disolved and lesson will have to be learned that similiar scope or expectation might not be feasible unless the IT infrastructure changes. This could either be (eg. license issues, data availability, technical complexity, migration dependecies). From the perspective of the risk assessment, the focus will be on identifying and mitigating potential risks associated with the use case. The risk assessment will be done in a structured manner, where the risks will be identified, evaluated and based on the severity and potential obstacles resolved appropriately. Risk assessment as a concept will have to be done by a proper Risk Officer as they are able to provide an in-depth knowledge on the domain and have the highest chance of catching the errors early on and preventing catastrofic failures down the line. The risk assessment has a stigma around it for the engineers which see it as a bureaucratic process declining innovation and creativity, but in reality the risk assessment is a crucial process which helps the teams and the organisation to identify potential risks and mitigate them before they become a problem. By having a serious risk assessment process, the teams and the organisation are able to commit to work on use cases which are “Risk Assessed” and are able to work on projects which are not likely to be declined down the line due the risks which were not identified early on.

- **Objective:** Assess technical, organizational, and risk factors. The objective of this stage is to assess the technical, organizational and risk factors associated with the use case. The focus of this stage is the identification of potential bottlenecks and risks which are associated with the use case and the ability to mitigate them before they become a problem. From the technical perspective this stage is not only about declining due to lack of infrastructure but also about pointing which resources can be levaraged in order to fullfil the use case.

- **Key Activities:** Technical feasibility study, risk analysis, resource check.

The key activity during this stage will be the technical feasibility study, risk analysis and resource check. From the perspective of technical feasibility study the usage of the framework such as (e.g., Technology Readiness Level (TRL) assessment, Data Maturity Model or AI Capability Maturity Model) could be used to evaluate the technical feasibility of the use case. The risk analysis is rather more tricky as each company is different and the field they are operating in requires different level of risk assessment. The common risk analysis frameworks such as (e.g., Failure Mode and Effects Analysis (FMEA), SWOT Analysis, or Risk Matrix) could be used to evaluate the risks associated. By doing so the key risks are able to be identified and further actions shall be taken in order to alleviate them

- **Roles:** Risk Officer, IT Department, Adoption Manager

The roles of this stage will be the Risk Officer, IT Department and the Adoption Manager. The main driver of the feasibility will be the combination of the IT Department and the Adoption Manager as they will work together in order to evaluate whether the use case is technically feasible and whether the resources are available. The teams are not involved in the process as the Adoption Manager is representing them and ensuring that their perspective is taken into account. The perspective of the IT Department is advocated to ensure that the strategic resources are available and that the use case is technically feasible. The Risk Officer is the one who is able to provide the risk assessment and ensure that the risks are identified and mitigated before they become a problem as they are the expert on the topic and their opinion is the most relevant on the topic.

- **Decision Points:** Are risks manageable? Is the use case feasible?

In this stage there are in fact 2 questions which need to be answered and they both must be answered with a “Yes” in order to proceed. The questions being: “Are the risks manageable?” and “Is the use case feasible?”. The first question focusing whether the impact of the risk and the followed mitigation actions are worth the effort and the resources which are needed to be allocated. The second question focusing on the technical feasibility and whether the actual use case and its implementation is feasible in the current state of the IT infrastructure and its disposable resources. The need for 2 questions is crucial as this decision points acts on 2 fronts, the technical feasibility one and the risk assessment one. Both of them are very crucial and none of them shall be neglected.

- **Output:** Go/no-go decision for pilot. The output of this stage closes the analytical and preparatory part of the SAIL framework where we have a justified idea from every single point, but the stage 3 is focused only on delivering on the questions of if the company’s digital and physical assets are ready for a new use case and if the risks associated with its implemetation are a threat to the company. The output of this stage will be a go/no-go decision for the pilot stage. Once the decision to proceed is made, the use case is able to proceed to the next stage of the adoption process, being the pilot design stage. If the outcome of this stage will be to iterate the process will not be light as once the reason to iterate is a risk based one, the idea might be in jeopardy and the ability to iterate will be limited. If the outcome of this stage is to abandon the use case, the use case will be documented and lessons learned will be gathered in order to improve the process in the future.

5.5.4 Step 4: Pilot Design

The pilot design stage is the stage in which the analysis is being transformed into action points and stories. In order to transform an use case into pilot the appropriate steps needs to be defined. The steps of each individual pilot in terms of its implementation plan are different and that is for the Adoption Manager and the Pilot Team to define. The pilot design stage is the stage where the use case is being broken down into smaller chunks and is being transformed into actionable items. The pilot design stage is crucial as it sets the foundation for the pilot execution stage and ensures that the pilot is properly scoped, resourced and governed. The steps from initial setup of repositories to architecture of the custom tool (if needed) is created and the tickets are spawned and assigned to respective team memebers. This is also the the stage in which the team starts to build confidence as the use case matures and starts to be something which the teams are able to see outlined infront of them like an actual project. The pilot design stage is the stage where the teams are able to see the light at the end of the tunnel and are able to see the potential of the use case. While they are still not implementing and solving the use cases proper planning and roadmap creation prepares the team for the oncoming future

- **Objective:** Plan the pilot implementation. The objetive of this step is the creation of a roadmap filled with milestones and actionable itmes which serve as tickets for the engineers to implement down the road. The goal here is to outline precise steps and direction for the team to take in order to convert the use case to a pilot project. The heavy lifting of this stage is done by the Adoption Manager who is responsible for the roadmap creation and the pilot team who is responsible for the technical input in regards to the implementation plan. The goal of this stage is to create a clear and concise plan which is able to be followed by the team and is able to deliver value in a timely manner.

- **Key Activities:** Roadmap creation, governance setup, refinement of success criteria.

The key activities which are done throughout this stage is the roadmap creation, governance setup and refinement of success criteria. The main activity is the roadmap creation which serves as the backbone of the project and its importance does not only lie in the team having a clear direction but also in the ability to communicate the plan to the stakeholders and the governance board. This actionable backlog gives the management trust and ability to track the work of the engineers on a cross-team basis. Very important activity done during this stage is the refinement of the roadmap with engineers. Once a proper discussion on the roadmap starts the engineers are able to share their feedback not only on the strategy but also the complexity, thus helping on the time and effort estimation.

- **Roles:** Adoption Manager, Pilot Team. The involved stakeholders in this stage are the Adoption Manager and the Pilot Team. The Adoption Manager is the one who is responsible for the creation of the roadmap and the pilot team is the one who is responsible for providing the appropriate technical input and ensuring that the roadmap is feasible and is able to be implemented in a timely manner. The Adoption Manager leads this stage from their front and works closely as a team with the Pilot Team to enhance the outcome.

- **Decision Points:** "Is the pilot implementation plan feasible and properly scoped?"

This stage focuses on creation of the pilot plan and its feasibility and therefore the most crucial question which needs to be answered is "Is the pilot implementation plan feasible and properly scoped?". The question was chosen in order to emphasize the feasibility of the use cases and now the actual pilot design. The scope is another variable which needs to be addressed and critically evaluated as the scope creep is a common problem in the pilot execution stage and by having a clear scope and boundaries the team is able to focus on delivering value and not on expanding the scope. Working on tasks and features outside of the scope hinders the value-delivery time even if the features are nice to have.

- **Output:** Pilot implementation plan.

The clear output of this Pilot Design stage is a well defined pilot implementation plan which is able to be followed by the team and a clear scope which is able to be communicated to the stakeholders and the governance board. The plan for the team creates a vision for the team and ensures the team that there is a clear path outlined in front of them and on the end of the path is a successful pilot which is able to solve their daily painpoints and bottlenecks. The plan is not set in stone and is able to be iterated upon, however by having a good and well refined plan the probability of failure is greatly reduced.

5.5.5 Step 5: Pilot Execution

Pilot Execution is the stage in which the actual implementation of the pilot takes place. This stage involves gradual ticket pulling from the backlog alligned with the roadmap in order to build the pilot solution. The pilot execution stage is the stage where the teams are building the pilot ticket after ticket while the Adoption Manager is responsible for ensuring that the pilot is properly governed and that the success criteria are being met. The pilot execution stage can be structured according to the teams usual project management structure such as (e.g., Agile, Scrum, Kanban) in order to ensure that the team is used to the work dynamic and do not have to go through a change management process. This stage ends once the first MVP is delivered and it can be tested by users in order to give feedback and validate the solution. This however is done in a controlled environment and the pilot team works in a cyclic manner in order to ensure that the solution is properly tested and validated before it is rolled out to a larger audience and the feedback is always considered.

- **Objective:** Build and test the AI solution in a controlled environment.

The objective of this stage is to build and test the pilot project and verify its functionality with a first set of users. The focus of this stage is to deliver a working solution which is able to be analysed and an impact of it can be measured. This impact or otherwise known as success criteria is defined in the analysis stage and taken into consideration during the pilot execution stage. Once the pilot does not meet the success criteria the team is able to iterate and improve the solution until the success criteria are met.

- **Key Activities:** Development, testing, user feedback collection.

The leading activities performed during this stage are development, testing and user feedback collection. This stage starts with the development of the pilot and only once a working solution is implemented the first batch of users shall provide feedback and observances in the shortest amount of time in order for the team to refine their solution. The thorough testing and feedback collection helps mitigate initial problems and ensures that the solution is impactful and user-friendly.

- **Roles:** Pilot Team, End Users.

During this stage there are 2 main roles involved, being the Pilot Team and the End Users. The Pilot team, being the one responsible for the actual implementation of the pilot and the end users, being the ones who are able to provide feedback and validate the solution. The end users are crucial for the success as their feedback helps improve the product as their opinions do not have internal bias and are new to the solution.

- **Decision Points:** Does the pilot meet adoption and performance criteria?

The question to be asked whether this stage is able to proceed or not is "DoesDoes the pilot meet adoption and performance criteria?". At the end of the pilot execution stage the team is able to evaluate whether the success criteria which were defined in the analysis stage are met or not. If the success criteria are met the team is able to proceed to the next stage of evaluation and scaling decision. The benefit of defining the criteria prior to the exeution stage is that emotions can be isolated as the decision is purely based on the merit.

- **Output:** Pilot results and lessons learned.

The output of this stage will be the pilot results and lessons learned. This means that there will be a working MVP of the use case outlined at the beginning which serves as valude delivery instrument having a potential to directly influence the enterprise and create feasible impact. The great benefit of having a structured approach is that throughout each of the decision on the roadmap is that there is a set of lessons learned on the way which can be used in the next iteration of the framework.

5.5.6 Step 6: Evaluation & Scaling Decision

The next stage in the SAIL framework is the stage of evaluation and scaling decision, the focus is now suddenly placed on the evaluation of the pilot results and the decision whether to continue the initiatives implemented in the previous stage of implementation. This decision shall not be taken lightly as this stage is a make it or break it. The decision to scale the pilot to a larger audience or to abandon the initiative is crucial as it determines whether more resources such as finances and time have to be invested into scaling and infrastructure purchasing. This decision shall be based on the evaluation of the pilot results coupled with the analysis done in the step 2 and 3. The decision to scale is solely focused on ROI and potential usability of the pilot solution. The evaluation process shall be done in a structured manner, by the governance board and the sponsor, as their opinion on the topic comes from the high level overview and are the most relevant stakeholders to make the decision.

- **Objective:** Evaluate pilot outcomes and decide on scaling. The objective of this stage will be to come to a decision. The decisions in this stage which we have are 4. Give a green light to scale and deploy, to iterate and improve the initial pilot to optimize it for better rollout, to abandon the initiative as it does not create value or to pivot and change the direction of the initiative and to leave it as it is. The last decision is something the engineers do not like as it is a sign of failure, but in reality it is a sign of maturity and understanding that not all ideas are good and not all ideas are feasible to be scaled. If the pilot stays in the pilot stage and is able to already deliver value and its scaling is not feasible, the decision to leave it as it is might be the best one.

- **Key Activities:** Analyze results, stakeholder review, cost-benefit analysis. The main activities done during this stage are the analysis of the results, stakeholder review and cost-benefit analysis. By having a great overview on the scaling and deployment costs the governance board and the sponsor are able to make an informed decision whether to scale or not. The ability to differentiate between the cost of scaling and the benefit of scaling is crucial as by doing so the calculation of the ROI is possible and data driven strategic decision can be done with high confidence.

- **Roles:** Governance Board, Sponsor, Adoption Manager. Responsible entities for this stage are the Governance Board, Sponsor and the Adoption Manager. Cost analysis and benefit analysis is done by the Adoption Manager while Sponsor is left to make the final decision together with the Governance Board. Their theoretical knowledge and high level overview comes very handy in this stage as they are able to see the bigger picture which often overhauls the technical perspective.

- **Decision Points:** Is scaling justified based on ROI and impact? The decision point whether to scale or not is based on the question "Is scaling justified based on ROI and impact?". This question directly underpins the core priniciple of this entire stage. Whether to scale or not is a very difficult decision as sometimes the impact is great but the cost of scaling is too high and sometimes the cost of scaling is low but the impact is negligible. The second alternative of simply leaving the pilot as it is and not scaling it is often overlooked, however it is a valid decision as sometimes the best one.

