STRATEGY TO DEVELOP PROJECT BY USING HYBRID APPROACH (AGILE AND DEVSECOPS METHODOLOGIES)

by

SUNAINA PAUL

DISSERTATION

Presented to the Swiss School of Business and Management Geneva

In Partial Fulfillment

Of the Requirements

For the Degree

DOCTOR OF BUSINESS ADMINISTRATION

SWISS SCHOOL OF BUSINESS AND MANAGEMENT GENEVA

MARCH , 2025

STRATEGY TO DEVELOP PROJECT BY USING HYBRID APPROACH

(AGILE AND DEVSECOPS METHODOLOGIES)

by

SUNAINA PAUL

APPROVED BY

Dissertation Chair - Dr. Gualdino Cardoso

RECEIVED/APPROVED BY:

Admissions Director

Dedication

I dedicate this thesis to my father Sri Vijay Nath and my late brother Vishal Pal for their unwavering belief in me. Their constant support and encouragement have shaped my positive mindset, reinforcing my belief that everyone has something valuable to contribute—whether to our home, our organization, or society at large. Your faith in me has been a guiding force, and for that, I am deeply grateful.

I also dedicate this thesis to my department, where I found the motivation to grow as a leader and bring greater value to my team. Your support, collaboration, and inspiration have played a crucial role in this journey.

This work is dedicated to all of you—thank you.

Acknowledgements

Completing this doctoral research has been an incredible journey, and I owe my gratitude to many who stood by me through it all.

First, I thank God for giving me the strength, health, and determination to see this journey through.

I am deeply grateful to Professor Dr. Minja Bolesnikov, my supervisor and dissertation chair, for his guidance, patience, and constructive feedback. Your support made a significant difference in my academic journey. I also extend my thanks to SSBM's faculty, for their encouragement and collaboration.

A special thanks to my department team members, for their valuable insights and continuous support in refining my research.

Above all, I want to express my deepest gratitude to my husband Rudra Singh, my mother-in-law Seema Singh, my mother Gayatri Devi, my brother Brijbhan Pal, and my son Advay Singh. Their love, sacrifices, and unwavering belief in me gave me the strength to persevere.

To each person who has contributed to this journey in any way—thank you from the bottom of my heart.

ABSTRACT

STRATEGY TO DEVELOP PROJECT BY USING HYBRID APPROACH

(AGILE AND DEVSECOPS METHODOLOGIES)

Sunaina Paul
2025

Dissertation Chair: Dr. Gualdino Cardoso

A project is undertaken to create unique product, result, or services by executing set of activities to meet project objective. The success of project is measured towards achievement of project objective, project quality, timelines, budget compliance and degree of customer satisfaction. Currently, in competitive business environment product development is more dynamic and changing very frequently. The common challenges observed with failed project which has experienced overrun & schedule delays as compared to successful project are related to project planning, estimation, Accepting/Managing changes in requirements and quality management. Today in era of frequently changing technology and requirement, an adaptive approach is much needed for the product development. At present for the most complex and highly unstable projects, the Agile methodologies are implemented to increase the efficiency of the project and to exceed the customers satisfaction. This research will provide an insight on the challenges that management and team face after transitioned to Agile Project Management. Outcome of this research would be useful for project managers who has transitioned to Agile Project

Management but are unaware of the difficulties. It will also help department in the organization who recently transitioned to Agile Project Management and revolutionize the workflow from within. Finally, team will get useful insights on how to handle this change, if they do not have any previous agile project management experience. In addition, it will propose a new conceptual framework i.e., hybrid framework that comprises of Agile and DevSecOps methodologies to enhance the predictability and efficiency of project.

Key words: Project, Traditional Project management, Customer satisfaction, Agile, DevSecOps, Predictability.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I:
INTRODUCTION

1.1 Introduction

In today's fast-paced and competitive business environment, organizations must constantly innovate and adapt to deliver high-quality products and services. Projects serve as structured endeavors aimed at achieving unique outcomes by executing a series of well-planned activities. The success of a project is measured by its ability to meet objectives while adhering to quality standards, timelines, budgets, and customer satisfaction. The choice of an appropriate project management approach required careful selection based on project complexity and requirements. The development approach significantly influences project success and is crucial in determining the efficiency and effectiveness of project execution.

Traditional project management methodologies, such as the Waterfall approach, follow a sequential process where each phase (planning, designing, developing, and testing) is completed before moving to the next. While this approach ensures thorough documentation and control, it lacks flexibility, making it difficult to accommodate changes during the project lifecycle. In contrast, Agile project management has emerged as a dynamic approach that promotes adaptability, continuous improvement, and customer collaboration. Agile enables teams to develop products iteratively, incorporate stakeholder feedback at every stage, and quickly respond to changing requirements making it well-suited for dynamic and evolving projects.

In the aerospace industry, where projects are inherently complex, integrating Agile methodologies presents both opportunities and challenges. Agile’s flexibility is well-suited for the rapidly evolving technological landscape, yet its implementation introduces new

layers of complexity that can impact project predictability. Although Agile enhances responsiveness and innovation, teams often struggle with ensuring predictable project performance, maintaining consistency in deliverables, and managing dependencies effectively.

This research focuses on the challenges organizations encounter after transitioning to Agile project management and explores how to enhance predictability while maintaining Agile's core principles. It proposes a hybrid framework that integrates Agile with DevSecOps methodologies to enhance project predictability, optimize delivery processes, mitigate risks, and ensure consistent value delivery to customers. By addressing these challenges, the study aims to contribute to the advancement of project management practices in complex industries like aerospace.

1.2 Research Problem

Agile project management has gained widespread adoption due to its ability to deliver early and frequent iterations. Unlike traditional methodologies which follow a fixed sequence of planning, designing, developing, and testing, Agile emphasizes flexibility, customer collaboration, and iterative development, allowing organizations to quickly respond to evolving market demands. However, while Agile excels in flexibility, adaptability and responsiveness, it introduces challenges related to project predictability, dependency management, and structured execution.

In the aerospace industry, where precision, compliance, and risk mitigation are critical, transitioning from traditional project management to Agile adds another layer of complexity. Agile's iterative approach does not always align seamlessly with long-term project planning, regulatory requirements, and risk management practices. Organizations adopting Agile often struggle with:

- Uncertainty in project outcomes due to evolving requirements and frequent changes.
- Dependency management across multiple teams working on interconnected deliverables.
- Lack of standardized processes for ensuring predictable project performance.
- Difficulty in balancing flexibility with structured execution in long-term projects.

Despite Agile's benefits in fostering innovation and responsiveness, project teams frequently encounter challenges in achieving predictable and consistent results. There is a need for a structured approach that retains Agile's flexibility, adaptability and responsiveness while improving the predictability of project performance. This research addresses these challenges and aims to bridge the gap by developing a structured approach that combines Agile with DevSecOps practices for enhanced predictability, risk management, and efficiency in project execution.

1.3 Purpose of Research

The purpose of this research is to examine the challenges faced by teams and management after transitioning to Agile project management and to explore strategies for improving project predictability. By identifying key pain points, constraints, dependencies, and risks that impact Agile project performance, the study seeks to propose a structured approach for "Predictive Agile Delivery" that balances flexibility, adaptability and responsiveness with consistency.

The specific objectives of this study are to:

1. Identify key challenges encountered by teams and management after transitioning to Agile.

2. Develop strategies to mitigate the pain points associated with Agile adoption.
3. Analyze the impact of Agile methodologies on project predictability.
4. Highlight the importance of predictable performance in project success.
5. Examine constraints and dependencies that hinder project outcomes.
6. Propose a hybrid framework integrating Agile and DevSecOps for optimized project delivery.

By accomplishing these objectives, the research will contribute to the development of a more structured and predictable Agile project management approach, ensuring that organizations can deliver both high-quality predictable results while maintaining Agile practices for long-term project.

1.4 Significance of the Study

This study holds significant value for project managers, teams, and organizations that have transitioned to Agile methodologies, particularly in industries where precision and predictability are essential. The research findings will be beneficial in the following ways:

- **For Project Managers**: Provides insights into managing Agile projects with improved predictability, reducing uncertainties, and improving decision-making.
- **For Agile Teams**: By providing practical insights helps Agile teams understand and navigate common challenges, ensuring smoother collaboration and better project outcomes.
- **For Organizations**: Offers a structured approach to integrating Agile with DevSecOps, enabling optimized project delivery and enhanced value creation.

- **For the Aerospace Industry**: Addresses specific challenges in aerospace projects, ensuring compliance, risk mitigation, and efficient execution.
- **For Agile Practitioners and Researchers**: Contributes to the academic and professional discourse on Agile project management and its evolution towards greater predictability.

By proposing a hybrid "Predictive Agile Delivery" framework, this study aims to revolutionize Agile project management by making it more predictable, structured, and efficient while retaining its core benefits of adaptability and innovation.

1.5 Research Questions

To address the research problem, this study seeks to answer the following key questions:

1. What are the challenges encountered by teams and management after transitioning to Agile project management?
2. What solutions can mitigate the pain points faced during Agile adoption?
3. How does the implementation of Agile impact project predictability?
4. Why is predictable performance important for any project success?
5. What constraints and dependencies hinder predictable project outcomes?

By addressing these questions, the study will provide actionable insights and strategic recommendations to enhance Agile project management, ensuring both flexibility and predictability in complex and high-risk industries.

1.6 Research Hypothesis

This study aims to explore whether Agile project management affects project predictability and efficiency in aerospace projects. The hypothesis can be stated as follows:

Primary Hypothesis (H_0 - Null Hypothesis)

H_0: Transitioning to Agile Project Management does not significantly impact the predictability of project performance in aerospace projects.