- **Output:** Scale/no-scale decision. Simplest summation of this stage is the output, being the scale/no-scale decision. This outcome can be the decision to scale or not to scale. This decision either moves us down the line of the SAIL framework or it stops the process and the use case needs to be documented and lessons learned need to be gathered, but the pilot will be either depricated or will stay as it is without further scaling. By going to this dead end the company goes back to the drawing board and starts the process from the beginning, however this time with more knowledge and experience, which helps the company to avoid going down the same path and making the same mistakes.

5.5.7 Step 7: Scale Deployment

One of the most exiting and impactful stages for the entire company is the stage of scale deployment. This stage transform internal pilots into full AI solutions which are able to be used by the entire organization. The scale deployment stage is the stage where the focus comes on rolling out the AI solution to a larger audience and collecting immediate feedback. In order to ensure successful scaling the AI solutions needs to be thoroughly tested and validated in the pilot stage, as the first impression is the most important one. People tend to base their trusts on the first impression and if the solution is sub-par, not sufficient and users do not see immediate value, the adoption will be low and the solution will be abandoned and not supported by the end users. The scale deployment stage is both technical but also organizational as the change management process is crucial for the success of the scaling. The creation of awareness is extremely important as it creates a buzz around the solution and helps the end users to understand the value of the solution.

- **Objective:** Roll out the solution organization-wide. Transformation from pilot to AI solution. This action is the sole objective of this stage. Transforming the pilot into wide adopted AI solution comes with a lot of challenges, which are overcome by proper planning and execution. The objective is to have a smooth rollout and infrastructural resources to support the solution. Successful scaling is not only about the technical implementation but also about the user stories which the solution is able to solve.

- **Key Activities:** Scaling, infrastructure setup, change management.

Key activities done during this stage are scaling, infrastructure setup and change management. The most important activity for successful scaling is the technological readiness. The infrastructure needs to be able to handle concurrent users while not experiencing downtime or latency. The infrastructure needs to be scalable and flexible in order to adapt to the changing needs of the organization. This simple activity can be solved by using a single variable - money. The more money is invested into the infrastructure the more reliable and scalable it becomes.

- **Roles:** Adoption Manager, IT, Business Units.

The roles involved in this stage are the Adoption Manager, IT and Business Units. The IT being responsible for technical scalability and infrastructural readiness, the Adoption Manager being responsible for the change management process and the Business Units being responsible for the user adoption and feedback collection, while working closely with the Adoption Manager. Their close collaboration is crucial as it will have impact on the user adoption as they will be promoting the solution and creating awareness around it. The Business Units are the ones who are able to provide direct feedback on the solution due to the fact that in each business unit there are different painpoints and use cases which the solution might solve.

- **Decision Points:** Is the organization ready for full-scale deployment? The decision point whether to proceed with the scaling or not is based on the question "Is the organization ready for full-scale deployment?". This question is crucial as it determines whether the organization is ready for the change and whether the end users are ready to adopt the solution. The readiness of the organization is determined by the change management process and the user adoption. If the organization is not ready for the change, the scaling will fail and the solution will be abandoned. The readiness of the organization is determined by the change management process and the user adoption. If the organization is not ready for the change, the scaling will fail and the solution will be abandoned.

- **Output:** Scaled AI solution. The genaral outcome of this stage is a scaled AI solution which delivers sustainable value over time. The scaled AI solution is able to solve the painpoints and bottlenecks of the end users and is able to either cut time to delivery, reduce costs or improve efficiency. The scaled AI solution shall be verified and validated by end users and the feedback shall be collected by the Business Units in order to forward the findings to the Adoption Manager who is able to iterate and improve the solution over time with the engineering team. The scaled AI solution is the backbone of the AI adoption process and is the foundation for the next stage of AI native optimization.

5.5.8 Step 8: AI Native Optimization

The final stage in the SAIL framework is the stage of AI native optimization. The last stage of this lifecycle is the stage where the focus comes on continuous improvement and optimization of the AI solution happens. During this stage the AI solution is already implemented and delivering value, however the probability of it being perfect and solving all the painpoints is very low. The AI native optimization stage is a continous process lasting over time, where the focus comes on continuous improvement and optimization of the AI solution. This stage however is problematic due to the fact on what to decide as "cut off date" for the optimization. The optimization process is a never ending story as there is always room for improvement and optimization. What shall not be forgotten is the rule of diminishing returns, where the more time and resources are invested into the optimization the less value is delivered. The optimization process shall be done in a structured manner and shall be based on the feedback collected from the end users up until the moment the amount of request slows down over time.

- **Objective:** Integrate, optimize, and continuously improve the AI solution.

The final goal of the AI native optimization stage is to integrate, optimize and continuously improve the AI solution. This means that focus is placed on the continuous improvement of the solution and its maintenance. The optimization process of incrementally adding new features and improving the existing ones is beneficial however the focus shall be placed on the maintenance rather already innovative features. This will ensure that the solution stays in the scope and will be used by the end users for the intended purpose. If too many requests are coming in, the solution might drift away from its original purpose and the end users will not be able to use it for the intended purpose. In such a case a new AI use case shall be created and the process starts from the beginning, however for a different use case.

- **Key Activities:** Performance monitoring, process reengineering, feedback loops.

Activities done during this stage are performance monitoring, process reengineering and feedback loops. The majority of work done during this stage is performance monitoring and feedback loops. The AI solution needs to be monitored in order to ensure that it is delivering value and that end users are satisfied not only with the value delivery but also with the user experience. The feedback loops are crucial to ensure high user engagment and adoption. This process shall be done with using interviews, surveys and in solution feedback collection forms. These activities ensure smooth operation and high user satisfaction.

- **Roles:** Adoption Manager, Engineering Team, End Users.

The last stage of the SAIL framework involves the Adoption Manager, Engineering Team and End Users. The key driver in this stage are the End Users as they are the ones who are able to provide direct feedback on the solution and are the ones not only using the tool but are also the ones providing the most valuable comodity in the process, being the feedback. The feedback is collected by the Adoption Manager and is forwarded to the Engineering Team who is responsible for the implementation of the feedback and the continuous improvement of the solution. The Adoption Manager is the one who is responsible for the overall process and ensuring that the feedback is collected and forwarded to the Engineering Team in a timely manner.

- **Decision Points:** Is the solution delivering sustained value? Are further optimizations needed? Whether the AI solution initiative can be marked as successful and the SAIL framework can be closed or not is based on the question “Is the solution delivering sustained value? Are further optimizations needed?”. These questions are crucial for the final decision as the question of sustained value delivery is the main reason why the initiative started and can be evaluated by the end users. The second question of whether further optimizations are needed is crucial as it determines whether the solution is able to be improved or not. This can be decided by calculation of a moving average of the feedback and the number of requests coming in per unit of time. If the number of requests coming in over time is on a downward trend the team can plan to sunset the solution and start working on a new use case.

- **Output:** AI-native business processes.

The final outcome of the entire SAIL framework is the AI-native business processes. This means that the company is able to yield sustainable value over time and is able to continuously improve and optimize the AI solution. The company have officially cleaned a painpoint which was outlined in the initial stage and was successfully solved by leveraging AI. This outcome marks as the ultimate goal of employment AI in the enterprise and is the foundation for future AI initiatives. The company does not only solve the pain point, but also gains extremely valuable experience and knowledge which is able to be leveraged in the future, not only for internal use cases but also for external use cases and product development. The usage and integration of AI in the enterprise is a never ending story and the SAIL framework is the foundation for future AI initiatives and the backbone for the AI adoption process.

5.6 Summary of the Framework

The SAIL framework provides a structured, lifecycle-based approach to AI adoption in software organizations. It integrates decision checkpoints, governance, and feedback loops to ensure scalable, coordinated, and reusable AI initiatives. This comprehensive table summarizes the key aspects of the SAIL framework and extracts them in one place for easier reference.

Step	Objective	Key Activities	Outputs	Governance Checks
1. Opportunity Scouting	Identify and prioritize AI use cases.	Problem discovery, stakeholder engagement, feasibility analysis.	Prioritized use case backlog.	Is the use case strategically relevant?
2. Strategic Alignment Review	Align opportunities with strategy.	Dual analysis by teams and governance board.	Approved use cases.	Does it align with strategy?
3. Feasibility & Risk Assessment	Assess technical and risk factors.	Feasibility study, risk analysis, resource check.	Go/no-go decision.	Are risks manageable and use case feasible?
4. Pilot Design	Define pilot implementation steps.	Roadmap creation, governance setup, success criteria.	Pilot implementation plan.	Is the plan feasible and scoped?
5. Pilot Execution	Build and test AI solution.	Development, testing, user feedback.	Pilot results and lessons learned.	Does pilot meet performance criteria?
6. Evaluation & Scaling Decision	Evaluate pilot and decide on scaling.	Results analysis, stakeholder review, cost-benefit analysis.	Scale/no-scale decision.	Should initiative be scaled?
7. Scale Deployment	Roll out solution organization-wide.	Change management, system integration, MLOps setup.	Deployed AI solution at scale.	Is deployment feasible?
8. AI-Native Optimization	Continuously improve AI solution.	Performance monitoring, process reengineering, feedback loops.	AI-native business processes.	Is continuous improvement sustainable?

Table 5.2: SAIL Adoption Stages

Table 5.2 is clearly showcasing all of the eight steps which are required to successfully complete the SAIL framework. This table is consisting out 5 scopes. The first being of the name of the step in the SAIL framework, followed by the objective which the step is aiming to do. The next column is focusing on the key activities which are required to be done in order to successfully complete the step. The second to last column focuses on the outputs which should be expected by the end of the step and the last column is focusing on the internal governance check.

By having a clearly outlined table, it is easier to reference, analyze and view how the entire SAIL framework operates and what are the key components of each individual step. This structured pathway is ensuring that AI initiatives are going to be strategically aligned, technically validated and verified in terms of the organizational governance check.

Chapter 6

USE CASE IMPLEMENTATION & VALIDATION

6.1 Introduction

This chapter will present an empirical application of the SAIL framework in realistic software engineering contexts. By walking through how SAIL is able to be applied to a diverse set of 4 use cases, the practical utility and adaptability of the framework will be demonstrated. Each use case will illustrate different entry points, challenges, and outcomes, showcasing SAIL's flexibility across various organizational scenarios.

6.2 Selection of Use Cases

The following use cases were carefully selected in order to represent the diversity of AI adoption scenarios in software organizations. They cover a range of functions from developer productivity to internal knowledge management, demonstrating SAIL's applicability across different domains and organizational needs. The use cases also vary in complexity, data requirements, and stakeholder involvement, providing a comprehensive validation of the framework.

Selected use cases:

Use Case	Description
AI-based Code Documentation Generation	Automated the process of generating and maintaining code documentation using AI models to analyze codebases and produce up-to-date documentation.

Use Case	Description
AI-driven refactoring	Introduced a new way how engineers tackle tasks which are related to optimising refactoring process by AI
Internal LLM Chatbot for Knowledge Sharing	Deploying an internal large language model (LLM) chatbot to facilitate knowledge sharing and quick information retrieval among employees.
Pull Request Review Automation	Automated the review process of pull requests using AI to identify potential issues, suggest improvements, and ensure code quality standards are met.

6.3 Application of SAIL to Use Cases

Each use case is mapped through the SAIL lifecycle, highlighting entry points, progression through stages, key decision points, governance involvement, and outputs. The following subsections provide concise walkthroughs for each use case.

6.3.1 AI-based Code Documentation Generation

Entry Point: The first use case in which we are going to explore the implementation process in the AI testing framework is AI-based code documentation generation.

This use case will be focused on the most tedious and boring task for any software engineer: the creation and ongoing maintenance of code documentation. The process of writing code documentation is very often overlooked while being too focused on the actual code writing. Once any engineer gets the motivation to write the documentation, the documentation is often not at the quality and understanding level required for other engineers to understand the code. This insufficient documentation is then leading to a chain of problems such as misunderstanding of the code, lowered efficiency and sometimes even bugs in the code due to the fact that the engineers do not understand the code properly. The issue is clearly present in most of the software organizations and by having this problem solved the efficiency of the engineers will increase drastically.

In order to verify the SAIL framework we will explore how this use case was implemented in a realistic software engineering context.

Opportunity Scouting:

This use case was identified by software engineers at the time and was initiated by the frustration of doing the task and not having the documentation properly up to date. The actual process of writing the documentation was not the hardest problem which the engineers have to tackle but the process of constantly changing it once the code changes.

Strategic Alignment:

The process of strategic alignment for this specific use case was done in the context of the overall strategic goals of the company. The first step was the consultation with the management team in order to understand the strategic goals of the company and how this use case is able to align with them. The internal company goals were focused on the improvement of the company's efficiency and thus this use case was able to get the green light. The second step that needed to be done was the selection procedure for the appropriate engineers to implement this use case. A set of 3 engineers were selected to tackle it. For the simplicity of the entire process these engineers were from one single team and thus dealing with their line manager became easier and more streamlined. By solving both sides of the equation, the strategy as a company and strategy as a process this use case is now able to proceed to the next stage.

In terms of the expected metrics the team have outlined the following ones:

- Reduce time spent on documentation creation by 90%
- Improve documentation quality by 20%
- Increase developer satisfaction with documentation process by 50%

Each of these metrics is directly tied to the pain point which the engineers are facing, and thus by solving them, the engineers will be able to focus on more value-adding activities.

The metrics will be measured at the end of the SAIL framework using the methods outlined in the table below.

Metric	**Measurement Method**
Reduce time spent on documentation creation by 90%	Developer time tracking survey.
Improve documentation quality by 20%	Documentation quality survey
Increase developer satisfaction with documentation process by 50%	Developer experience survey

Technical Feasibility & Data Assessment:

The first initial step done by the engineers was the Technical Feasibility and Data Assessment check. The engineers have managed to identify that the best approach for this use case is to leverage an existing LLM model and fine-tune it on the internal codebase of the company. The process which they have chosen was utilising the Github Copilot technology which was already preapproved by the safety and security team of the company. The engineers have managed to identify that the codebase of the company is sufficient in order to fine-tune the model and thus the technical feasibility check was passed.

In terms of the data assessment, the engineers have selected the most appropriate code base to start on, this being the Python scripts which were among the most edited and worked upon and thus would have the highest return on effort once our pilot becomes the standard.

In terms of creating the strategy for the deployment, the team did research on how to distribute this use case to the widest number of engineers at once, and this is done via a VSCode extension. The skill to create it was not currently in the team and this was identified as a gap which needed to be filled before proceeding to the next stage.

Pilot Design:

The pilot stage for this use case involved creation of the smallest deliverable product which other engineers are able to use and gain immediate value. This MVP was a VSCode extension which is able to generate code documentation for python scripts. The process of creating this MVP had many challenges such as the lack of experience in creating VSCode extensions which had to be tackled by the engineers as an immediate learning curve and the first process of trial and error.

The architecture of this solution was designed as an integration of GitHub Copilot in an agentic model with an underlying Claude Sonnet 4.5 model. This combination will be then enhanced with the special usage of markdown files whose purpose is to be guiding the Agentic model to the best efficiency possible.

The sole purpose of the markdown files is to specify custom rules in terms of documentation, so that the model is then able to retrieve these files and know what the current standard of the internal rules and configurations is.

This approach gives the model an actual leverage and a touch of customization over just simply using the out of the box Github Copilot functionality.

By the end of the pilot design stage, a clear roadmap was created serving as the backbone of the entire use case. This roadmap consisted of all the steps needed to be done from the onboarding until the final deployment, release, and rollout of this use case. This roadmap does not only serve the engineers to track the time inside of it, but also shows the management the complexity of the use case and gives them the ability to plan this use case over a period of time in a more structured way.

Once all of this was created, we were then able to move into the next stage.

Pilot Execution:

The pilot execution stage for this use case started with the process of the actual onboarding and learning about how to successfully create and deploy a VSCode extension. Online resources were gathered mostly from the actual Microsoft services to understand how engineers are able to achieve this goal. We opted for the route of onboarding only one engineer instead of three because in the meantime the other two engineers had already started writing specific markdown files which would then guide the AI model in the agentic mode.

By having this distributed approach, we are able to not be slowed down by the learning curve of creating this VSCode extension and continue working on the use case.

The pilot execution stage lasted for around 3 weeks and by the end of it a working MVP was created which was able to generate code documentation for python scripts. The MVP was then tested by a small group of engineers and feedback was collected in order to improve the solution.

After the successful and improved MVP, the team was then able to move into the next stage of evaluation and scaling decision.

Evaluation & Scaling Decision:

The process of evaluation and the scaling decision for this use case involved the actual analysis of the results and the interconnection between various entities such as the sponsor, the management team, and the security team.

The analysis looked at our success criteria which were outlined in the strategic alignment stage and evaluated whether they were met or not.

The numbers which we gathered from the documentation creation process involved metrics of reduction in the time spent on creation of documentation, improvement in documentation quality, and the overall satisfaction.

In the pilot stage, we managed to get the time spent on creation of the documentation reduced by 70%. The documentation quality was improved by 15% and the overall satisfaction of the engineers with the documentation process was increased by 40%.

Based on these metrics, we can clearly see that the MVP has a very strong potential and is already able to yield very promising results. As these results are already on the track to be successful, we are going to proceed with the creation of the final product and scaling it.

If the metrics were not on track in terms of the promised and expected numbers, we would have to either reevaluate whether the targets which we set were too high or whether the observed metrics are objectively too low.

Scale Deployment: The next step in the SAIL framework is the stage of scale deployment. This stage involves the actual rollout of the AI solution to a larger audience and collecting immediate feedback. During the scale deployment phase, the engineers successfully managed to tweak the markdown files to be more appropriate for the code documentation while still collecting very valuable feedback from the engineers. This has created a cycle of improvement, leading into a better and more rigid solution.

The process of deployment has been successfully achieved by establishing a private VS Code marketplace which is self-hosted and where every single AI use case developed internally is able to be deployed. This marketplace isolates the contact from outside world and therefore ensures security and safety of the internal tools.

As every single engineer has the ability to simply download this VS Code extension inside their VS Code editor, the friction between installation and usage is completely minimized, thus improving the process of change management and reducing the resistance to change.

AI Native Optimization: The final stage in the SAIL framework is the stage of AI native optimization. This stage involves the continuous improvement and optimization of the AI solution.

As the previous stage has been more focused on the distribution, the next stage needs to be focused on the actual improvement of the solution. Inside the company, we positioned our team as a center point of AI and allowed engineers to comment on the repository of this use case. By collecting this valuable feedback, we were able to improve the solution and tweak it according to the developers' preferences. By having a wider audience of users using the solution, we were able to yield very promising results and succeed in the process of continuous improvement.

After successfully implementing all the suggestions and comments, we were able to close the initiative as successful and move on to the next use case with the following results highlighted in the table below.

Metric	Result Achieved
Reduce time spent on documentation creation by 90%	Achieved a 90% reduction in time spent on documentation creation.
Improve documentation quality by 20%	Improved documentation quality by 22% based on developer surveys.
Increase developer satisfaction with documentation process by 50%	Increased developer satisfaction by 55% according to feedback forms.

The process of optimization does not, however, end at simply receiving the results of the survey. The process of continuous improvement is a never-ending story unless the use case diminishes in terms of the usage by the engineers. For this use case, a set amount of questions and remarks were still added even after several months of usage. This means that the process of continuous improvement is still ongoing, however, we are getting a value from a product which is already implemented and requires close to minimal support and maintenance. This example clearly demonstrates us that it is possible to create an AI product which is delivering sustainable value. In order for the highest amount of engineers to properly use this tool, a set of workshops and training sessions were created in order to onboard the engineers and show them the value of this tool. By having this change management process in place, the adoption of the tool was significantly increased and thus the overall impact of the use case was maximized.

6.3.2 AI-driven Refactoring

Entry Point:

The use case to be explored and implemented is the process of developing an AI-native agent pilot which will be responsible for refactoring any part of the existing code base. This use case is going to be also based on GitHub Copilot with Claude Sonnet 4.5 and a set of markdown files guiding the agentic model on how to refactor the code.

The process of refactoring is extremely important and needs to have a high level of quality and a concrete set of rules as each engineer views what is the best for the company and the code differently. This brings many ideas to the table, but does not determine what is the best approach for refactoring. Certain engineers prefer verbosity for the sake of readability and others prefer conciseness for the sake of efficiency. This conflict of interest does not have a concrete result and a concrete base. For a concrete decision point to be done, a set of rules needs to be evaluated, created, and enforced. This is exactly where the agentic AI model comes into play.

By having a set of markdown files and guiding the AI agentic model to enforce a strict refactoring, we are able to eliminate this problem between two engineers disagreeing on what is the best approach for refactoring.

Opportunity Scouting: This opportunity of having an AI-driven refactoring comes from the need to have a very high level of code which is being monitored by various assessments. Assessments such as: AST analysis, Cyclomatic complexity, Code duplication and other code quality metrics are extremely important for the overall health of the codebase and thus having a tool which is able to automatically refactor the code based on these metrics is extremely valuable.

The opportunity to develop an internal AI-driven tool which is able to refactor any type of code base based on the set of well-maintained markdown files used for configuration brings the company, on the first glance, an immediate return, not only from the perspective of having a clear, well-maintained and readable code, but also having long-term clean code for the new engineers to be able to onboard themselves on it and bring value in a shortened amount of time.

Strategic Alignment: The process of strategic alignment for this use case was in fact very simple, because once the management was aware of the problem that many engineers have many different coding standards, having a unified approach to this problem was extremely valuable.

The main approach when negotiating with the upper management focused on the long-term value of having a clean codebase rather than focusing on the fact that it's done by AI due to the novelty. This approach only enforced our integrity and seriousness of the use case.

The vision of having a unified approach to declutter the current processes and help engineers have a more readable code while still preserving the efficiency of it was extremely valuable and thus the management was on board with this initiative from the very beginning.

In order for maximum efficiency, a team of two people were selected to tackle this use case. Due to the nature of the company and the trust which the AI community of practice had from the upper-level management, conveying the message of selecting just two engineers from the given team was not a problem that needed to be tackled as upper management stood on our side and trusted our judgment.

Technical Feasibility & Data Assessment: From the perspective of technical feasibility and data assessment, we had to decompose this problem. The first step was to identify whether there is already a set of rules which the company is currently refactoring by. The second question was whether the code base was already in a sub-par state and the use case even had the need to be created. This would mean that if the code base was ordered in an amazing and perfect and clean state according to the latest coding standards, the need for this use case would be completely abolished.

After further evaluating the current code base, the conclusion came to it being in a sub-par state and the need for this use case became apparent. In terms of the rules and standards of refactoring throughout the entire company, the only existing rules were the ones enforced by the PEP8 standard developed under Python. This however, shall not be confused with refactoring, PEP 8 is a linting standard and the vision of our AI agent is concrete refactoring.

As this use case will be once again developed under GitHub Copilot with the already existing LLM model Claude Sonnet 4.5, the technical feasibility was already passed as the safety and security team have already preapproved this technology for internal usage.

When it comes to estimating the impact and expectations from this use case, the team outlined the following metrics:

- Reduce code complexity by 25%
- Improve code readability by 30%
- Increase developer satisfaction with refactoring process by 40%

The metrics outlined closely describe what the engineers are struggling with and what the main problem is which they are facing. By solving these problems, the process of refactoring will become extremely streamlined and the engineers can focus on other tasks which are more value-adding. The metrics will be measured at the end of the SAIL framework using the methods outlined in the table below.

Metric	**Measurement Method**
Reduce code complexity by 25%	Junior Engineers Survey
Improve code readability by 30%	Code review surveys and readability scoring tools.
Increase developer satisfaction with refactoring process by 40%	Developer experience survey

Pilot Design:

The Pilot Design stage started with research from the team on the best practices when it comes to refactoring. The latest and greatest standards were gathered, documented, and their input was written in the markdown files. The focus of the team was purely Python, as it's the most dominant and impactful language in the company. By having clearly written rules which are enforced, we are able to ensure that the code quality is constantly improved and the engineers are able to focus on more value-adding activities.

After thorough research, the team decided to proceed to implement their own GitHub Copilot agent based on Claude Sonnet 4.5 as the main driver.

The designed process of how to use this tool was created and came to the following steps:

1. Engineer selects the code segment to be refactored. 2. Engineer activates the AI refactoring tool via VSCode extension. 3. The tool retrieves the relevant markdown configuration files. 4. The AI agent analyzes the selected code segment based on the rules specified in the markdown files 5. The AI agent generates refactored code suggestions. 6. The engineer reviews the suggestions and accepts or modifies them.

By having a very straightforward process when it comes to usage of the tool, the friction for adoption is minimized. The vision is to then package this functionality inside the VS Code extension and simply launch it on other engineers' computers. This clearly defined architecture and process standard then led the team to the next stage of pilot execution.

Pilot Execution:

The pilot execution stage for this use case started with a rigorous process of following the roadmap outlined in the step before. From the beginning of the entire roadmap, the goal was to implement an MVP which fulfills the objective of refactoring Python code based on the input and standards from the markdown files. The two engineers tackled this use case over the span of three months and successfully implemented all of the tickets outlined in the roadmap and developed the first version of the MVP. Once the MVP was done, a presentation to showcase the first value creation was done in the company to start the internal process of marketing and awareness. By having the first MVP presented to the wider audience, the team was able to collect valuable feedback from the engineers and improve the solution even further.

After the successful improvement based on the presentation, the MVP was then forwarded to another team which conducted rigorous testing and gave a series of comments that helped the team improve the solution and refine it according to the engineers' preferences.