Alternative Hypothesis (H_1 - Alternative Hypothesis)

H_1: Transitioning to Agile Project Management significantly impacts the predictability of project performance in aerospace projects.

This hypothesis will be tested through a combination of literature review, empirical data analysis, and stakeholder feedback from aerospace project teams.

CHAPTER II:

REVIEW OF LITERATURE

2.1 Introduction

Project management methodologies have evolved significantly over the past few decades, with Agile emerging as one of the most widely adopted approaches. Agile methodologies emphasize adaptability, collaboration, and iterative development, enabling organizations to respond effectively to changing requirements and uncertainties in projects (Beck, K., et al. 2001). However, despite its benefits, Agile still faces challenges in achieving predictable project performance, which is essential for organizations to optimize resources, reduce risks, and improve decision-making.

The increasing complexity of projects, especially in the software industry, has led to the exploration of hybrid methodologies that integrate Agile with DevSecOps principles. DevSecOps introduces security as an integral part of software development, fostering continuous integration, automated testing, and real-time risk management (Kim et al., 2016). This research focuses on how a hybrid Agile-DevSecOps framework can enhance project predictability while maintaining flexibility.

This literature review explores key concepts, frameworks, best practices, and challenges related to Agile and DevSecOps. It evaluates existing research on Agile methodologies, discusses the importance of predictable project performance, and investigates how Predictive Agile Delivery can optimize project outcomes. The review aims to identify gaps in current methodologies and propose a structured approach for integrating Agile with DevSecOps to achieve enhanced project delivery.

2.2 Agile Manifesto

The Agile Manifesto was introduced in 2001 by a group of software development practitioners seeking to address the limitations of traditional project management approaches, such as Waterfall (Beck, K., et al. 2001). Traditional approaches were rigid, sequential, and often resulted in delays due to extensive planning and documentation requirements. Agile, in contrast, focused on iterative development, enabling teams to quickly adapt to changes and deliver incremental value.

The Agile Manifesto consists of four Values, and 12 Principles:

The Agile values emphasize a preference for the elements on the left over those on the right as presented in Figure 2.1 Agile Values, rather than a complete disregard for the latter. Processes, tools, comprehensive documentation, contract negotiations, and detailed planning have their place in project execution and organizational workflows. However, Agile prioritizes individuals, interactions, collaboration, working software, and responsiveness to change, as these factors are more directly aligned with delivering value efficiently.

Processes and tools can enhance productivity, but they should not replace direct communication and teamwork, which are essential for addressing complex challenges. Similarly, while documentation provides clarity and reference, excessive reliance on it can slow down progress and reduce flexibility. Agile favors working solutions over exhaustive documentation to ensure continuous delivery of functional and high-quality outcomes.

Contract negotiations and formal agreements are necessary for defining project scopes and responsibilities, but rigid contractual obligations can hinder adaptability in dynamic environments. Agile methodologies advocate for close collaboration and flexibility in decision-making to accommodate evolving requirements. Likewise, while

planning is an integral part of any project, Agile values responding to change over following a fixed plan, recognizing that the ability to adapt is crucial in addressing uncertainties and emerging opportunities.

By emphasizing the left side of the Agile values while still acknowledging the importance of the right, Agile promotes a balanced approach that enables efficiency, innovation, and responsiveness to stakeholder needs.

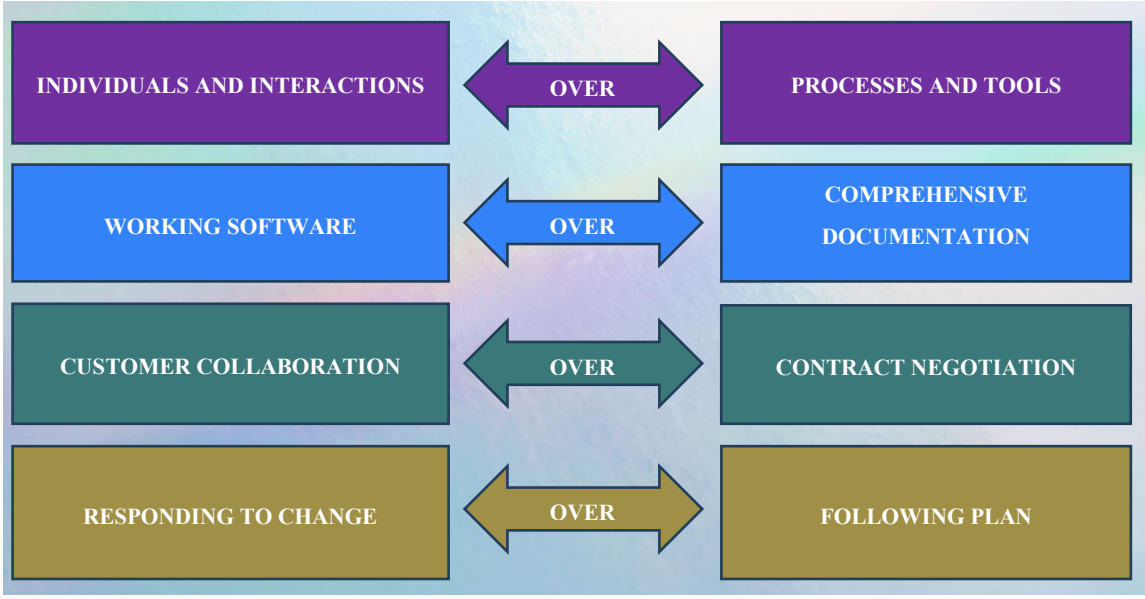


Figure 2.1
Agile Values

Agile principles are a set of 12 guidelines that emphasizes flexibility, collaboration, and customer-centric development in software and project management. Rooted in the Agile Manifesto, they emphasize individuals and interactions over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to change over following a plan. Agile principles as presented in Figure 2.2 Agile Principles, promote adaptability, teamwork, and continuous improvement to ensure efficient project execution. The approach prioritizes customer

satisfaction by delivering valuable solutions early and frequently, allowing for continuous feedback and refinement. It embraces changing requirements at any stage of development, ensuring flexibility to meet evolving needs. Regular delivery of functional increments keeps progress steady, and stakeholders engaged. Strong collaboration between business stakeholders and development teams fosters alignment and shared understanding. Agile empowers motivated individuals, encouraging accountability and innovation. Clear and direct communication, preferably face-to-face, enhances efficiency and quick decision-making. Success is measured by the delivery of working solutions rather than excessive documentation, keeping the focus on functionality. Sustainable development is encouraged by maintaining a consistent work pace, preventing burnout, and ensuring long-term productivity. A commitment to technical excellence and well-structured design enhances quality and efficiency. Simplicity is valued, with teams focusing only on essential tasks to maximize productivity. Self-organizing teams are trusted to make decisions, leading to better problem-solving and innovation. Continuous reflection and process adjustments enable teams to improve their performance over time. These principles collectively help organizations become more flexible, customer-focused, and efficient, driving successful project outcomes.

Figure 2.2
Agile Principles

2.3 Agile Frameworks and Practices

2.3.1 Introduction

Agile methodologies have revolutionized software and product development, offering flexibility, adaptability, and iterative improvements (Beck, K., et al. 2001). Various Agile frameworks exist, each catering to different team sizes, industries, and project complexities. This section explores Agile frameworks and best practices, analyzing their evolution, strengths, challenges, and applications in real-world scenarios.

2.3.2 Agile Frameworks and Practices

Agile frameworks provide structured methodologies for iterative and incremental development. This section presents a comparative review of major Agile frameworks.

2.3.2.1 Scrum

Scrum is an Agile framework that enables teams to develop products through an iterative and incremental approach. It was introduced by Jeff Sutherland and Ken Schwaber in the 1990s and has since become one of the most widely used Agile methodologies across industries (Schwaber, K., & Sutherland, J. 2017). Scrum is based on empirical process control, meaning that it relies on real-time observations, continuous feedback, and adaptation instead of rigid, upfront planning. The framework is built on three key principles: transparency, inspection, and adaptation, which help teams respond effectively to changing requirements.

Scrum consists of three primary roles: Product Owner, Scrum Master, and Development Team. The Product Owner is responsible for defining and prioritizing the Product Backlog, ensuring that the team focuses on the most valuable features. The Scrum Master facilitates the process by coaching the team on Scrum principles and removing impediments, while the Development Team collaborates to deliver high-quality increments of the product. Work is organized into Sprints, which are time-boxed iterations lasting one to four weeks. Each Sprint begins with Sprint Planning, where the team selects backlog items to work on. Daily Scrum (Stand-up) meetings allow for progress tracking and issue resolution. At the end of the Sprint, the team conducts a Sprint Review to showcase completed work and a Sprint Retrospective to reflect on process improvements.

Scrum is guided by five core values: Commitment, Courage, Focus, Openness, and Respect, which help create a culture of trust and collaboration. While Scrum works well

for small, cross-functional teams, scaling Scrum in large organizations often requires frameworks like Scrum@Scale, Nexus, and the Scaled Agile Framework (SAFe). However, implementing Scrum can present challenges, such as organizational resistance to change, difficulty in adopting an Agile mindset, and the need for strong collaboration and self-management. Despite these challenges, Scrum remains a powerful framework that enhances productivity, fosters continuous improvement, and enables teams to respond efficiently to evolving project requirements (Schwaber, K., & Sutherland, J. 2017).