The most valuable feedback was to have a tiered approach when it comes to the refactoring levels. This means that the engineers wanted to have the ability to select whether they want a light refactoring, medium refactoring, or a heavy refactoring. This feedback was extremely valuable as it allowed the team to improve the solution and make it more flexible to the engineers' preferences.

Another very valuable feedback was to implement team-specific markdown files. This means that each team is able to have their own set of rules and standards when it comes to refactoring. This, on the first glance, might seem counterproductive, but on the other side a solution needed to be found.

The new feedback came from the principle that the whole company cannot sustainably live off a single coding standard due to different customer programs being delivered to different customers, each requiring completely different requirements and standards, and therefore a more modular approach came into the discussion and further implementation.

All the feedback gathered in their data collection stage was then implemented and the team was able to move into the next stage of evaluation and scaling decision.

Evaluation & Scaling Decision:

Once the process of evaluation and scaling decision came to place, the team had already gathered trust of management, the engineers, and had already received company-wide recognition. This has outperformed the existing expectations from the beginning of simply keeping it as an MVP and thus the decision to proceed, reject or iterate was more than clear. The need for this use case was clearly in the company and it immediately solved another pain point which the engineers were facing.

Now the correct question would be to ask whether this is not enough and if any extra effort put into this use case would yield a positive return. On one side, the solution was good enough as it was already delivering value and engineers started using it, but on the other side, the team saw a potential for further improvement and thus the decision to proceed with scaling and deployment was made. This improvement came to place mostly in the aspect of having more languages supported rather than just Python. By having more languages supported the overall impact of the use case would be significantly increased and thus the decision to proceed with scaling and deployment was made.

From the perspective of metrics, the following numbers were achieved during the pilot stage.

Metric	**Result Achieved**
Reduce code complexity by 25%	Junior Engineers Survey indicated a 30% reduction in code complexity.
Improve code readability by 30%	Improved code readability by 40% based on developer surveys.
Increase developer satisfaction with refactoring process by 40%	Increased developer satisfaction by 30% according to feedback forms.

Scale Deployment: As the decision to scale and deploy the solution was made, the team started the process of preparing the solution for the wider audience. While entering into the stage, the solution was overly packaged as a VS Code extension ready to be deployed. The main challenge which the team was facing is the lack of supported languages and therefore the same logic was implemented for further languages being: (Shell Script, C++ and C).

The process of scaling the solution into more languages could unfortunately not be reused from the current status done in Python as new research needed to be conducted. The reason for this is that each language has its own set of standards and rules when it comes to refactoring, and therefore the team needed to go through the same process of research, documentation, and implementation for each language separately.

After successfully implementing the additional languages, the team could then proceed with the deployment of the solution to a larger audience of engineers. This was achieved by once again utilizing the private marketplace hosted inside the company. The engineers could simply download the VS Code extension from the marketplace and use it immediately inside their editors. By simply having this option the friction for adoption was minimized and the team was one step closer to the final stage of AI native optimization.

This however does not paint the whole picture. Only having the use case available to the engineers does not change the behavior and the current mentality of them. Many of the software engineers didn't even know that the marketplace offered such a great tool. This had to be done via rigorous training, workshops, and internal company marketing to convey the message that there exists a tool and this tool shall be utilized in order to gain significant improvements in terms of time, code quality, and a common vision for great code standards.

AI Native Optimization: The final stage of the SAIL framework is the stage of AI native optimization. Throughout the stage, the team gathered many more comments regarding the code refactoring, the latest changes, the standards, architecture, and principles which the engineers would like to see implemented. The larger audience once again allowed for better final outcome. We have utilized the principle of having a public repository, where engineers could simply comment their requests and suggestions which they would like to see and thus giving people the power to control the direction of the use case.

By having this continuous improvement process in place, the team was able to successfully implement the most requested features and suggestions and thus improve the solution even further.

The ongoing iterative optimization process wasn't so active as expected. The external engineers were initially very proactive when it came to suggesting new standards, but once a common ground was achieved only very few people were commenting on changes. This gives us a conclusion that the solution has reached a point of maturity and the engineers are satisfied with the current state of it.

This claim can be backed by the final survey done among the engineers which yielded the following results:

Metric	Result Achieved
Reduce code complexity by 25%	Junior Engineers Survey indicated a 28% reduction in code complexity.
Improve code readability by 30%	Improved code readability by 35% based on developer surveys.
Increase developer satisfaction with refactoring process by 40%	Increased developer satisfaction by 45% according to feedback forms.

The final survey clearly tells us that the solution was implemented in a fashion which was tailored not only to the general engineers but also to the juniors. The juniors are usually the engineers with the least amount of experience with code, and by tailoring also to their understanding, we are able to ensure that the solution is sustainable in the long term as the new hires will also be able to utilize it.

If we only extrapolate the fact that juniors preferred this tool, we might come to a conclusion this tool makes code very inefficient and very verbose. However, this is not the case as another global survey was sent to all engineers and the result of 45% satisfaction increase clearly indicates that the solution is well balanced. If the verbosity would be increased and thus compromising on the efficiency, senior engineers would report this use case in the survey as underperforming and thus giving us a clear indication that the solution is not well balanced.

Therefore, we can strongly conclude that the use case is well balanced between the verbosity so the juniors understand and the efficiency so the seniors are also satisfied.

6.3.3 Pull Request Review Automation

Entry Point:

The following use case of having an automated pull request review system is extremely valuable for any software organization. Every single line of code just added inside of the code base of an organization needs to go through a pull request review. This means that a human is usually reviewing code coded by another human. No matter the size of the organization, this process is extremely time consuming and tedious. Oftentimes, junior engineers are creating pull requests which are sub-par to the quality of the code inside of the code base and senior engineers are usually tasked with reviewing these pull requests. This creates a bottleneck as senior engineers are usually the busiest people inside of the organization and having them review code which is not up to the standards is a waste of their time. By having this process automated by an AI agent, senior engineers are able to save expensive precious time to work on more critical features and functionalities, rather than reviewing code which could be done by an AI agent.

An AI agent which is able to review pull requests based on a set of markdown files guiding it on what are the standards and principles of the company is extremely valuable. This agent will be able to review the code, give comments, suggest improvements and even approve or reject the pull request based on the quality of the code. This will create a significant time saving for senior engineers and thus allowing them to focus on more value adding activities.

The benefit of having an AI agent running on a virtual machine is that it does not only see the static content of the pull request and review it from the perspective of linting and organization of the code, it is also able to execute the code as it has resources dedicated for this purpose.

This gives the AI agent a significant advantage over human reviewers as it is able to test the code, run unit tests and integration tests and thus give a more holistic review of the pull request.

Opportunity Scouting:

The phase of opportunity scouting involves identification of the pain point, this being the pull request review process. This use case was pitched by senior engineers as they are the ones responsible for reviewing the pull requests and keeping the quality of the code base high. Once the company has an AI-driven pull request review automation, it can restructure the entire process of code review and allow senior engineers to focus on more value-adding activities. The time saved from this process is extremely valuable as senior engineers are usually the most expensive resources in the organization and having them review code which could be done by an AI agent is a waste of their time.

Strategic Alignment: The complexity of this use case is greater than the already existing ones. This means that the strategic alignment process needs to be done with multiple entities. This use case is not focused on the alignment of the vision, mission, and the goal of the company; this use case comes to life because it can revolutionize how the company operates. The key strategic entities which the team will have to align with are the IT department, the Vice President of Engineering, and the Safety and Security Department.

As this use case requires more manpower, a team of four engineers was assembled. The team consisted of two senior engineers, one junior engineer, and one regular professional. This gave the team the ability to tackle the complexity of the use case from multiple angles and ensure that the solution is well rounded and robust.

The strategy alignment on this use case was done via a series of meetings with the key stakeholders. The first step was to present the use case to the IT department and get their buy-in. The second step was to present the use case to the Vice President of Engineering and get his buy-in and initial tips while advancing on the roadmap. The third step was to present the use case to the Safety and Security team and verify what type of infrastructure is needed in order to ensure that the solution is compliant with the company's security standards.

All of these steps are very crucial to ensure that the use case is well researched and aware of all the potential roadblocks down the line. If all the strategic points wouldn't be properly aligned, any factor could totally disintegrate the entire idea. This would mean that all the effort put into the use case would be wasted, and thus the strategic alignment is extremely crucial for the success of this use case.

Technical Feasibility & Data Assessment: From the perspective of the technical feasibility and data assessment, the team had to first analyze if the already existing infrastructure is sufficient for an AI agent to be hosted on. The team had to also identify whether it is even possible to create API calls between the current infrastructure and the used services to fulfill the use case of reviewing pull requests with an AI agent. The technical feasibility check was conducted very closely with the IT department.

The final conclusion from the IT department was that all the services and our approach is safe and secure and the only need was to deploy a separate virtual machine acting as a server on which the AI agent will be hosted and run. This virtual machine will be connected via API calls to the existing code repository and thus will be able to review pull requests in close to real time.

The data assessment part involved identification of the existing code base and the quality of it. The team had to ensure that the code base is sufficient in order to train and fine-tune the AI agent to be able to review pull requests effectively. The team identified that the code base is indeed sufficient and thus the technical feasibility and data assessment check was passed.

In order for the team to try to aim for certain benchmarks, the following metrics were established as our success criteria.

- Achieve an automated pull request review
- Reduce the time spent on pull request reviews by 100%
- Increase developer satisfaction with pull request review process by 50%

This time one of the metrics does not have a clear number next to it, but it is rather a yes or no observation. The two other metrics are clearly defined in terms of the optimization. It is very important to point out that the first metric, to achieve an automated pull request review, is a remarkable achievement already on its own and thus the other two metrics are more of a cherry on top. Even though on the first glance the first metric might not be useful for the observer, any type of automated pull request review already brings value in terms of saving time for senior engineers. In order for us to understand whether the use case was successful or not, the team will conduct measurements at the end of the SAIL framework using the following methods.

Metric	Measurement Method
Achieve an automated pull request review	System logs and monitoring
Reduce the time spent on pull request reviews by 100%	No pull request reviews anymore
Increase developer satisfaction with pull request review process by 50%	Developer experience survey

Pilot Design:

The pilot design stage for this use case involved a series of investigations, research, and planning. The first step was to identify that in order to have an AI agent review pull requests, a virtual machine acting as a server is a precondition. This virtual machine will be hosting our AI agent which will be directly communicated via API calls to fetch the latest changes and be able to execute them independently on the code base which it will have in its current workspace.

The backbone of our AI agent will be once again GitHub Copilot with Claude Sonnet 4.5 as the main LLM model. This model has already successfully proven itself in the previous use cases even though its complexity was not up to the level of reviewing pull requests and executing code on human behalf.

The next step in the pilot design stage was to create a set of markdown files guiding the AI agent on what are the coding standards and principles of the company. This is extremely crucial as the AI agent needs to understand what is expected from it in order to review pull requests effectively. The markdown files will be constantly updated and improved based on feedback from senior engineers and changes in coding standards.

The next step in terms of our pilot design stage was the creation of a set of markdown files which are going to be guiding the AI agent and helping it understand the dependencies between files, functions, and logical principles. This is done not because AI is not able to detect patterns and act upon them, but because for every single execution, AI doesn't have to do this tedious process and it can already use the existing knowledge base. The reasoning for this action is that it should significantly reduce the time required for the AI agent to review pull requests and thus improve the overall efficiency of the solution.

The following step will require an establishment between the virtual machine hosting the AI agent and the code repository via API calls. This will allow the AI agent to fetch the latest pull requests and review them in close to real time.

The step which will determine what the success of the pilot design is is the definition of the actions which the AI agent will be able to take. As already mentioned in the entry point, the AI agent will be able to review the code, give comments, suggest improvements, and even approve or reject the pull request based on the quality of the code. This will create a significant time saving for senior engineers and thus allow them to focus on more value-adding activities. This will have to be marked down and will act as the core identity of the AI agent.

In the final step, the pilot will be tested on a sample of pull requests and gather feedback from senior engineers. This feedback will be crucial in order to refine the AI agent and improve its performance. This use case will rely on the feedback loop from senior engineers more than any other use case as we are losing the sense of human review and thus the trust from senior engineers needs to be built up in order for the solution to be successful.

After establishing all the required components and architecture for successful execution of the pilot, aligning the strategy with the key stakeholders, and passing the decision whether the use case is technically feasible or not, the team was ready to proceed with the pilot execution stage.

Pilot Execution:

The pilot execution stage started with an unexpected challenge. This challenge involved restriction of API access and configuration of the firewall requiring tedious process of approval from the IT department and the safety and security department. The reasoning for this restriction was the security concerns regarding the access of an external virtual machine to the internal code repository.

After successful implementation of the agent, the procedure to implement the markdown files acting as a knowledge base for the AI agent began. This process involved thorough research and documentation of the coding standards and principles of the company. The team had to ensure that the markdown files were comprehensive and covered all the necessary aspects of code review.