2.3.2.2 Kanban

Kanban is a flow-based Agile framework designed to enhance workflow efficiency and continuous delivery by visualizing tasks, limiting work in progress (WIP), and improving process flow. Originally developed in Toyota's production system, Kanban has been widely adopted in software development and business operations to optimize resource utilization and minimize bottlenecks. The framework operates on a pull-based approach, ensuring that new tasks are started only when the team has available capacity, which helps maintain a steady flow of work and reduces lead times (Anderson, D. J. 2010). Kanban is guided by core principles, including visualizing the workflow, managing WIP limits, making process policies explicit, monitoring flow efficiency, and fostering continuous improvement. Unlike Scrum, which follows time-boxed iterations, Kanban provides flexibility, allowing teams to adapt to changing priorities without rigid sprint structures (Leopold, K., & Kaltenecker, S. 2015). This adaptability makes it particularly beneficial for teams with unpredictable workloads, such as IT support, operations, and continuous delivery environments. By consistently analyzing work patterns and refining processes, Kanban enables organizations to enhance productivity, deliver value efficiently, and respond to changing business needs effectively (Kniberg, H., & Skarin, M. 2010).

2.3.2.3 Extreme Programming (XP)

Extreme Programming (XP) is an Agile software development methodology that focuses on adaptability, collaboration, and high-quality software delivery through iterative development and continuous feedback. Introduced by Kent Beck in the late 1990s, XP was designed to address the challenges of rapidly changing requirements while ensuring software reliability and efficiency (Beck, K. 1999). The methodology is structured around five key values: communication, simplicity, feedback, courage, and respect. These values form the foundation of XP practices, which include test-driven development (TDD), pair programming, continuous integration, small releases, and collective code ownership (Beck, K., & Andres, C. 2004).

One of the distinguishing aspects of XP is its emphasis on close customer collaboration. Unlike traditional development models that define requirements upfront, XP involves the customer throughout the development process. Requirements are captured in the form of user stories, which guide development in short iterations, allowing teams to adapt to changing business needs dynamically (Martin, R. C. 2003). Another core principle is collective code ownership, where all developers share responsibility for the codebase, ensuring consistency and maintainability.

XP incorporates engineering best practices to improve software quality and minimize defects. Test-driven development (TDD) is a fundamental practice where tests are written before the actual code, ensuring robust functionality and reducing bugs. Pair programming, in which two developers work together on the same task, enhances code quality, fosters knowledge sharing, and reduces errors. Continuous integration enables teams to merge code changes frequently, allowing early detection of defects and maintaining a stable software build (Beck, K., & Andres, C. 2004).

Unlike other Agile frameworks, XP is particularly suited for projects with high uncertainty and rapidly changing requirements. By emphasizing continuous improvement, rapid iterations, and customer feedback, XP enables teams to develop flexible, high-quality software solutions efficiently. Its focus on adaptability and engineering excellence makes it a valuable methodology for teams aiming to balance speed, collaboration, and software reliability (Fowler, M. 2001).

2.3.2.4 Scaled Agile

Agile methodologies have gained widespread adoption in software development due to their iterative nature, adaptability, and focus on customer collaboration. However, as organizations grow, applying Agile principles at scale presents challenges that require structured approaches to coordination, alignment, and governance. This section reviews the existing literature on Scaled Agile, exploring its evolution, theoretical foundations, key frameworks, benefits, and challenges.

Evolution of Scaled Agile

The concept of Agile scaling emerged as organizations sought to extend Agile methodologies beyond single teams to enterprise-wide adoption. Initially, Agile frameworks such as Scrum and Extreme Programming (XP) were designed for small, co-located teams working on relatively independent tasks (Beck, K., & Andres, C. 2004). However, as Agile principles were applied to larger, more complex projects, organizations encountered difficulties in coordinating multiple teams, ensuring cross-functional collaboration, and maintaining strategic alignment (Leffingwell, D. 2019).

The first attempts to scale Agile were largely ad hoc, with organizations experimenting with variations of Scrum and Kanban to manage large teams. Over time, structured frameworks like Scaled Agile Framework (SAFe), Large-Scale Scrum (LeSS),

and Disciplined Agile Delivery (DAD) were developed to provide formalized guidance for scaling Agile in complex environments (Larman, C., & Vodde, B. 2010). These frameworks integrate Agile with Lean principles, DevOps, and governance structures to optimize efficiency and business agility.

Theoretical Foundations of Scaled Agile

Scaled Agile draws from several theoretical foundations, including Lean Thinking, Systems Thinking, and Agile Manifesto principles.

1. **Lean Thinking –** Originally developed for manufacturing, Lean principles emphasize value stream optimization, waste reduction, and continuous improvement. Agile scaling frameworks adopt Lean concepts such as value stream mapping, flow efficiency, and continuous delivery to enhance large-scale software development (Knaster, R., & Leffingwell, D. 2020).

2. **Systems Thinking –** Large organizations operate as complex systems with interdependent teams and processes. Scaled Agile frameworks encourage a holistic view of software delivery, ensuring that dependencies, risks, and cross-team interactions are effectively managed (Larman, C., & Vodde, B. 2010).

3. **Agile Manifesto and Agile Principles –** The core Agile values—individuals and interactions, working software, customer collaboration, and responsiveness to change—remain foundational in scaled Agile approaches. However, frameworks like SAFe and LeSS introduce structured governance models and planning mechanisms to align Agile with enterprise needs (Leffingwell, D. 2019).

Key Scaled Agile Frameworks

Multiple frameworks exist to support Agile adoption at scale, each addressing specific organizational needs. The three most widely recognized frameworks are Scaled Agile Framework (SAFe), Large-Scale Scrum (LeSS), and Disciplined Agile Delivery (DAD).

1. **Scaled Agile Framework (SAFe) –** SAFe is the most widely adopted Agile scaling framework, designed to provide a structured approach to large-scale Agile implementation. It organizes Agile adoption into four levels: Team, Program, Large Solution, and Portfolio. SAFe incorporates Lean principles, Agile Release Trains (ARTs), and Program Increment (PI) Planning to synchronize multiple teams and align Agile initiatives with business objectives (Leffingwell, D. 2019).

2. **Large-Scale Scrum (LeSS) –** LeSS is an extension of Scrum designed for scaling Agile across multiple teams while maintaining simplicity and minimal additional roles. It emphasizes empirical process control, continuous feedback, and decentralization to enable large teams to function effectively without excessive hierarchy (Larman, C., & Vodde, B. 2010).

3. **Disciplined Agile Delivery (DAD) –** DAD is a process-decision framework that allows organizations to tailor Agile scaling approaches based on their unique context. Unlike SAFe and LeSS, DAD integrates Agile, Lean, and DevOps practices while providing flexibility in selecting Agile scaling strategies (Ambler & Lines, 2012).

Benefits of Scaled Agile

Research on Agile scaling highlights several benefits of adopting structured frameworks for large-scale Agile transformation:

- Enhanced Coordination and Alignment – Frameworks like SAFe introduce structured mechanisms such as Agile Release Trains (ARTs) and Lean Portfolio Management (LPM) to align Agile teams with strategic business goals (Knaster & Leffingwell, 2020).

- Improved Time-to-Market – Scaled Agile methodologies emphasize continuous integration, automated testing, and DevOps practices, enabling organizations to deliver high-quality software at a faster pace (Leffingwell, 2019).

- Greater Transparency and Collaboration – Scaled Agile promotes cross-team synchronization, shared backlogs, and regular system demos, fostering a culture of openness and continuous feedback (Larman, C., & Vodde, B. 2010).

- Enterprise Agility and Innovation – By integrating Agile with Lean and DevOps, scaled Agile frameworks help enterprises respond swiftly to market changes, reduce waste, and foster innovation (Ambler & Lines, 2012).

Challenges in Implementing Scaled Agile, despite its advantages, literature highlights significant challenges in implementing and sustaining scaled Agile:

- Cultural Resistance and Change Management – Many organizations struggle with shifting from traditional hierarchical management to a Lean-Agile leadership

model that empowers teams and decentralizes decision-making (Ambler & Lines, 2012).

- Complexity in Large-Scale Coordination – Managing dependencies across multiple Agile teams, integrating technical architectures, and aligning business priorities require structured frameworks like SAFe and LeSS (Knaster & Leffingwell, 2020).

- Balancing Governance and Flexibility – While Agile promotes autonomy and flexibility, large enterprises require governance structures to manage compliance, security, and financial controls (Leffingwell, 2019).

Technical Challenges in Automation and DevOps – Scaled Agile frameworks depend on high levels of automation, test-driven development (TDD), and continuous integration (CI) to ensure efficient software delivery. However, legacy systems and infrastructure limitations often pose significant hurdles (Larman, C., & Vodde, B. 2010).

2.4 Scaled Agile Frameworks (SAFe) Theory, Practices and Challenges

The Scaled Agile Framework (SAFe) is a comprehensive approach designed to scale Agile principles across large enterprises. It integrates Lean, Agile, and DevOps methodologies to enhance business agility, optimize value delivery, and align teams towards common objectives (Leffingwell et al., 2021). SAFe provides structured guidance for organizations transitioning from traditional project management approaches to Agile methodologies. While it offers numerous benefits, including improved collaboration, faster time-to-market, and enhanced transparency, its implementation comes with challenges,

such as predictability issues, cultural resistance, governance complexities, and dependency management.

Theoretical Foundations of SAFe

SAFe is built on several key theoretical principles, including Lean Thinking, Agile principles, Systems Thinking, and DevOps. Lean Thinking, as introduced by Womack and Jones (2003), focuses on maximizing customer value while minimizing waste. SAFe applies Lean principles through value stream mapping, continuous improvement, and flow optimization. The Agile Manifesto, established by Beck et al. (2001), emphasizes customer collaboration, adaptive planning, and iterative development. SAFe extends these Agile principles beyond small teams, ensuring that large enterprises maintain agility across multiple teams and departments.

Systems Thinking, as described by Senge (1990), encourages a holistic approach to problem-solving. SAFe integrates Systems Thinking by organizing teams around value streams rather than isolated functions, ensuring that every process contributes to business outcomes. Additionally, SAFe incorporates DevOps practices to enhance automation, collaboration, and continuous delivery. DevOps principles, such as Continuous Integration (CI) and Continuous Deployment (CD), allow organizations to deliver high-quality software more efficiently (Kim et al., 2016). By integrating these theoretical foundations, SAFe provides a structured yet flexible framework for Agile transformation at scale.