This, however, did not end up being sufficient as during the initial testing phase, several senior engineers discovered that the AI agent was approving syntactically correct but architecturally flawed and insufficient code. Due to this issue, the team was prompted to do further refinement, deeper analysis, and alignment sessions with the VP of Engineering to guide them towards a successful path. During these sessions, a proper strategy was developed for the AI to understand the architecture, not only on the principle of syntax but also under the principles of design and architecture.

Another flaw which the junior engineer identified was a very critical bug where an agent failed to detect circular dependency across multi-file pull requests. This led to the implementation of a multi-pass review architecture which analyzes the full repository context before approving changes. Having an approach such as this significantly improved the quality of the pull request reviews and thus increasing the trust from senior engineers.

During the execution phase, senior engineers discovered a harsh truth from the principle of change management: the trust. This lack of trust came from the fact that the agent's review comments lacked contextual reasoning explanations and this prompted the team to implement a verbose logging mode where the AI justifies every rejection with specific markdown guideline references. This further justification improved the perceived trust and helped engineers understand the reasoning behind the rejections.

A very positive aspect of this execution phase, was the unexpected win which the team achieved. This win being the instance in which the agent discovered a legacy code pattern violation that human reviewers had been consistently missing for months. This discovery validated the use case to previously skeptical VP of Engineering and secured additional resources for scaling beyond pilot phase.

The additional resources beyond the pilot phase were extremely crucial as they will serve us as a quality gate on whether to proceed with scaling and deployment or to iterate further. This however shall not be the same determining factor as metrics and impact will be guiding us in the next stage of evaluation and scaling decision.

Evaluation & Scaling Decision: As already mentioned, the impact and the trust gained from the VP of Engineering gave not only the team but the engineers already testing this pilot the confidence that the solution is indeed valuable and thus the decision whether to proceed with scaling and deployment or to iterate further was not as complex as in other use cases.

In order to follow the SAIL framework correctly, the team had to evaluate the impact of the pilot and measure the success criteria outlined in the technical feasibility and data assessment stage.

The first metric which was outlined was the achievement of an automated pull request review. As this metric was more of a yes or no observation and the source of truth was the system logs and monitoring, the team can confirm that a fully automated agent reviewing pull requests is indeed achieved.

The second metric involves the reduction of time spent on reviewing and monitoring pull requests by 100%. The vision was that every single pull request opened in the company would be left to the AI agent and that humans would not be in touch with reviewing the code anymore. This, however, has not become the case. In order to set the reality straight, the engineers are still overseeing the pull request review process and thus the time spent on pull request reviews is not reduced by 100%. This metric shall, however, be questioned, whether the reason is the incompetence of the AI agent or the lack of trust from senior engineers. It is very natural that humans do not trust an AI agent straight out of the box. The process of change management requires time, results and proven track record of success in order for humans to trust the AI agent. What, however, has to be mentioned is the significant decrease in terms of the actual time spent on pull request reviews by senior engineers.

In the current state, the AI is able to review the pull request based on the markdown files and give comments and suggestions. The senior engineers are only overseeing the process and giving the final approval in terms of whether the code in the pull request truly fulfills the quality standards and its purpose matches the requirements outlined in the ticket.

Based on the feedback from senior engineers, the time spent on pull request reviews has been reduced by 70%, which is a significant achievement on its own. In this case, the expectation of reducing the time spent on pull request reviews by 100% was not achieved, but it could be argued that the original expectation was too high and the achieved result is still extremely valuable.

The last metric involved the increase of developer satisfaction with our new pull request review process by 50%. This metric was measured by an actual developer survey. The results from the survey indicated an increase in developer satisfaction by 40%. Once again, the final metric did not meet the original expectation and once again it would be appropriate to question whether the original expectation was too high.

If we simply extrapolate the increase in developer satisfaction by 40% and looked at the number from a birds eye view, we can clearly see that this is a significant achievement. What still has to be questioned is how can we improve this?

As in the current stage the use case is still in its early times and is by far not perfect, there is a lot of room for improvement.

The current potential improvements which the team has identified are mostly related to improvement of logging and dynamic tracking of the changes. By having a more dynamic tracking of the changes, the AI agent will be able to understand the context of the pull request better and thus give more accurate reviews.

After the final evaluation of all the metrics and an overall impact of the pilot, the team has come to a conclusion that the use case is indeed valuable and it will be the right decision to proceed with the process of scaling and deployment rather than iterating further to improve the pilot.

Scale Deployment: The process of scaling this use case came down to two main divisions: the technical scaling and the organizational scaling. From the perspective of technical scaling, the team had to simply ensure that the agent will be able to handle the load of the entire organization rather than just a small subset of engineers testing the pilot. This was achieved by scaling vertically the virtual machine hosting the AI agent and thus ensuring that it has enough resources to handle the load.

The larger unknown variable came from the perspective of organizational scaling due to the change management process of having to abandon the process of review and having humans review the code. The management process involved a series of hackathons, workshops, and solution demos during which the usage and its benefits were presented to the wider audience. By having this rigorous change management process, the team were successfully onboarded and trust was gained. It is very important to mention that the whole organization did not digest the change at once. The process of scaling was going from team to group to department to division and finally to the entire organization.

The process of scale and deployment was heavily supported by the upper management, especially the VP of Engineering. This support was extremely crucial to gain trust for the engineers as they were more likely to trust the solution if it was endorsed by the upper management. In terms of the timelines, the aspect of technical deployment was achieved almost instantly due to the already existing infrastructure. The organizational deployment, however, took around three months as the change management process is always a time-consuming process.

Once the solution was successfully scaled, deployed, and people were onboarded, it was time to decide whether we are done with the use case or if we want to give it further iterative optimization and continuous improvement in the next stage.

The decision came to the reasoning that the use case is indeed valuable and the state of the pull request review automation is not perfect yet. New requirements and feedback are constantly coming in, and this gives us a clear indication that the use case is not finished yet and will still require further iterative optimization and continuous improvement in the next stage of AI-native optimization.

AI Native Optimization: The stage of the SAIL framework which concludes the entire process is the AI native optimization. After the use case being successfully designed, analyzed, refined, implemented, scaled, rolled out, it's time to maintain it, optimize it, improve it further, and yield continuous and sustainable value over time. The capital investment in terms of time and resources has already been made and therefore the best option for the team was to try to ensure that the use case meets the current requirements of the engineers and provides value in terms of the pull request review process. The question of what the engineers want and expect can only be measured by one way and it's talking to the customers, in this case the engineers, as this tool is only internal.

Based on the survey done several months after the deployment of the solution, the team gathered more metrics from the already outlined methods.

Metric	**Result Achieved**
Achieve an automated pull request review	System logs and monitoring confirm continuous operation
Reduce the time spent on pull request reviews by 100%	Time spent on pull request reviews reduced by 75%
Increase developer satisfaction with pull request review process by 50%	Developer satisfaction increased by 55%

Based on the gathered numbers in the AI-native optimization step, we can clearly see a difference between the stage of pilot execution and the AI-native optimization. This being mainly in the last metric in terms of the developer satisfaction. The reasoning behind increased developer satisfaction is the fact that the team was able to implement further improvements based on the feedback gathered from engineers and thus fulfilling their needs better. It can also be noticed that the final metric has actually surpassed the original expectations, which is a remarkable achievement on its own. The final metric tells us that two out of the three expected metrics have successfully passed our initial threshold. As already mentioned in the pilot execution phase, the metric regarding the time spent on pull request reviews being reduced by 100% was always a very ambitious goal and therefore the achieved result of 75% is still extremely valuable.

It is also very important to note that this use case will stay here, but hopefully the maintenance will not. The use case should reach its maturity, meaning the point during which engineers are satisfied with the functionality and performance of the solution and therefore the need for further improvements will be minimal. Important to mention, down the line there should be a team maintaining the solution and understand the codebase as well as the architecture behind it.

The team is planned to stay in the stage of AI native optimization as long as the demand for changes and improvements is present. Once the team reaches the point of diminishing returns and the engineers are satisfied with the solution, the team will be disbanded and the maintenance will be handed over to a smaller team responsible for maintaining the solution.

6.3.4 Internal AI Chatbot for Employee Queries

Entry Point: The following use case focuses on a very important and crucial aspect of any organization, being the internal communication and knowledge sharing. The distributed knowledge and lack of well-connected documentation is an issue troubling most of the organizations in software industry. This problem will be solved by ad internal LLM based chatbot with capability of answering questions and providing relevant information in close to real time. This chatbot does not only answer questions per page, but is capable of creating an internal interconnected mind map of knowledge which contains every single detail the company decides to put in it and allow for interal search and knowledge sharing. This capability of restricting in what to add in the knowledge base is crucial as the company is able to restrict any sensitive information, which shal not be availabe for other employees. The chatbot will be able to answer questions regarding internal tools, processes, status of projects and other relevant information which the engineers might be questioning. This chatbot will be able to save time and increase productivity of the engineers as they will not have to search for relevant information and the access to information at any-time and thus reducing the load on employees with knowledge silos.

Application of SAIL: - **Opportunity Scouting:** The opportunity scouting phase for this use case will involve identification of the painpoint of knowledge sharing and internal communication. The entity most likely coming up with this idea is the Junior Engineers and new hires, which have a tough time during onboarding as a new environment in a poorly structured company document-wise. The average time of engineers trying to search for relevant documentation is around 30 minutes per day, which translates to 10 hours per month per engineer. For a team of 1000 engineers this translates to 10000 hours per month of time wasted on searching for relevant documentation. The potential monetary value of this time is around EUR 800,000 per month (assuming EUR 80/hour fully loaded cost). The time and money wasted on searching for relevant information is a very significant amount, which bleads through the hands of the company.

- **Strategic Alignment:** During the strategic alignment phase the team will have to ensure that the initiative alligns with the strategic goals and OKRs of the company. The key strategic goals and OKRs in this case could be
- Reduce time spent searching for information by 50%
- Improve employee satisfaction score regarding access to information by 20%
- Increase onboarding satisfaction score by 15%

These objectives and key results represent an internal improvement of the companies processes. Not nescessarily a direct revenue generating initiative, however the impact on the internal processes and efficiency of the workflows creates time and space for more value adding activities. This indirect value creation is a process which is often overlooked and therefore might create friction in terms of buy-in from management and governance board. However, with correct framing of the value proposition and focus on the time savings and generation of space of engineers to do more value adding activities during this time shall be the priority when negotiating the buy-in.

- **Technical Feasibility & Data Assessment:** From the perspective of technical feasibility the teams will have to evaluate the connections and project management tools which they are using. The company has to realise their infrastructure assets in terms of server capacities both in RAM and Graphics cards as this solution will require an interal LLM to be hosted on premises. An option to outsource this processes opens up a lot of security concerns and therefore the company shall avoid this path. When focusing on appropriate technical feasibility the company needs to follow the outlined formula:
- Size of the company (numbers) * activity rate (% of users which shall be using the tools at once) = Required capacity to host of users at once

If the company has 1000 employees and the activity rate is 10% (100 users at once) the company will need to have a server capacity of around 80GB of RAM and 2 A100 GPUs in order to ensure smooth operation of the LLM based chatbot. The price of a new A100 GPU is around EUR 10,000 and the RAM is around EUR 500/16GB. This means that the initial investment for the hardware will be around EUR 25,000 which is a very reasonable price for the value it is able to deliver as the company bleeds through EUR 80,000 per month on searching for relevant information. The ROI of this investment is very high and the payback period is around 1 month including the price of employees working on the implementation of the solution.

- **Pilot Design:** The stage of pilot design will involve outlining the steps required for the solution to come from an idea to first MVP which is usable for the end users. The first step shall be setting up the hardware and the LLM model and choosing the model which is the most suitable. The initial implementation shall be done with a smaller model like Llama 2 7B or Falcon 7B in order to ensure that the model is able to run on the hardware available. The second step shall be the data collection and cleaning, as the data quality is crucial for the performance of the model. The data shall be collected from internal documentations, either PDFs or tools like Confluence or Sharepoint. Once this data is gathered a process to create a vector database containing this data shall be created. The next step will be the integration of the LLM with the vector database and the creation of a simple UI which is able to query the LLM and provide relevant answers. The final step will be the testing and feedback collection from a small group of users in order to ensure that the solution is working as intended and is able to deliver value. The success metrics for this pilot will be:
- Reduction in time spent searching for information by 40%
- User satisfaction score of 4/5 or higher
- **Pilot Execution:** From the perspective of pilot execution the team will be following the outlined map from the pilot design stage. The pilot design stage should take into consideration deviation from the plan due to unepexcted realms of the AI world. This stage does not only focus on implementation but on rapid review cycles and small testing by external users. The short demos shall be included for external teams to verify quality of the output during the implementation part as the most important metric of this implementation is the quality of output. Once engineers do not trust the output the adoption rate will be logarithim and will plateu quick. The pilot execution is expected to be delivered in 6 weeks as this gives enough time for the engineers to onboard to topic, setup infrastructure, collect data, clean data, train model, integrate model and test the solution. After 6 weeks it is not expecte that the full solution will be delivered but an MVP which fullfils the main requirements and meets the success criteria.
- **Evaluation & Scaling Decision:** From the perspective on whether to scale the solution the main question is what does scaling mean? From the perspective of hardware the need to scale is out of the scope as either the company has to double in size or

the amount of concurent users has to double, if neither of these metrics dramatically change the need to scale hardware-wise is not in the question. The question where to scale or not comes only when the MVP was delivered on a sample of the internal documentation and not the whole internal map was created. The decision to scale in that scenario involves inclusion of more data sources and more data into the vector database and retraining of the model. This stage than becomes very crucial as with more data the model is more likely to hallucinate and provide wrong answers as it might misunderstand concepts incorrectly written in the documentation. The decision to scale also involves the decision on adding new features such as inclusion of links to sources whic the AI chatboot is retrieve the information from. This feature is very important as it increases the transparency of the model and therefore increases the trust of the users in the output. The decision to scale still relies on the outcome of the metrics from the evaluations. These dependent metrics need to be fullfilled in order to proceed with the scaling decision.