Key Practices in SAFe

SAFe structures Agile teams and workflows to optimize collaboration, value delivery, and continuous improvement. One of its core elements is Value Stream Organization, which shifts the focus from individual team outputs to end-to-end value

delivery. Instead of traditional functional hierarchies, SAFe encourages aligning teams around value streams to enhance agility and efficiency (Scaled Agile, Inc., 2023).

The Agile Release Train (ART) is another fundamental practice in SAFe. ARTs consist of multiple Agile teams (typically 5–12) that collaborate to deliver value through Program Increments (PIs), which last 8–12 weeks. Each ART follows a synchronized cadence to ensure coordinated development, reducing misalignment and bottlenecks. PI Planning is a structured event where all ART members define objectives, identify dependencies, and commit to deliverables for the upcoming increment (Leffingwell et al., 2021). This practice ensures alignment across teams and enhances predictability.

SAFe also incorporates Lean Portfolio Management (LPM) to align strategic goals with Agile execution. LPM ensures that funding, prioritization, and execution are aligned with business objectives. Key elements of LPM include lean budgeting, portfolio Kanban, and strategic roadmap planning. Furthermore, SAFe integrates DevOps through a Continuous Delivery Pipeline, consisting of Continuous Exploration (CE), Continuous Integration (CI), Continuous Deployment (CD), and Release on Demand. This approach enhances automation, reduces deployment risks, and accelerates feature delivery.

Another key practice in SAFe is the **Inspect and Adapt (I&A) Workshop**, which occurs at the end of each PI. This retrospective process allows teams to review performance, identify improvement areas, and adjust processes accordingly. It reinforces the principle of continuous learning and optimization. These structured yet adaptable practices make SAFe a powerful framework for large-scale Agile implementation.

Challenges in SAFe Implementation

Despite its advantages, SAFe implementation presents several challenges, particularly in complex organizations. One of the most significant challenges is

predictability. While SAFe aims to improve predictability through structured planning, real-world execution often deviates from forecasts due to estimation inaccuracies, cross-team dependencies, and evolving priorities. Teams may struggle with estimation accuracy, leading to unreliable PI commitments (Scaled Agile, Inc., 2023). Additionally, dependencies between ARTs can create bottlenecks, further reducing predictability.

Cultural resistance to change is another common challenge. Many organizations transitioning to SAFe face opposition from employees accustomed to traditional project management methodologies. Agile requires decentralized decision-making and iterative planning, which can be difficult for teams that have historically followed hierarchical structures (Leffingwell et al., 2021). Effective change management strategies, including executive sponsorship and Agile training programs, are essential for overcoming cultural resistance.

Complexity in large organizations also poses a challenge. SAFe introduces multiple layers of roles, events, and governance structures, which can be overwhelming for organizations new to Agile scaling. If not implemented correctly, SAFe can create excessive bureaucracy, reducing the flexibility it aims to achieve. Organizations must tailor SAFe to their specific needs rather than applying it as a rigid framework.

Another major challenge is **alignment between business and IT**. While SAFe promotes business-IT collaboration, achieving seamless alignment requires continuous engagement. Business stakeholders may not fully participate in Agile processes, leading to misaligned priorities and unexpected scope changes (Kim et al., 2016). Implementing Lean Portfolio Management (LPM) and Weighted Shortest Job First (WSJF) prioritization techniques can help bridge this gap.

Dependency management and coordination issues further complicate SAFe adoption. Large-scale Agile teams often struggle with interdependencies that slow down

development cycles. Without proactive dependency tracking, teams may face delays and integration issues. Tools like Jira Advanced Roadmaps and cross-team synchronization meetings can help address these challenges.

Finally, **governance and compliance** are significant concerns, especially in regulated industries such as aerospace, finance, and healthcare. SAFe's flexibility can sometimes conflict with stringent regulatory requirements. To address this, organizations must incorporate compliance checkpoints into SAFe workflows, ensuring that Agile practices align with industry regulations without compromising agility.

2.5 Roles and Responsibilities in SAFe at ART Level

The Agile Release Train (ART) is a fundamental component of the Scaled Agile Framework (SAFe), consisting of multiple Agile teams working collaboratively to deliver value in a synchronized manner. The ART operates within Program Increments (PIs), ensuring alignment between business and technical stakeholders to optimize value delivery (Scaled Agile, Inc., 2023). Several key roles within the ART contribute to its success, each with distinct responsibilities.

The **Release Train Engineer (RTE)** acts as the servant leader and primary facilitator of the ART. The RTE is responsible for ensuring smooth ART execution by coordinating Program Increment (PI) planning, managing dependencies, removing impediments, and driving relentless improvement through Inspect and Adapt (I&A) workshops. Additionally, the RTE collaborates with Product Management, System Architects, and Business Owners to enhance ART performance (Scaled Agile, Inc., 2023).

Product Management plays a critical role in defining and maintaining the Program Backlog, ensuring that features align with business objectives and customer needs. This

role involves engaging with customers and stakeholders to prioritize backlog items based on value and feasibility. Product Management also works closely with Product Owners to maintain alignment between ART-level strategy and team execution (Leffingwell, 2021).

The **System Architect/Engineer** is responsible for establishing and communicating the architectural vision for the ART. This role ensures that technical decisions align with enterprise architecture while balancing business needs. System Architects guide Agile teams in technical decision-making, oversee the implementation of enablers, and support cross-team coordination for scalable and resilient solutions (Scaled Agile, Inc., 2023).

Business Owners represent key stakeholders who provide strategic direction and ensure that the ART delivers measurable business value. Their responsibilities include participating in PI planning, offering guidance on prioritization and funding, and evaluating progress through System Demos and Inspect and Adapt sessions. Business Owners play a crucial role in ensuring ART alignment with broader enterprise goals (Leffingwell, 2021).

Agile Teams within the ART are responsible for delivering features and enablers iteratively throughout the PI. Teams collaborate closely with Product Owners to refine stories, maintain a sustainable pace, and continuously improve through retrospectives. Agile teams participate in key SAFe ceremonies, such as PI Planning, System Demos, and ART Sync meetings, ensuring alignment and transparency across the ART (Scaled Agile, Inc., 2023).

The **Product Owner (PO)** serves as the primary liaison between Agile Teams and Product Management. The PO owns the Team Backlog, ensuring that work is well-defined, prioritized, and ready for development. By collaborating with Agile Teams, the PO ensures

that customer and business needs are met while maintaining alignment with the overall ART objectives (Leffingwell, 2021).

The **Scrum Master/Team Coach** facilitates Agile ceremonies, supports team collaboration, and removes impediments that hinder progress. Scrum Masters play a key role in fostering a culture of continuous improvement, coaching teams on Agile best practices, and ensuring alignment with ART-level objectives. They also collaborate with the RTE to track ART execution and identify areas for process enhancements (Scaled Agile, Inc., 2023).

A **RACI matrix** (Responsible, Accountable, Consulted, Informed) helps clarify the roles of different stakeholders within the ART and their level of involvement in key activities. In SAFe, the RTE is primarily accountable for ART execution, ensuring synchronization among Agile Teams, Product Management, and System Architects. Product Management is responsible for prioritizing the Program Backlog, defining features, and engaging with Business Owners to align ART efforts with enterprise goals. The System Architect provides architectural guidance, ensuring technical feasibility and alignment with long-term enterprise strategy. Business Owners act as key decision-makers, consulting with Product Management and RTEs while being accountable for business value delivery. Agile Teams and Product Owners work closely together to execute the backlog, with Scrum Masters facilitating team-level collaboration and continuous improvement (Scaled Agile, Inc., 2023).

Overall, clear role definitions and accountability mechanisms within the ART enable effective execution of SAFe practices. By maintaining a structured approach to

ART-level responsibilities, organizations can enhance agility, foster alignment, and optimize value delivery across the enterprise. Table 2.1 provide example of RACI Matrix.

Legend:

- **R (Responsible):** The role that executes the activity.
- **A (Accountable):** The role ultimately answerable for the completion of the activity.
- **C (Consulted):** The role providing input or expertise before the activity is performed.
- **I (Informed):** The role receiving updates on the activity's progress or outcome.

This RACI matrix provides a structured approach to role accountability within the ART, ensuring clear ownership and efficient execution of SAFe practices (Scaled Agile, Inc., 2023).

Table 2.1
RACI Matrix -Provide structured approach to Role Accountability

Activity/Responsibility	*Release Train Engineer (RTE)*	*Product Management*	*System Architect/ Engineer*	*Business Owners*	*Agile Teams*	*Product Owner*	*Scrum Master*
PI Planning Facilitation	*A*	*C*	*C*	*C*	*I*	*R*	*R*
Program Backlog Prioritization	*C*	*A*	*C*	*R*	*I*	*C*	*I*
Feature Definition	*C*	*A*	*C*	*C*	*I*	*R*	*I*
System Architecture Guidance	*C*	*C*	*A*	*C*	*I*	*I*	*I*
ART Execution Tracking	*A*	*C*	*I*	*I*	*R*	*C*	*R*

System Demo Coordination	*A*	*R*	*C*	*I*	*R*	*C*	*R*
Inspect & Adapt Workshops	*A*	*C*	*C*	*C*	*R*	*C*	*R*
Continuous Improvement	*R*	*C*	*C*	*C*	*R*	*C*	*A*

2.6 Common Practices in Agile Project

Effort Estimation in Agile Projects

Agile estimation is a critical practice that helps teams forecast the effort required to complete work items while maintaining flexibility. One of the most widely used techniques is Planning Poker, where team members individually estimate effort using story points and discuss discrepancies to reach a consensus (Scrum Institute, n.d.). Another method, T-shirt sizing, categorizes tasks into broad effort-based groups (XS, S, M, L, XL), making it useful for high-level estimations (Net Solutions, 2022). The Fibonacci sequence (1, 2, 3, 5, 8, etc.) is another common technique that accounts for uncertainty by increasing the gap between successive numbers. Teams may also use affinity mapping, where similar-sized tasks are grouped to streamline estimation, and bucket system estimation, which organizes tasks into predefined effort categories.