- **Scale Deployment:** The scale deployment process will involve introduction of the new data and decision on which features to include. This proccess will be handled by close collaboration of the AI engineers and the end-customers (engineers) leaving feedback on the experience both from UI perspectivity and output quality. This process shall be on the quicker side as the pilot on its own does not change drastically but rather improves the quality of the output and the experience of the users.

- **AI-Native Optimization:** The continuou improvement and optimization will be done by retraining and optimising the internal LLM based chatbot which enhances the productivity of the engineers. Towards the end of the optimization phase new metrics will be calculated and the team will evaluate whether the solution is still delivering value or not. The goal of this stage will be to ensure that the internal LLM based chatbot has close to 100% adoption within the engineering teams. Some tasks of continuous improvement could be:

- Refinement of response accuracy based on user feedback.

- Adding features such as source linking, multi-language support, and voice interaction.

6.4 Summary of Findings

The outlined use cases are a perfect showcase of the immense potential of AI adoption in software companies. The structured approach which the SAIL framework provides allows teams and companies to tackle complex problems in a very manageable way and decompose them into smaller iterative steps. Each of the use cases outlined in this chapter demonstrates significant time and cost savings as well as improvements in productivity and employee satisfaction. As already mentioned, the use cases presented are not only theoretical but have been implemented in real-life scenarios and thus the numbers provided are based on real-life data and feedback from engineers. This elevates the credibility of the outlined use cases and thus provides a solid foundation for further exploration and implementation of similar use cases in other software companies.

In terms of the actual metrics, the following table lists the summary of the actual impact measured across all the outlined use cases:

Use Case	Metrics Gathered	Impact Achieved
AI-based Code Documentation Generation	Time spent on documentation creation Documentation quality Developer satisfaction with documentation process	90% reduction in time spent on documentation creation 22% improvement in documentation quality 55% increase in developer satisfaction
AI-driven Refactoring	Code complexity Code readability Developer satisfaction with refactoring process	28% reduction in code complexity 35% improvement in code readability 45% increase in developer satisfaction
Pull Request Review Automation	Automated pull request review achievement Time spent on pull request reviews Developer satisfaction with review process	Continuous automated operation confirmed via system logs 75% reduction in time spent on reviews 55% increase in developer satisfaction
Internal LLM Chatbot for Knowledge Sharing	Time spent searching for information User satisfaction score Employee satisfaction regarding information access Onboarding satisfaction	40% reduction in time spent searching for information User satisfaction score of 4/5 or higher achieved

The outlined use cases showcase the immense potential of AI adoption in software companies. The structured approach provided by the SAIL framework allows companies to tackle complex problems in a manageable way, breaking them down into smaller, iterative steps. Each use case demonstrates significant time and cost savings, as well as improvements in productivity and employee satisfaction.

Based on the gathered metrics, it is evident that AI adoption can lead to significant improvements. As a summary, in this chapter we can conclude that the outlined use cases are a perfect showcase of the immense potential of AI adoption in software companies. This chapter detailed the application of the SAIL framework to four distinct use cases, each demonstrating significant time and cost savings, as well as improvements in productivity and employee satisfaction. By incorporating management and engineering perspectives, the SAIL framework ensures that AI initiatives are strategically aligned, technically feasible, and effectively executed. The metrics gathered from each use case provide concrete evidence of the value that AI can bring to software companies when adopted in a structured manner. These findings set the stage for a deeper discussion on the implications of AI adoption in the subsequent chapter.

Chapter 7

Discussion

7.1 Key Insights

The entire objective of this thesis was to create a structured framework which would help software companies to adopt AI in a structured manner and increase the probability of success. The creation of the SAIL framework serves as the first life-cycle framework which works on iterative steps and adapts to each use case accordingly. The SAIL framework decomposes a large AI initiative into 8 steps each with a quality gate being the decision to proceed, iterate or stop. In regards to the insights from the use cases, the most significant findings are the cleverness of the framework to really granularize the complex problem of AI adoption into smaller manageable pieces. The ease of the implementation of the framework was one of the key success factors as it does not require a lot of resources to implement it. Other frameworks which are focused on project management such as PRINCE2 or Agile are very resource intensive and require a lot of time and effort to implement them. The lightweight nature of SAIL was one of the key success factors. The most resource-intensive stage of SAIL is the pilot execution phase, which in fact is not actually resource-intense as the only resource-intense part is the actual implementation of the pilot and thus it being the engineering time, which would otherwise be spent on feature development. The surprising success factor is the fact that the governance board does not require a lot of time and effort as the decision making is very straightforward and the quality gates are very clear.

The role of the governance board faded out when use cases utilize the same or similar technology, as the risks and potential complications are already known and therefore the governance board does not have to spend a lot of time evaluating the risks. A great example for this is the usage of GitHub Copilot for code generation and assistance. The risks and potential complications are already known and therefore the governance board does not have to spend a lot of time evaluating the risks.

In terms of insights which were gathered, it can be seen that the SAIL framework is a great fit for problems which usually require a lot of trial and error and are not straightforward. The more complex the problem the more value SAIL is able to deliver as it breaks down the complexity into smaller manageable pieces.

The outlined numbers showcase the immense potential of AI adoption in software companies; this however is only possible to be achieved with a structured approach which companies lack. If multiple of these use cases are implemented at once and proper change management is done, the impact on the productivity of the engineers and the overall efficiency of the company is truly revolutionary. This marks a historic moment for companies to leverage AI in order to create sustainable competitive advantage.

A very important insight which was gathered is the fact that the SAIL framework is not actually a guarantee of success. The best comparison for SAIL is a GPS navigation system. It provides a structured approach and a roadmap to the destination, however it does not guarantee that you will reach the destination.

This type of mentality positions the SAIL framework as a means of delivery and not as a final product. The success of the AI adoption initiative still relies on the input of ideas and creativity of the engineers and the management. That is one of the reasons why the SAIL framework focuses on an “Engineer-first” mentality as engineers know their daily struggles and pain points the best.

7.2 Contributions to Theory and Practice

The SAIL framework is contributing to both theory and practice in several ways. From the perspective of theoretical contribution, the SAIL framework is linking to Dynamic Capabilities Theory, as it operationalizes the sensing, seizing and transforming capabilities in the process of AI adoption. The SAIL framework actually extends the current standard by converting both physical resources and digital assets into capabilities which are able to create value. The value in this case being a custom AI agent, which is able to solve a specific problem and thus create value for the company.

From the perspective of innovation diffusion theory, the SAIL framework extends the current theory by providing concrete metrics and implementation steps rather than being too abstract. The fact that the pilots under the SAIL framework emphasized on iteration, the probability for success is in fact higher due to a feedback loop gathered from each iteration. Each iteration and each pilot does not only serve as a proof of concept but also as a learning opportunity. The innovation diffusion theory works on the principle of trial and error and thus the SAIL framework embraces this principle by having multiple pilots rather than one big bang approach.

The actual criticism of Innovation Diffusion Theory of being too abstract was addressed and proven wrong by providing concrete implementation steps and metrics which can be measured and tracked. As the SAIL framework emphasizes on practicality in terms of having clear steps and quality gates, it provides a clear roadmap for companies to follow when adopting AI.

The great aspect of the SAIL framework is that it guides the development of new innovative branches rather than focusing on existing ones. The entire premise of it is to help CTOs and engineering leaders navigate the complex world of AI adoption and thus yield sustainable advantage from internal AI initiatives. By introducing quality gates, the SAIL framework introduces a level of transparency and a reality check which helps prioritize and rank high-impact AI projects and thus reduce the risk of implementing high effort low impact AI initiatives.

The structured approach allows engineering teams to start with brainstorming and putting an engineer-first mindset as a root cause of internal struggles. The structured approach also allows navigation during uncertainty via opportunity scouting and thus provides a replicable roadmap through multi-pilot design.

It could be noticed that during the execution of the pilots the impact of the SAIL framework depends heavily on the creativity and quality of the use cases proposed. The quality of the output can be only as good as the input provided. As already mentioned, the SAIL framework does not guarantee success in terms of the impact on the organization; it only guarantees a structured approach towards AI adoption. The actual ROI is only guaranteed by the creativity and quality of the use cases proposed.

The main impact on the daily lives of engineers is the fact that the engineers have a backbone and a structure to follow when proposing AI initiatives instead of having to figure it out on their own and work in a chaotic manner.

The framework, however, does still have its limitations and areas of improvement which will be discussed in the next section.

7.3 Limitations

This entire thesis has focused on introduction of the SAIL framework as a structured lifecycle framework for AI adoption in software companies. This implies that the framework is only applicable to software companies and not other industries. This might be seen as a limitation as other industries such as financial services, manufacturing and others might also be interested in AI adoption; however, due to the specifics of these industries the SAIL framework might not be a perfect fit due to increased regulatory requirements and slower pace of change.

Another limitation is the rapid evolution of the AI landscape. AI landscape and technology is changing on daily basis and therefore it is very important to mention the fact that when a new use case starts to be developed with the SAIL framework it might happen that by the end of the development cycle the technology has already changed and therefore the initial assumptions are no longer valid. This is however not the limitation of the SAIL framework itself but rather a limitation of the AI world as a whole. The rapid evolution of new technology and design principles and paradigms is a challenge for any framework and suddenly the teams are entering unknown territory.

What however is the current limitation of the SAIL framework is the fact that it's only composed out of eight steps. Someone might argue that it's too broad and generic and does not give concrete steps which need to be done. The SAIL framework is however designed to be still hands on and also apply to various different use cases. As the landscape of AI does not encompass only generative AI or machine learning algorithms or natural language processing, but also computer vision, autonomous agents and others, the SAIL framework had to be designed to still encompass all of these.

7.4 Implications and Next Steps

In terms of the implications and the next steps, we have to differentiate the implications for theory and practice. From the practical point of view, the SAIL framework is easy to apply and easy to understand. Based on the use cases analyzed throughout this entire thesis, it's ready to be deployed into organizations. The next steps, therefore, are going to be applying the SAIL framework in real-life organizations and gathering more data and feedback from the users.

From the theoretical point of view, it opens endless doors for further research and exploration. Extra research and exploration of each of the steps of the SAIL framework can be done in order to further improve the framework and make it more robust. Each of the steps can be further explored in order to provide more concrete steps and guidelines for companies to follow.

There is no limit in terms of how much further research can be done in order to improve the framework and make it more reliable and robust.

What would be considered as next steps would be research in terms of how to potentially expand and try to integrate the SAIL framework into already existing industries such as financial services, manufacturing and others. This would obviously require extra research, understanding of the industries, regulations, and specifics, but it would be a great next step in terms of implementing, revolutionizing and bringing innovation into more conservative and slower-paced industries.

What, however, has to be mentioned as a next step is the fact that the current project management frameworks such as PRINCE2 and Agile do not encompass AI adoption and therefore there is a great opportunity to integrate the SAIL framework into these already existing frameworks in order to provide a more holistic approach towards project management with AI adoption. This would be a great next step in terms of further research and exploration.

7.5 Reflections on the framework and its application

The SAIL framework represents a significant advancement in the structured adoption of AI within software organizations. The major AI-based use cases which SAIL is perfect at employing and solving are use cases which are focused on internal process optimization and enhancement of productivity.

In terms of all the use cases, the key pattern in them is either code base optimisation or workflow optimisation. On the other hand, these use cases are in fact the key processes which software companies are focusing on in order to increase their productivity and efficiency. Coding takes majority of the time of software engineers and therefore any optimisation in this area is a direct increase in productivity. From the managerial perspective the use cases for workflow optimisation and improvement are mainly focused on orchestrating the engineering workforce in the most efficient and sustainable manner. If managers are able to achieve this better and faster with AI use cases the productivity of the whole organisation increases.

Use cases which do not make sense to be implemented with the SAIL framework are use cases which are not directly linked to internal process optimisation or are too simple to be implemented with SAIL. A perfect example for this is the usage of already existing AI-based tools and simply enabling them. SAIL as a framework requires complexity and therefore simple use cases such as turning on page summarisation in Confluence for all engineers or enabling Grammarly for all employees do not make sense to be implemented with SAIL. These 2 outlined use case can be implemented by simply turning on the feature and therefore the entire process of SAIL is not required as these processes do not need granularization and decomposition into smaller manageable pieces.

SAIL is a perfect fit for medium to complex use cases which require a structured progress for initiatives to be successful. Use cases which might come with a level of risk and uncertainty are also a perfect fit for SAIL as it has multiple quality gates, review stages and stakeholder involvement. The complexity of the use case is directly proportional to the value SAIL is able to deliver. The more complex the use case the more value SAIL is able to deliver as it breaks down the complexity into smaller manageable pieces and therefore increases the probability of success.

Chapter 8

Conclusion and Final Thoughts

8.1 Summary of Research

This research is aimed at addressing the critical need for a structured approach of AI adoption in software companies. The entire research is trying to answer the question of "How can software organizations effectively adopt and scale AI technologies through a structured lifecycle framework?" and the simple answer to this question is the SAIL framework. This framework was developed based on an 8-step iterative process, each step having a clear quality gate with the main objective of pushing the boundaries of the complex problem of AI adoption into smaller manageable pieces. This type of a new, revolutionary and innovative framework answers the research question as it provides the structure and the roadmap for companies to follow in order to successfully adopt AI technologies. The question is stated in a way of how to adopt and scale AI technologies, and that is exactly what SAIL encompasses. The core functionality of the framework being focus on iterative pilots and rapid review cycles allows for quick learning and adaptation, which is crucial in the fast-evolving AI landscape.

The key outcomes from the debate in chapter 7 clearly show the unique value proposition of having a structured approach to AI adoption. The outlined use cases demonstrate significant potential for an increase in internal efficiency of processes and productivity of engineers. The outlined metrics show everything which is possible to achieve with a solid and structured approach. The concrete set of metrics such as time savings, cost savings and other benefits provide a clear benchmark for evaluating the success of AI initiatives.

The mentality of the framework being "Engineer-first" attribute of the framework as its starting point is opportunity scouting by engineers. The outcome of the framework will be only as good as its input. By having high-quality input from engineers which understand their domain the entire pipeline of SAIL will be benefited.

In order to give a clear summary of this research the SAIL framework can be summarized as following:

- SAIL framework is a structured lifecycle framework for AI adoption in software companies.
- It consists of 8 iterative steps with clear quality gates working on the principle of "Iterate, Proceed, Stop".
- It guides companies from opportunity scouting to AI-native optimization by focusing on governance, strategic alignment, ability to pilot and scale initiatives.
- It is emphasizing an "Engineer-first" mentality which leverages the expertise of engineers
- The framework was successfully validated through 4 diverse use cases showcasing its versatility and applicability.
- SAIL provides a full replicable roadmap for medium to complex AI initiatives from scouting to scaling.

8.2 Integrated Contributions of the SAIL Framework

The SAIL framework makes a very unique contribution in incorporating multiple aspects of strategic, technical, and operational considerations of AI adoption into a single cohesive navigable process. Unlike other existing models, which are very much heavy and rigid, SAIL is lightweight, flexible, iterative while still providing discipline through its quality gates.

Its novelty lies in three core aspects:

1. Operationalization of AI Adoption as a Lifecycle Capability

- SAIL treats AI adoption as a repeatable organizational capability rather than a one-off project, this meaning that it is creating a dynamic capability for the organization to continuously sense, seize and transform with AI technologies. This principle allows for new technology being continuously integrated into the organization as it evolves.

2. Practical Roadmap with Built-in Metrics

- By providing concrete steps and success metrics the iterative pilot strategy allows organizations to learn and adapt quickly while still observing and measuring impact of the solutions. This practical roadmap is a significant advancement over abstract models which lack actionable guidance and accountability.

3. Bridging Strategy, Governance and Innovation

- SAIL offers a unique balance between strategic alignment, governance oversight and innovation agility. It allows teams to experiment and pilot while ensuring alignment with broader organizational goals and risk management through governance involvement.

Together these three key aspects position SAIL as not only a theoretical contribution but also a highly practical framework acting as an AI enabler for software organizations.

8.3 Limitations and Caveats

As with any great and innovative idea, the SAIL framework comes with its own limitations and caveats.

- The first limitation currently standing. In the scope of sale framework being used only for software companies. While inventing the sale framework The primary focus was software companies, and all the aspects of it were tailored precisely towards them. The limitation comes in the narrowness of the scope. And uncertainty whether? This framework will also fit. In other process industries.
- Second main limitation of the framework is its input-reliance. The framework and its impact on the company will be only as good as its input. If the engineers reate low-quality input the entire pipeline of SAIL will be affected in lower ROI and lesser impact on the company. The poor input might lead to poor output.
- Third limitation is the fact that SAIL is not a guarantee of success due to the aspect of change management and human factors. The framework provides a structured approach and a roadmap, however the guarantee to actually reach the destination is not there due to human nature of not trusting new technologies. This aspect is unfortunately out of the scope of SAIL and is a matter of change management and leadership. This dependency on human factors might come with a level of uncertainty and risk which is not accounted for in the framework, however might also extremely benefit if the change management is done properly and people have a natural curiosity and interest in new technologies.

8.4 Practical Implications

For any organization whose scope is directly software And its chooses to apply the cell framework the implications and the benefits are truly revolutionary.

By applying the sale framework, the organizations can expect to achieve:

- Structured approach to AI adoption. Based on the outcome of this thesis, it can be stated that several teams have managed to adopt AI solutions In a structured manner by following sale framework. This structured approach minimized chaos optimized results and led to succesful adoption of AI technologies.

- Increase productivity and efficiency. Our use cases have demonstrated the potential of saving vast amounts of time and costs by automating significant processes, while not only freeing up engineers time but also increasing their satisfaction and engagement.
- Faster innovation cycles. The principle of introducing decision gates allows teams to iterate, reject, or proceed based on the current state of the use case. This gives the team the ability To be extremely. agile and adapt quickly to the fast-evolving AI landscape. The framework emphasizes on the principle of "Fail-fast-learn-faster" which is crucial in the current fast-moving AI world.
- Reduced uncertainty and risk. In decision gates inside of the framework Serve as a mirror in terms of reality check and progress evaluation. By having clearly established decision gates, the engineers and management are always up to date and are completely eliminating any personal biases and feelings towards a use case. As it often happens, managers try to commit to a Decision and stick to it through thick and thin, even if the reality is different. By having clear quality gates, the risk of committing to a poor use case is minimized.
- Increased innovation. The framework encourages engineers to take action, innovate and create new ideas which might be impactful for the business. By having structured approach. The company can leverage the ideas of engineers into a strategic position. Which elevates the entire company and pushes the boundaries of what is currently possible.

The practical implications of SAIL extend beyond just the immediate benefits of individual AI initiatives. The framework fosters a culture of innovation, learning and curiosity within the organization. It tries to position the company as a learning organization and emphasizes on the importance of continuous improvement and mentality of "Engineer-first" and "Fail-fast-learn-faster". It puts down the fear of failure and encourages experimentation and creativity as the key drivers of success in the fast-evolving AI landscape.

It is important to note that the successful adoption of SAIL and adoption of AI in the company truly lies in the commitment of the leadership and the engagement of engineers. The framework on its own is just a set of steps with quality gates, however the real success comes from the people and their commitment to leverage this opportunity. The framework is only a means of delivery and not the final product. It might give all the opportunity and direction, however the actual journey and destination is up to the people.

Bibliography

Agrawal, A., Gans, J. S., and Goldfarb, A. (2022). The economics of artificial intelligence: Implications for the future of work. *Journal of Economic Perspectives*, 33(2):3–30.

Ahmad, A., Alshurideh, M., and Al Kurdi, B. (2021). Digital transformation metrics: a conceptual view. *International Journal of Data and Network Science*, 5(4):314–328.

Al-Ali, M. and Marks, A. (2022). A digital maturity model for the education enterprise. *Perspectives: Policy and Practice in Higher Education*, 26(1):1–13.

Ali, Z. and Nicola, H. (2018). Accelerating digital transformation: Leveraging enterprise architecture and ai in cloud-driven devops and dataops frameworks.

Almaiah, M. A. (2022). Measuring institutions' adoption of artificial intelligence applications in online learning environments: Integrating the innovation diffusion theory with technology adoption. *Electronics*, 11(20):3291.

Amling, A. (2024). A guiding framework to enable successful ai adoption. OpenReview.

Anny, D. (2024). Integrating ai-driven decision-making into enterprise architecture for scalable software development.

AWS (2024). What is mlops? - machine learning operations explained. https://aws.amazon.com/what-is/mlops/. Retrieved from https://aws.amazon.com/what-is/mlops/.

Baier, L., Jöhren, F., and Seebacher, S. (2019). Challenges in the deployment and operation of machine learning in practice. *Proceedings of the European Conference on Information Systems (ECIS)*. Study on MLOps challenges in production environments.

Bain & Company (2024). Ai: The ambitions are bold, but the talent is scarce. https://www.bain.com/insights/ai-the-ambitions-are-bold-but-the-talent-is-scarce-snap-chart/.

Benbya, H., Davenport, T. H., and Pachidi, S. (2020). Artificial intelligence in organizations: Current state and future opportunities. *MIS Quarterly Executive*, 19(4):ix–xxi. Special issue on AI in organizations.

Berente, N., Gu, B., Recker, J., and Santhanam, R. (2021). Managing artificial intelligence. *MIS Quarterly*, 45(3):1433–1450. Special issue on managing AI in organizations.

Bhatia, R. and Sandiri, S. (2025). The evolution of digital financial architecture: Artificial intelligence-driven agility and scalability in enterprise solutions & customer excellence. *Journal of Next-Generation Research 5.0*.

Brynjolfsson, E., Li, D., and Raymond, L. R. (2023). Generative ai at work. *National Bureau of Economic Research Working Paper*.

Canat, M., Català, N. P., and Jourkovski, A. (2018). Enterprise architecture and agile development: Friends or foes? In *2018 IEEE 22nd International Enterprise Distributed Object Computing Conference (EDOC)*, pages 1–10.

Chen, T. and Gascó-Hernandez, M. (2024). The adoption and implementation of artificial intelligence chatbots in public organizations: Evidence from us state governments. *The American Review of Public Administration*.

Chowdhury, S., Dey, P., Joel-Edgar, S., Bhattacharya, S., Rodriguez-Espindola, O., Abadie, A., and Truong, L. (2022). Unlocking the value of artificial intelligence in human resource management through ai capability framework. *Human Resource Management Review*, 32(4).

Chui, M., Manyika, J., Miremadi, M., Henke, N., Chung, R., Nel, P., and Malhotra, S. (2018). Notes from the ai frontier: Insights from hundreds of use cases. *McKinsey Global Institute Discussion Paper*.

Coombs, C., Hislop, D., Taneva, S. K., and Barnard, S. (2020). The strategic impacts of intelligent automation for knowledge and service work: An interdisciplinary review. *The Journal of Strategic Information Systems*, 29(4).

Dasgupta, A. and Wendler, S. (2019). Ai adoption strategies. Technical report, Oxford Internet Institute.

Davenport, T. H., Guha, A., Grewal, D., and Bressgott, T. (2020). How artificial intelligence will change the future of marketing. *Journal of the Academy of Marketing Science*, 48(1):24–42.

Davenport, T. H. and Ronanki, R. (2019). Artificial intelligence for the real world. *Harvard Business Review*, 96(1):108–116.

Dearing, J. W. (2009). Applying diffusion of innovation theory to intervention development. *Research on Social Work Practice*, 19(5):503–518.

Deloitte (2023). Digital maturity index. https://www.deloitte.com/global/en/Industries/industrial-construction/perspectives/digital-maturity-index.html. Accessed November 2025.

Deloitte (2025). Ai trends 2025: Adoption barriers and updated predictions. Deloitte Insights. Retrieved from https://www.deloitte.com/us/en/services/consulting/blogs/ai-adoption-challenges-ai-trends.html.

Dwivedi, Y. K., Hughes, L., Ismagilova, E., Aarts, G., Coombs, C., Crick, T., Duan, Y., Dwivedi, R., Edwards, J., Eirug, A., et al. (2021). Artificial intelligence (ai): Multidisciplinary perspectives on emerging challenges, opportunities, and agenda for research, practice and policy. *International Journal of Information Management*, 57:101994.

Dzreke, S. (2024). The competitive advantage of ai in business: A strategic imperative. *Journal of Strategic Management*.

Enholm, I. M., Papagiannidis, E., Mikalef, P., and Krogstie, J. (2022). Artificial intelligence and business value: a literature review. *Information Systems Frontiers*, 24:1709–1734.