Estimation is often done during backlog refinement or sprint planning meetings to ensure tasks are appropriately sized before development begins. Continuous refinement of estimates as more information becomes available ensures accuracy in planning and execution.

Product Backlog Grooming (Refinement) and Optimization

Backlog grooming (also called backlog refinement) is an ongoing process that ensures the backlog remains well-structured and prioritized for upcoming sprints. The product owner and development team collaborate to clarify user stories, update

requirements, and break down large tasks into smaller, more manageable ones (Scrum Institute, n.d.). Techniques such as user story splitting help refine complex backlog items by breaking them into smaller components based on workflow steps, data variations, or business rules.

Tools like JIRA, Trello, or Azure DevOps facilitate backlog refinement by providing visual workflows and tracking progress. Story mapping is another effective technique that structures backlog items based on user journey and business value, ensuring a logical development sequence. Grooming sessions typically occur once per sprint, enabling teams to maintain a backlog that is well-defined, estimated, and prioritized.

Backlog Prioritization for Maximum Value Delivery

Effective prioritization ensures that Agile teams focus on delivering the most valuable features first. Several techniques guide prioritization:

- MoSCoW Method (Must-have, Should-have, Could-have, Won't-have) categorizes backlog items based on criticality and business necessity (Easy Agile, 2023).
- Weighted Shortest Job First (WSJF) ranks backlog items by dividing the cost of delay by the estimated effort, helping teams optimize their development sequence.
- Kano Model classifies features into basic needs, performance needs, and delight factors, ensuring teams prioritize features that maximize user satisfaction.
- Impact-Effort Matrix visualizes tasks based on their expected impact and required effort, enabling teams to balance high-value, low-effort tasks for quick wins.

- RICE Scoring (Reach, Impact, Confidence, Effort) assigns scores to backlog items based on their potential reach, impact, confidence level, and effort required.

Backlog prioritization is a continuous activity that evolves based on stakeholder input, market dynamics, and technical constraints. Agile frameworks like Scrum prioritize based on business value, while Kanban focuses on continuous flow with a just-in-time prioritization approach.

Risk and Dependency Management in Agile

Agile methodologies emphasize proactive risk identification and dependency management to ensure smooth project execution. Risks can stem from technical uncertainties, resource constraints, external dependencies, or changing requirements. Techniques such as risk-based backlog ordering help teams address high-risk items early in development.

- Risk-adjusted backlog prioritization integrates risk assessment into backlog ranking, ensuring high-risk tasks are tackled early.
- Dependency mapping visually represents relationships between tasks, ensuring teams account for interdependencies when planning sprints.
- Cross-team coordination tools such as SAFe's Program Increment (PI) Planning and Scrum of Scrums facilitate dependency resolution in scaled Agile environments.
- Failure Mode and Effects Analysis (FMEA) systematically evaluates potential failure points in a system, enabling teams to mitigate risks before they escalate.

Regular risk review sessions and impediment boards help teams track and resolve risks and dependencies dynamically throughout the project lifecycle (Rapidr, 2023).

Retrospective Meetings for Continuous Improvement

Sprint retrospectives are a cornerstone of Agile's commitment to continuous improvement. These meetings, held at the end of each sprint, allow teams to reflect on their performance and identify actionable improvements. Common retrospective formats include:

- Start-Stop-Continue: Teams list practices they should start, stop, or continue in future sprints.
- 4Ls (Liked, Learned, Lacked, Longed For): Encourages a balanced evaluation of successes and areas for growth.
- Mad-Sad-Glad: Helps teams understand emotional responses to the sprint, addressing frustrations constructively.
- Sailboat Retrospective: Uses a visual metaphor where the sail represents what propels the team forward, while anchors represent obstacles.

The key outcome of retrospectives is a commitment to small, incremental process improvements that enhance efficiency and team dynamics (Easy Agile, 2023). Agile promotes continuous improvement through inspection and adaptation (Schwaber & Sutherland, 2017), refer Table 2.2 for common Inspection and reflection techniques in Agile.. Teams regularly assess their processes, identify gaps, and refine workflows. These activities ensure better efficiency, and quality in Agile projects.

Table 2.2
Common Inspection and Reflection Techniques in Agile

Technique	Description	Purpose
Sprint Retrospective	Teams review what went well, what didn't, and what to improve.	Continuous team improvement.
Sprint Review	Showcases completed work to stakeholders for feedback.	Product validation and feedback loop.
Daily Standups	Team members discuss progress, blockers, and next steps.	Identifies issues early.
Root Cause Analysis (RCA)	Investigates the cause of recurring issues.	Prevents defects from reoccurring.
5 Whys	Asking "why" multiple times to find the root cause of a problem.	Deep problem analysis.
Kaizen Approach	Continuous incremental improvements in Agile practices.	Long-term process optimization.

2.7 Comprehensive Comparison of Agile Frameworks

Comparison of Agile frameworks across multiple dimensions, covering roles, governance, scalability, process structure, best practices, and more, as described in Table 2.3, Table 2.4, Table 2.5, Table 2.6, Table 2.7, Table 2.8, and Table 2.9.

Table 2.3
High-Level Overview Comparison

Framework	Focus Area	Best For	Primary Goal	Governance Level
Scrum	Iterative development, team collaboration	Small to mid-size teams	Delivering working software in short cycles	Team-Level
Kanban	Flow efficiency,	Support/operations teams	Optimizing workflow &	Team & Operational

	continuous delivery		minimizing cycle time	
Extreme Programming (XP)	Engineering excellence	Engineering-intensive teams	Improving code quality & rapid feedback loops	Team-Level
SAFe (Scaled Agile Framework)	Large-scale Agile transformation	Enterprises with multiple Agile teams	Aligning Agile at team, program, and portfolio levels	Team, Program, Portfolio, Enterprise
LeSS (Large-Scale Scrum)	Scrum scaling for large teams	Organizations wanting pure Scrum at scale	Scaling Scrum while minimizing complexity	Team & Program
Disciplined Agile (DA)	Hybrid Agile & enterprise agility	Enterprises with diverse teams	Tailoring Agile based on context-specific needs	Team, Program, Portfolio

Table 2.4
Role and Responsibilities Comparison

Framework	Key Roles	Governance	Decision-Making
Scrum	Product Owner (PO), Scrum Master (SM), Development Team	Decentralized	Team-level decisions; PO manages backlog
Kanban	Service Request Manager, Service Delivery Manager, Team Members	Minimal governance	Continuous improvement through visual boards
XP	Customer, Programmer, Coach, Tracker, Tester	Decentralized	Direct developer-customer interaction
SAFe	Development Team, Scrum Master, Product Owner, Release Train Engineer (RTE), LPM, Architects, Business Owners	Hierarchical (Team → Program → Portfolio → Enterprise)	Strategic decisions made at enterprise level, tactical at team level

LeSS	PO, SM, Cross-functional teams	Decentralized but aligned	Shared backlog across teams, fewer managerial layers
DA	Agile Coach, Enterprise Architect, Team Leads	Context-driven governance	Decision-making tailored per team/process

Table 2.5
Scalability & Enterprise Adoption Comparison

Framework	Scalability	Best For Large Enterprises?	Coordination Between Teams
Scrum	Limited to a few teams	No	Each team works independently
Kanban	Can scale, but not structured for large enterprises	No	Workflows managed independently
XP	Team-Level only	No	Focuses on code quality, not enterprise-wide planning
SAFe	Highly scalable (team, program, portfolio, enterprise levels)	Yes	Coordinated Agile Release Trains (ARTs)
LeSS	Scales Scrum for multiple teams	Yes	One PO for all teams, fewer layers than SAFe
DA	Fully customizable enterprise framework	Yes	Governance adapts to organizational needs

Table 2.6
Process Structure & Workflow Comparison

Framework	Process Type	Iterations	Work Management
Scrum	Iterative & Incremental	Sprints (1-4 weeks)	Sprint Backlog, Product Backlog
Kanban	Continuous flow	No iterations	WIP limits, visualized workflow
XP	Iterative, fast feedback loops	1–2-week iterations	Continuous integration & automated testing
SAFe	Hybrid (Iteration + Planning at multiple levels)	Program Increment (PI) – 8-12 weeks	Agile Release Trains (ARTs), Kanban at Portfolio level
LeSS	Iterative Scrum	Sprints	One backlog, multiple Scrum teams
DA	Adaptive (mix of Scrum, Kanban, Lean, etc.)	Variable	Governance based on organization's needs

Table 2.7
Engineering & Technical Practices Comparison

Framework	Technical Focus	Code Quality & Automation	CI/CD Support
Scrum	Medium	Not enforced, but encouraged	Supports, but not a core part
Kanban	Low	Focuses on process, not code	No specific CI/CD practices
XP	Very High	TDD, Pair Programming, Refactoring	Strong CI/CD integration
SAFe	High	Built-in quality & DevOps integration	Supports DevSecOps

LeSS	Medium	Encourages XP practices	Recommended but not enforced
DA	Medium-High	Varies per implementation	Supports CI/CD, but not mandatory

Table 2.8
Metrics & Performance Tracking Comparison

Framework	Primary Metrics Used	Business Impact Focus
Scrum	Velocity, Sprint Burndown, Lead Time	Customer value delivery per Sprint
Kanban	Cycle Time, Throughput, WIP Limits	Flow efficiency & continuous delivery
XP	Defect Rate, Test Coverage, Pair Programming Hours	Code quality & developer productivity
SAFe	Business Agility, ART Velocity, PI Predictability	Enterprise-level agility & alignment
LeSS	Sprint Performance, Team Collaboration	Cross-team productivity
DA	Depends on organization's setup	Context-driven business impact

Table 2.9
Adoption Challenges Comparison

Framework	Adoption Barriers	Common Failure Points
Scrum	Requires discipline & strong PO	Lack of leadership buy-in
Kanban	Hard to scale beyond single teams	WIP limits misunderstood
XP	Demands high engineering maturity	Resistance to Pair Programming & TDD
SAFe	Complex implementation, requires training	Overhead & rigid structure
LeSS	Needs strong Scrum alignment	Difficult for traditional organizations
DA	Requires customization per team	Over-complication due to too many options

2.8 Challenges Faced by Teams After Transitioning to Agile Project Management

Transitioning to Agile project management offers numerous benefits, including enhanced flexibility, faster delivery, and improved customer satisfaction. However, this shift also presents several challenges that teams must navigate to ensure successful adoption and sustained performance. One of the most significant barriers to Agile adoption is cultural resistance and organizational inertia. Agile methodologies emphasize self-organizing teams, decentralized decision-making, and iterative development cycles, which often conflict with traditional hierarchical structures. Studies indicate that cultural clashes remain a prominent challenge, with 47% of organizations citing resistance to Agile

implementation as a major impediment (Digital.ai, 2023). This resistance frequently stems from a lack of understanding and a reluctance to deviate from established processes.

Another major issue is inconsistencies in processes and practices across teams. When organizations attempt to blend Agile with existing methodologies without a clear strategy, it leads to fragmentation and inefficiencies. Around 46% of organizations report challenges in maintaining consistent Agile practices across teams, which hinders collaboration and slows down project progress (Digital.ai, 2023). This inconsistency is often due to inadequate training and a lack of standardized procedures, which further complicates Agile adoption. Additionally, a broader organizational resistance to change contributes to the struggle, with 42% of organizations experiencing reluctance to adopt new tools, processes, or roles associated with Agile frameworks (Digital.ai, 2023). Employees and management alike may resist change due to uncertainty, comfort with existing workflows, or skepticism about Agile's long-term benefits.

The lack of necessary skills and experience is another critical factor impeding Agile adoption. Agile methodologies require teams to be proficient in new ways of working, including Scrum, Kanban, and Lean principles. However, 42% of organizations identify a lack of Agile expertise as a significant barrier (Digital.ai, 2023). Without adequate training, teams may struggle to correctly implement Agile practices, leading to reduced efficiency and frustration. Moreover, the absence of leadership participation weakens Agile initiatives. Leadership plays a crucial role in guiding Agile transformations, yet 41% of organizations report a lack of leadership involvement in Agile adoption (Digital.ai, 2023). When leadership does not actively support the transition, teams may face unclear objectives, insufficient resources, and a lack of authority to drive necessary changes.

Closely related to leadership participation is inadequate management support and sponsorship, which is reported as a challenge by 40% of organizations (Digital.ai, 2023).

Without strong management backing, Agile teams often struggle to secure the resources and organizational buy-in needed to sustain Agile adoption. Another issue is the misunderstanding of Agile processes, where teams misinterpret Agile principles, leading to flawed implementations. Some organizations mistakenly believe that Agile means less planning or structure, which results in chaotic workflows and unmet project objectives (Scrum Alliance, n.d.). Addressing these misconceptions requires educational initiatives and clear communication to align team members' understanding of Agile methodologies.

As organizations attempt to scale Agile beyond small teams, they encounter challenges in scaling Agile effectively. Coordinating multiple teams, maintaining consistent Agile practices, and aligning Agile methods with organizational goals become complex as Agile adoption expands. Frameworks like SAFe (Scaled Agile Framework) offer solutions, but selecting and implementing the right scaling strategy remains a major hurdle for many organizations (Bick et al., 2018). Additionally, stakeholder engagement and expectation management pose significant challenges. Agile's iterative nature may conflict with stakeholder expectations for fixed timelines and detailed upfront planning, leading to misalignment and dissatisfaction (Denning, 2018).

Another persistent challenge is estimation and predictability issues. While Agile emphasizes adaptability, this flexibility can make long-term planning and budgeting difficult. Many organizations struggle with estimation techniques such as story points and velocity tracking, which are often perceived as less precise compared to traditional project management methods (Cohn, 2005). This uncertainty creates concerns for stakeholders who require reliable projections for budgeting and resource allocation. Furthermore, governance and compliance challenges arise in highly regulated industries where documentation and reporting requirements must be met. Agile's preference for flexibility and minimal documentation can sometimes conflict with these regulatory demands,

requiring organizations to strike a balance between compliance and Agile principles (Leffingwell, 2019).

2.9 Challenges in Transitioning to Agile Project Management for Different Roles

Transitioning to Agile project management presents unique challenges for different roles within an organization due to their distinct responsibilities and expectations. While Agile methodologies emphasize flexibility, collaboration, and efficiency, they also introduce role-specific difficulties that organizations must address to ensure successful adoption.

Executives and senior leadership often struggle to align Agile's iterative and adaptive nature with traditional business expectations that prioritize long-term strategic planning and budgeting (Denning, 2018). Agile's focus on continuous value delivery challenges leaders to redefine success metrics beyond budget adherence and deadlines, requiring new performance indicators such as customer satisfaction and business value delivery (Digital.ai, 2023). Moreover, shifting from a hierarchical, command-and-control leadership style to a decentralized decision-making approach demands a cultural transformation, which many organizations find difficult to implement (Leffingwell, 2019). Additionally, leaders in regulated industries face governance and compliance risks, as Agile's fluid processes must still adhere to stringent documentation and reporting standards (Bick et al., 2018).

Project managers and Scrum Masters also encounter challenges in Agile environments, primarily due to redefined roles and responsibilities. Traditional project managers often struggle with their evolving role as responsibilities become distributed across Scrum Masters, Product Owners, and self-organizing teams (Cohn, 2005). Managing stakeholder expectations is another significant hurdle, as Agile's iterative

approach may conflict with stakeholders' preference for fixed scope, timelines, and deliverables (Denning, 2018). Scaling Agile across multiple teams further complicates coordination and consistency, particularly in large organizations (Bick et al., 2018). Additionally, project managers must balance Agile's flexibility with traditional reporting structures, as many organizations still require standardized documentation and performance tracking (Leffingwell, 2019).

Product Owners (POs) play a crucial role in defining and prioritizing the product backlog, but they face challenges in ensuring alignment with business goals while accommodating evolving requirements. Managing a well-defined backlog that reflects changing business needs and technical constraints is complex (Cohn, 2005). Furthermore, many organizations struggle to clearly define the PO role, resulting in conflicts among business leaders, development teams, and stakeholders (Digital.ai, 2023). Balancing competing priorities—such as stakeholder demands, customer needs, and technical feasibility—poses additional difficulties (Denning, 2018). Integrating continuous customer feedback while maintaining a cohesive product vision is particularly challenging in large-scale projects (Leffingwell, 2019).

Development teams experience Agile transformations firsthand, requiring them to adapt workflows, collaboration methods, and technical approaches. Resistance to Agile's iterative nature, frequent feedback loops, and evolving requirements is a common challenge for developers accustomed to traditional models (Bick et al., 2018). Agile also increases accountability, as self-organizing teams must estimate efforts, manage their work, and ensure timely delivery (Digital.ai, 2023). Dependency management becomes a significant issue in large-scale Agile projects, where teams often rely on others, causing

bottlenecks and delays (Leffingwell, 2019). Additionally, Agile's rapid development cycles can lead to technical debt if teams lack proper technical discipline, making long-term maintenance challenging (Cohn, 2005).

Quality Assurance (QA) and testing teams must shift from traditional waterfall testing to Agile's continuous testing and integration approach. This transition requires early and frequent testing, which may be a significant adjustment for teams accustomed to sequential testing phases (Leffingwell, 2019). Moreover, Agile's fast-paced iterations necessitate test automation, but many QA teams lack the required tools or expertise (Digital.ai, 2023). The evolving nature of Agile requirements also makes test planning more complex, as testers must validate functionality against frequently changing specifications (Denning, 2018). Close collaboration with development teams is essential, yet misalignment in testing strategies and development workflows can create inefficiencies (Bick et al., 2018).

Business analysts (BAs) also face unique challenges in Agile environments. Since Agile minimizes upfront documentation, BAs must capture evolving requirements iteratively, which can be particularly difficult in complex projects (Cohn, 2005). Maintaining stakeholder alignment is another significant challenge, as changing priorities can lead to communication gaps (Denning, 2018). Agile's demand for rapid decision-making based on iterative feedback can be difficult if BAs do not have direct access to key stakeholders (Leffingwell, 2019).

User Experience (UX) and User Interface (UI) designers must integrate their workflows with Agile development teams while maintaining design consistency. Agile's short sprints may not provide adequate time for in-depth research and design exploration, often leading to rushed UX/UI decisions (Bick et al., 2018). Additionally, frequent changes

in requirements can make it difficult to ensure a consistent user experience across iterations (Digital.ai, 2023). Many organizations struggle to integrate UX/UI designers effectively into Agile teams, resulting in last-minute design modifications rather than a collaborative approach (Denning, 2018).

Overall, transitioning to Agile project management requires organizations to address the unique challenges faced by each role. Executives must balance Agile's flexibility with business predictability, project managers must redefine their roles while managing expectations, and development teams must embrace new ways of working. Addressing these challenges requires comprehensive change management strategies, continuous training, and strong leadership support to ensure a smooth Agile transformation.