Ern, S. L. W., Yang, H., Wu, Y., Ambakkat, R. S., and Dilnutt, R. (2025). Modifying togaf® to optimize artificial intelligence and strategic value for organizations. *Enterprise Architecture Professional Journal (EAPJ)*.

Felzmann, H., Villaronga, E. F., Lutz, C., and Tamò-Larrieux, A. (2019). Transparency you can trust: Transparency requirements for artificial intelligence between legal norms and contextual concerns. *Big Data & Society*, 6(1):2053951719860542.

Fenwick, A., Molnar, G., and Frangos, P. (2024). The critical role of hrm in ai-driven digital transformation: a paradigm shift to enable firms to move from ai implementation to human-centric adoption. *Discover Artificial Intelligence*.

Fitriani, L., Khodra, M. L., and Surendro, K. (2023). Togaf-based enterprise architecture framework for utilizing artificial intelligence. In *2023 International Conference on Computer Science, Information Technology and Engineering (ICoSITE)*, pages 1–6.

Forth, P., Reichert, T., de Laubier, R., and Chakraborty, S. (2020). Flipping the odds of digital transformation success. Boston Consulting Group. Retrieved from https://www.bcg.com/publications/2020/increasing-odds-of-success-in-digital-transformation.

Fosso Wamba, S., Queiroz, M. M., and Trinchera, L. (2021). Dynamics between blockchain adoption determinants and supply chain performance: An empirical investigation. *International Journal of Production Economics*, 229.

Fountaine, T., McCarthy, B., and Saleh, T. (2019). Building the ai-powered organization. *Harvard Business Review*, 97(4):62–73. Framework for AI transformation with leadership, technology, and data dimensions.

Gama, F., Tyskbo, D., Nygren, J., Barlow, J., Reed, J., Svedberg, P., and Nygren, J. M. (2022). Implementation frameworks for artificial intelligence translation into health care practice: scoping review. *Journal of Medical Internet Research*, 24(1):e32215.

Garousi, V., Keleş, A. B., Değirmenci, S., and Jafarov, Z. (2025). Assessing individual and organizational competency in ai-assisted software engineering: The ai4se-mm maturity model. *arXiv preprint*. Preprint available at ResearchGate.

Gupta, S., Meier-Hellstern, K., and Satterlee, M. (2018). Artificial intelligence for enterprise networks. In *Artificial Intelligence for Enterprises*. Taylor & Francis.

Haefner, N., Parida, V., Gassmann, O., and Wincent, J. (2023). Implementing and scaling artificial intelligence: A review, framework, and research agenda. *Technological Forecasting and Social Change*, 197:122878. Key framework article on AI implementation phases: proving concept, productionizing, and platformizing.

Hassan, M., Kushniruk, A., and Borycki, E. (2024). Barriers to and facilitators of artificial intelligence adoption in health care: scoping review. *JMIR Human Factors*.

Hechler, E., Oberhofer, M., and Schaeck, T. (2020). *Deploying AI in the Enterprise: IT Approaches for Design, DevOps, and Governance*. Springer.

Holmström, J. (2022). From ai to digital transformation: The ai readiness framework. *Business Horizons*, 65(3):329–339.

Hradecky, D., Kennell, J., Cai, W., and Davidson, R. (2022). Organizational readiness to adopt artificial intelligence in the exhibition sector in western europe. *International Journal of Information Management*, 62:102431.

Iansiti, M. and Lakhani, K. R. (2020). Competing in the age of ai: Strategy and leadership when algorithms and networks run the world. *Harvard Business Review*, 98(1):60–67.

IBM (2024). What is ai governance? https://www.ibm.com/think/topics/ai-governance.

Jarrahi, M. H., Memariani, A., and Guha, S. (2023). The principles of data-centric ai: Toward a human-centered ai paradigm. *Communications of the ACM*, 66(7).

John, M. M. (2025). An empirical guide to mlops adoption. *Information and Software Technology*, 171:107001.

Jöhnk, J., Weißert, M., and Wyrtki, K. (2021). Ready or not, ai comes—an interview study of organizational ai readiness factors. *Business & Information Systems Engineering*, 63(3):277–293.

Kaddoumi, T. and Watfa, M. (2022). A foundational framework for agile enterprise architecture. *International Journal of Lean Six Sigma*, 13(2):313–333.

Keding, C. (2021). Understanding the interplay of artificial intelligence and strategic management: four decades of research in review. *Management Review Quarterly*, 71:91–134.

Kemell, K. K., Saarikallio, M., and Nguyen-Duc, A. (2025). Still just personal assistants?–a multiple case study of generative ai adoption in software organizations. *Information and Software Technology*.

Kurup, S. and Gupta, V. (2022). Factors influencing the ai adoption in organizations. *Metamorphosis*.

Kučević, E., Grünewald, F., and Schanz, N. (2024). Towards a simplified ai adoption framework: Success factors for the implementation of artificial intelligence information systems. In *International Conference on Human-Computer Interaction*, pages 69–85. Springer.

Lakarasu, P. (2022). Mlops at scale: Bridging cloud infrastructure and ai lifecycle management. *Social Science Research Network*. SSRN: 5272259.

Lichtenthaler, U. (2020). Five maturity levels in developing ai: From isolated ignorance to systematic intelligence. *Journal of Innovation Management*, 8(4):39–50.

Liu, Y. (2024). How ai impacts companies' dynamic capabilities. *Journal of Management Studies*.

Madanchian, M. and Taherdoost, H. (2025). Ethical theories, governance models, and strategic frameworks for responsible ai adoption and organizational success. *Frontiers in Artificial Intelligence*.

Makarius, E. E., Mukherjee, D., Fox, J. D., and Fox, A. K. (2020). Rising with the machines: A sociotechnical framework for bringing artificial intelligence into the organization. *Journal of Business Research*, 120:262–273.

McKinsey & Company (2024). Mlops so ai can scale. https://www.mckinsey.com/capabilities/tech-and-ai/our-insights/mlops-so-ai-can-scale.

McKinsey & Company (2025). The state of ai in 2025: Agents, innovation, and transformation. https://www.mckinsey.com/capabilities/quantumblack/our-insights/the-state-of-ai. Accessed November 2025.

Mecca, A. (2024). The influence of artificial intelligence implementation on firm scaling: an exploratory approach. Master's thesis, Luleå University of Technology.

Mikalef, P. and Gupta, M. (2021). Artificial intelligence capability: Conceptualization, measurement calibration, and empirical study on its impact on organizational creativity and firm performance. *Information & Management*, 58(3):103434.

Nogare, D. and Silveira, I. F. (2025). Mlops for machine learning model lifecycle automation—a systematic literature review. *Authorea Preprints*.

Nwashili, O. G. (2025). Scaling ai features in large organizations: A product management perspective. *International Journal of Economics and Business Management*. Available at irasspublisher.com.

Pandiri, S. (2024). Bridging the gaps in ai transformation: An evidence-based framework for scalable adoption. *California Management Review*.

Parlov, N., Mateša, B., and Mladinić, A. (2025). Structuring ai risk management framework: Eu ai act fria, gdpr dpia and iso 42001/23894. In *2025 14th Mediterranean Conference on Embedded Computing (MECO)*.

Patnaik, P. (2024). Exploring determinants influencing artificial intelligence adoption, reference to diffusion of innovation theory. *Technological Forecasting and Social Change*, 203:123345.

Praveen, R., Shrivastava, A., Sharma, G., and Sreenivasu, M. (2024). Overcoming adoption barriers: Strategies for scalable ai transformation in enterprises. In *2024 International Conference on Intelligent Systems and Emerging Technologies*. IEEE.

Pumplun, L., Tauchert, C., and Heidt, M. (2019). A new organizational chassis for artificial intelligence-exploring organizational readiness factors. *Proceedings of the European Conference on Information Systems (ECIS)*.

Radhakrishnan, J., Gupta, S., and Prashar, S. (2022). Understanding organizations' artificial intelligence journey: A qualitative approach. *Pacific Asia Journal of the Association for Information Systems*.

Raisch, S. and Krakowski, S. (2021). Artificial intelligence and management: The automation-augmentation paradox. *Academy of Management Review*, 46(1):192–210.

Ransbotham, S., Khodabandeh, S., Kiron, D., Candelon, F., Chu, M., and LaFountain, B. (2020). Expanding ai's impact with organizational learning. *MIT Sloan Management Review*. Survey of 2,500+ companies on AI adoption challenges.

Rinta-Kahila, T., Penttinen, E., Salovaara, A., and Soliman, W. (2022). From resistance to adoption: Exploring the organizational challenges and opportunities of ai implementation. *Information Systems Frontiers*.

Robertson-Dunn, B. (2012). Beyond the zachman framework: Problem-oriented system architecture. *IBM Journal of Research and Development*, 56(4):1–10.

Rogers, E. M. (2003). *Diffusion of innovations*. Free Press, 5 edition.

Schneider, J., Abraham, R., Meboldt, M., and Vom Brocke, J. (2023). The anatomy of an ai-powered organization: Cognitive computing, innovation diffusion, and dynamic capabilities. *Business & Information Systems Engineering*, 65:369–385.

Schrage, M., Muttreja, V., and Kwan, A. (2022). How the wrong kpis doom digital transformation. *MIT Sloan Management Review*, 63(4):1–10.

Sepanosian, T., Milosevic, Z., and Blair, A. (2024). Scaling ai adoption in finance: modelling framework and implementation study. In *International Conference on Business Intelligence and Big Data*, pages 195–208. Springer.

Shaban, A. A. and Zeebaree, S. R. M. (2025). Building scalable enterprise systems: The intersection of web technology, cloud computing, and ai marketing.

Sharma, R. (2024). Scaling ai operations: Designing effective enterprise infrastructure. In *AI and the Boardroom: Insights into Governance*. Springer.

Sherson, J., Rabecq, B., Dellermann, D., and Rafner, J. (2023). A multi-dimensional development and deployment framework for hybrid intelligence. In *HHAI 2023: Augmenting Human Intellect, Proceedings of the Second International Conference on Hybrid Human Artificial Intelligence*, volume 368 of *Frontiers in Artificial Intelligence and Applications*, pages 429–432. IOS Press.

Shrestha, Y. R., Ben-Menahem, S. M., and von Krogh, G. (2019). Organizational decision-making structures in the age of artificial intelligence. *California Management Review*, 61(4):66–83.

Singh, J., Mishra, V., Kathuria, S., and Kharub, M. (2025). Demand-side dynamics of ai marketplace adoption: Aim2 prescriptive framework for healthcare. *Journal of Enterprise Information Management*.

Start-Up Nation Central (2024). Israeli tech ecosystem overview 2024. https://lifinder.startupnationcentral.org/reports/2024-annual-report?__cf_chl_tk=M75HdmQzHVy31UV6guei0jzCj.WnknBwjC9QEvBwYwE-1762609619-1.0.1.1-2rZ7SqeK6E4AIQ8HaqtIlzPq1h5kuy8crNZzINyqvFk.

Startup Genome (2024). Global startup ecosystem report 2024. https://startupgenome.com/report/the-global-startup-ecosystem-report-2024/introduction.

Stone, J., Patel, R., Ghiasi, F., and Mittal, S. (2025). Navigating mlops: Insights into maturity, lifecycle, tools, and careers. *IEEE Transactions on Artificial Intelligence*.

Sundaramurthy, S. K. and Ravichandran, N. (2022). Ai-powered operational resilience: Building secure, scalable, and intelligent enterprises. *Artificial Intelligence and Machine Learning Research*.

Tamburri, D. and Tonnarelli, M. (2022). Challenges in adoption and scaling of ai: A case study at a high-tech firm and research roadmap. In *International Workshop on Engineering Artificial Intelligence Systems*, pages 53–68. Springer.

Teece, D. J., Pisano, G., and Shuen, A. (1997). Dynamic capabilities and strategic management. *Strategic Management Journal*, 18(7):509–533.

The Open Group (2022). The togaf standard, version 10 enterprise edition. The Open Group.

Vidgen, R., Hindle, G., and Randolph, I. (2020). Exploring the ethical implications of business analytics with a business ethics canvas. *European Journal of Operational Research*, 281(3):491–501. Framework for AI ethics and governance.

Wamba-Taguimdje, S.-L., Fosso Wamba, S., Kala Kamdjoug, J. R., and Tchatchouang Wanko, C. E. (2021). Influence of artificial intelligence (ai) on firm performance: the business value of ai-based transformation projects. *Business Process Management Journal*, 27(7):1893–1924.

Westerman, G., Bonnet, D., and McAfee, A. (2019). *Leading digital: Turning technology into business transformation*. Harvard Business Review Press.

Wilson, H. J. and Daugherty, P. R. (2023). Collaborative intelligence: Humans and ai are joining forces. *Harvard Business Review*.

Yablonsky, S. (2021). Ai-driven platform enterprise maturity: From human led to machine governed. *Kybernetes*, 51(10):2977–3011.

Zachman, J. A. (1987). A framework for information systems architecture. *IBM Systems Journal*, 26(3):276–292.