2.10 Gaps in the Literature

Agile methodologies have significantly transformed project management across various industries, including software development, automotive, and aerospace. The Agile Manifesto, introduced by Beck et al. (2001), emphasizes adaptability over rigid planning, making Agile suitable for dynamic environments. However, aerospace projects operate in compliance-driven, high-stakes settings where predictability, stability, and long-term planning are essential (Boehm & Turner, 2004). Despite Agile's widespread adoption, accurately predicting project outcomes, delivery timelines, and cost estimations remains challenging in complex aerospace programs (Serrador & Pinto, 2015). This literature review identifies gaps in predictability within Agile methodologies by analyzing key components such as Agile values, frameworks, practices, estimation techniques, and backlog management, as shown in Figure 2.3. It highlights the absence of structured forecasting methods within Agile and emphasizes the need for hybrid approaches that integrate Agile with predictive project management strategies.

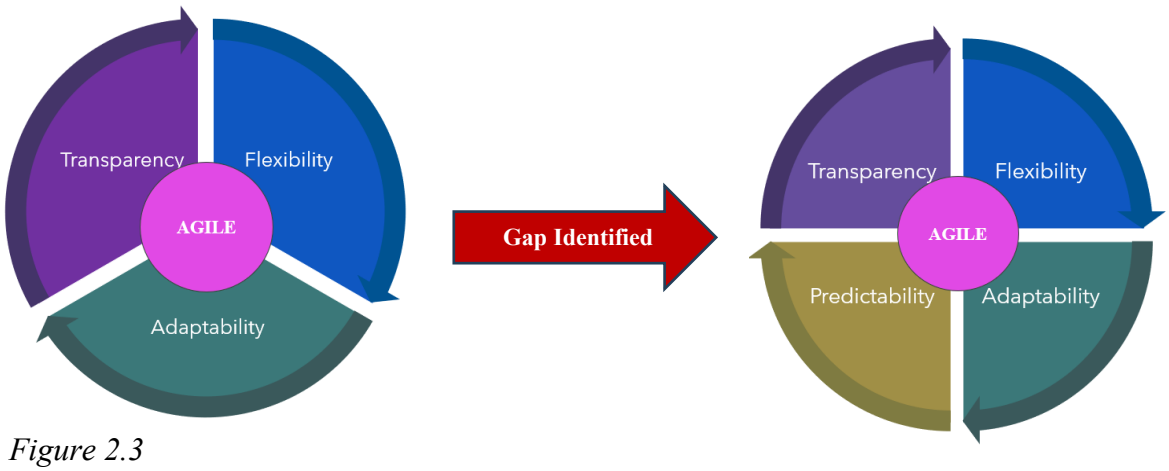


Figure 2.3
Gaps in predictability within Agile methodologies

The Agile Manifesto consists of four core values and twelve principles that emphasize customer collaboration, working solutions, and responsiveness to change (Beck et al., 2001). While these principles enhance adaptability, they do not explicitly address predictability in terms of cost, schedule, and risk assessment. For instance, the principle of "responding to change over following a plan" fosters flexibility but may conflict with the structured governance requirements of aerospace projects, where regulatory compliance and long-term stability are critical (Boehm & Turner, 2004). Additionally, the Agile principle of "delivering working software frequently" ensures responsiveness but can introduce volatility in overall project roadmaps, making long-term forecasting difficult (Highsmith, 2009). Furthermore, Agile's emphasis on "simplicity—the art of maximizing the amount of work not done" contradicts the thorough documentation, risk assessments, and compliance adherence required in aerospace projects (Cockburn, 2007). Similarly, the principle of "continuous attention to technical excellence and good design" supports adaptability but lacks a quantitative approach to measuring project stability and predictability over time (Serrador & Pinto, 2015). The 12th Agile Principle—which states that teams should "reflect on how to become more effective and adjust behavior

accordingly"—promotes continuous improvement but lacks an explicit focus on flow efficiency and predictability. Reflection alone does not ensure that teams actively optimize their processes to enhance stable throughput and delivery confidence. Existing literature on Agile methodologies focuses on enhancing flexibility and collaboration but lacks frameworks that support long-term stability and predictability, which is essential in aerospace project management.

Several Agile frameworks, including Scrum, the Scaled Agile Framework (SAFe), Large-Scale Scrum (LeSS), and Disciplined Agile (DA), have been adopted in large-scale projects. However, these frameworks prioritize iterative delivery and responsiveness rather than ensuring predictable outcomes (Schwaber & Sutherland, 2017). Scrum empowers teams with iterative development and rapid feedback loops. However, its velocity-based planning assumes a stable team composition and backlog, which is often unrealistic in aerospace projects (VersionOne, 2021). Additionally, Scrum's sprint reviews and retrospectives focus on incremental improvements but do not provide robust mechanisms for forecasting overall project completion timelines (Sutherland, 2014). Similarly, SAFe introduces Program Increment (PI) planning to align multiple Agile teams, yet it does not account for systemic delays, regulatory changes, or dependencies common in aerospace projects (Leffingwell, 2019). Moreover, SAFe lacks historical data analysis capabilities, limiting its ability to predict long-term delivery trends. LeSS promotes transparency and large-scale Agile coordination; however, it does not include structured risk management frameworks, making it difficult to predict potential project disruptions (Larman & Vodde, 2010). Additionally, its backlog-driven prioritization approach introduces variability in feature deliveries, which negatively impacts long-term forecasting. Overall, while Agile frameworks enhance team and program-level alignment, they do not provide robust,

predictive project management mechanisms suitable for highly regulated industries like aerospace.

Agile methodologies incorporate practices such as daily stand-ups, sprint planning, and retrospectives, all of which improve communication and adaptability. However, their contribution to long-term predictability remains limited. Daily stand-ups, for example, enhance team communication and issue resolution, but their focus on short-term tasks can lead to misalignment between daily progress and overall project predictability (Rising & Janoff, 2000). Similarly, sprint planning allows teams to commit to achievable goals, but these commitments rely on story point estimations, which are subjective and fail to account for unforeseen blockers or dependencies (Schwaber & Sutherland, 2017). Additionally, while retrospectives encourage continuous improvement, they address predictability issues reactively rather than proactively, meaning disruptions are managed only after they have already occurred (Derby & Larsen, 2006). Consequently, while Agile practices optimize team collaboration, they lack structured methodologies for ensuring predictability across entire aerospace projects.

Agile estimation techniques, such as story points, velocity tracking, and relative sizing, offer a qualitative approach to effort estimation but fail to ensure long-term forecasting accuracy. Story points provide useful effort estimations within small teams; however, their lack of standardization makes comparisons across teams unreliable (Cohn, 2005). Furthermore, story points do not directly translate to time-based forecasts, making it difficult to establish accurate long-term schedules (Hiranabe, 2008). Similarly, velocity tracking helps teams plan for future iterations, but it assumes that team capacity remains constant—an assumption that rarely holds true in aerospace projects, where staffing and

workloads fluctuate due to regulatory constraints and evolving project requirements (VersionOne, 2021). Additionally, velocity tracking does not account for unplanned disruptions, regulatory changes, or risk factors, further limiting its effectiveness in ensuring predictability. Thus, Agile estimation techniques lack a structured, quantitative methodology for maintaining predictability in large-scale aerospace projects.

Backlog grooming plays a crucial role in Agile project management by re-prioritizing work based on evolving requirements. However, frequent backlog changes introduce variability that can negatively impact long-term predictability. Agile backlogs often lack structured risk assessment techniques, which makes it difficult to maintain stable delivery commitments in high-stakes aerospace projects (Leffingwell, 2019). Frequent reprioritization disrupts baseline scheduling, making long-term forecasting unreliable (Highsmith, 2009). As a result, backlog grooming tends to prioritize adaptability over predictability, which contradicts the needs of aerospace programs that require stable, long-term planning.

2.11 Conclusion

The evolution of Agile methodologies has revolutionized project management by prioritizing adaptability, collaboration, and iterative development. These methodologies have been widely adopted across industries, including aerospace, due to their ability to respond quickly to changing requirements and dynamic project environments. However, the aerospace sector is characterized by highly regulated, compliance-driven, and risk-sensitive project landscapes, necessitating a structured approach to project predictability. While Agile enhances flexibility and responsiveness, its effectiveness in ensuring

predictable project performance, cost estimation, schedule adherence, and risk mitigation remains a subject of concern.

Existing literature primarily highlights the benefits of Agile in fostering team collaboration, continuous improvement, and customer-centric development but lacks a comprehensive analysis of its ability to ensure consistent and predictable project outcomes in large-scale, complex industries such as aerospace. The Agile Manifesto (Beck et al., 2001) emphasizes responding to change over following a plan, which inherently contradicts the aerospace industry's need for structured forecasting, regulatory compliance, and long-term planning. Although frameworks such as Scrum, SAFe, LeSS, and Disciplined Agile (DA) provide guidance for scaling Agile in large projects, they do not sufficiently address systemic uncertainties, long-term risk factors, and forecasting limitations that affect project predictability.

A critical gap in Agile methodologies is the absence of structured mechanisms for measuring and improving predictability. Traditional Agile estimation techniques, including story points, velocity tracking, and relative sizing, focus on short-term effort estimation rather than long-term forecasting accuracy. The reliance on qualitative estimation approaches introduces variability in project scheduling, making it difficult to align Agile workflows with fixed regulatory deadlines and compliance requirements (Cohn, 2005; Serrador & Pinto, 2015). Furthermore, backlog grooming, while essential for refining priorities, often leads to frequent changes that disrupt baseline scheduling and introduce unpredictability in project timelines (Leffingwell, 2019). This challenge is further compounded in large-scale aerospace programs that require rigorous compliance documentation, structured validation processes, and risk assessment frameworks to ensure project stability.

Another major limitation of Agile methodologies in aerospace projects is the lack of robust risk mitigation strategies integrated into Agile frameworks. While Agile emphasizes adaptability, reactive risk management approaches such as sprint retrospectives and iterative feedback loops are insufficient in mitigating high-impact risks associated with aerospace projects. These projects involve long development cycles, multi-tiered dependencies, certification requirements, and extensive integration testing, which demand predictability-focused strategies rather than purely iterative adaptations. Agile's limited focus on quantitative risk modeling and predictive analytics restricts its effectiveness in ensuring consistent and measurable project outcomes in aerospace environments (Derby & Larsen, 2006).

The integration of Agile with predictive methodologies, such as Predictive Agile Delivery (PAD) and Agile-DevSecOps frameworks, has been proposed as a potential solution to address these limitations. DevSecOps introduces continuous security integration, automated testing, and real-time risk assessment, which can enhance Agile's ability to maintain structured predictability while preserving its flexibility (Kim et al., 2016). Moreover, a hybrid approach that incorporates historical data analysis, statistical-assisted forecasting, and risk modeling could significantly improve Agile's ability to provide long-term predictability in aerospace projects. This structured approach would enable organizations to balance Agile's responsiveness with the aerospace sector's critical need for stability, compliance, and long-term resource optimization.

In conclusion, while Agile methodologies have brought significant improvements in adaptability, collaboration, and iterative development, they fall short in ensuring predictability in large-scale, high-risk industries such as aerospace. The lack of standardized forecasting models, structured estimation techniques, and quantitative risk management frameworks limits Agile's effectiveness in achieving stable, predictable

project outcomes. Addressing this gap requires a systematic integration of predictive project management principles within Agile frameworks, enabling organizations to achieve a balance between agility and structured project governance. Future research should focus on developing hybrid methodologies that combine Agile's flexibility with data-driven forecasting models, predictive risk assessment techniques, and structured compliance frameworks to enhance the predictability and stability of aerospace project management processes.

CHAPTER III:

METHODOLOGY

3.1 Overview of the Research Problem

This chapter presents the research methodology adopted to investigate predictability challenges in Agile project management and develop a conceptual framework for Predictive Agile Delivery in aerospace projects. Agile methodologies emphasize flexibility and iterative progress, making them highly effective in dynamic environments. However, this flexibility introduces challenges in planning, setting deadlines, and maintaining predictable performance. This research aims to systematically explore these challenges and provide a structured approach for optimizing Agile-based project predictability.

A mixed-methods research design is employed, integrating qualitative and quantitative approaches. The study is structured into five distinct phases, incorporating literature review, surveys, focus groups, interviews, data analysis, and conceptual modeling. Each phase addresses specific research questions and contributes to the development of a structured Predictive Agile Delivery framework.

The methodology ensures academic rigor and practical applicability, focusing on empirical evidence, industry practices, and theoretical insights. This structured approach will provide valuable contributions to Agile project management, particularly within the aerospace sector. Figure 3.1, define research approach step by step.

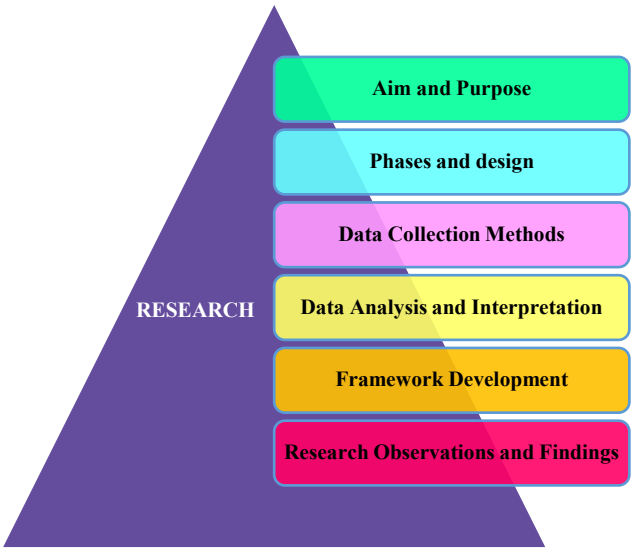


Figure 3.1
Research Approach

3.2 Research Aim and Purpose

Agile project management is known for its flexibility, but its unpredictable nature can lead to challenges in planning, scheduling, and execution. The study aims to address these challenges by identifying key constraints and dependencies that hinder predictability in aerospace project management and proposing solutions to enhance Agile delivery efficiency.

It seeks to systematically analyze industry challenges, existing literature, and project management constraints to define clear research objectives and study phases. By integrating insights from industry data, expert interviews, surveys, and literature reviews, the primary goal of this study is to develop a conceptual framework for Predictable Project Performance and Optimized Project Delivery in Agile-managed aerospace projects.

The ultimate goal is to provide project managers with actionable strategies and methodologies that improve Agile delivery through a predictive approach, ensuring better planning, risk mitigation, and performance optimization.

To address the research problem, this study seeks to answer the following key research questions:

1. What are the challenges encountered by teams and management after transitioning to Agile project management?
2. What solutions can mitigate the pain points faced during Agile adoption?
3. How does the implementation of Agile impact project predictability?
4. Why is predictable performance important for any project success?
5. What constraints and dependencies hinder predictable project outcomes?

By addressing these questions, the study will provide actionable insights and strategic recommendations to enhance Agile project management, ensuring both flexibility and predictability in complex and high-risk industries.

This study aims to explore whether Agile project management affects project predictability and efficiency in aerospace projects. The hypotheses can be stated as follows:

Primary Hypothesis (H_0 - Null Hypothesis): H_0: Transitioning to Agile Project Management does not significantly impact the predictability of project performance in aerospace projects.

Alternative Hypothesis (H_1 - Alternative Hypothesis): H_1: Transitioning to Agile Project Management significantly impacts the predictability of project performance in aerospace projects.

These hypotheses will be tested through a combination of literature review, empirical data analysis, and stakeholder feedback from aerospace project teams.

3.3 Research Design

This study employs a mixed-methods research approach, integrating both qualitative and quantitative techniques to achieve a comprehensive, data-driven understanding of Agile project predictability. Mixed-methods research provides the advantage of triangulation, allowing different types of data to complement and validate each other, ensuring a holistic exploration of the problem. The research is conducted in three distinct phases as shown in

Figure 3.2, each designed to contribute to a systematic analysis of Agile predictability and the development of a structured Predictive Agile Delivery Framework.

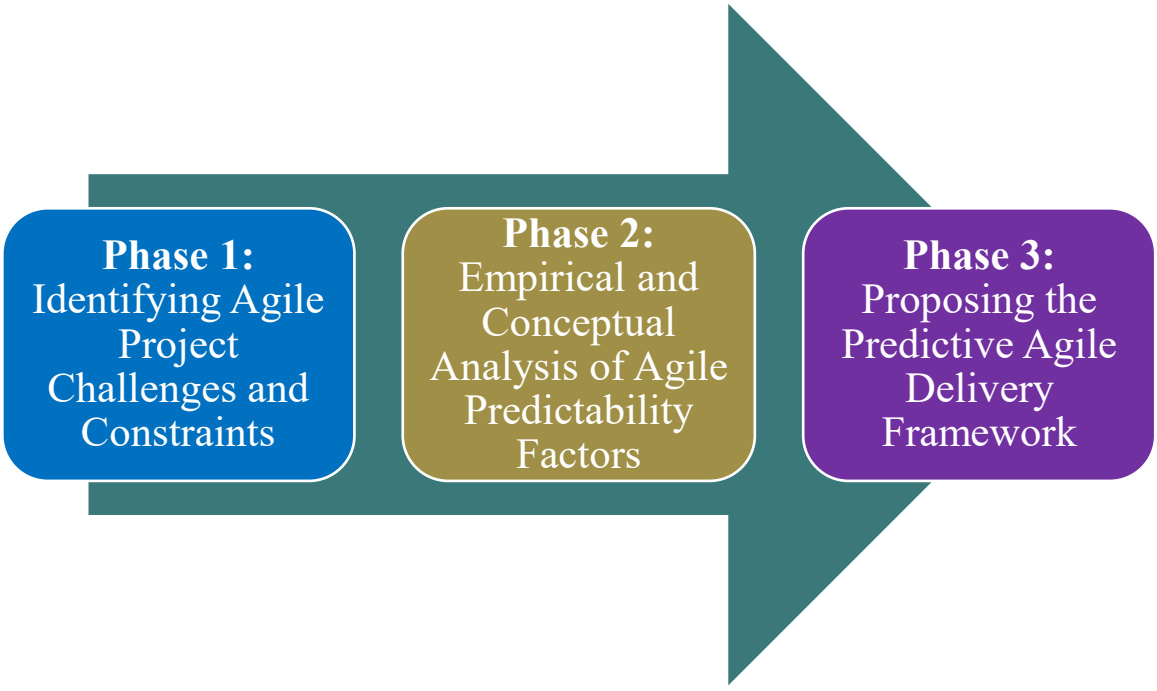


Figure 3.2
Research Phases

Phase 1: Identifying Agile Project Challenges and Constraints

The first phase of the study focuses on identifying Agile project challenges and constraints using data collection techniques such as surveys, focus groups, and interviews. This phase is essential for establishing a foundational understanding of the critical issues affecting Agile predictability.

- Surveys was distributed to Agile practitioners, project managers, and industry professionals to gather quantitative data on common project constraints, variability in Agile performance, and the effectiveness of Agile methodologies in different organizational contexts.
- Focus groups brough together Agile teams and stakeholders to discuss real-world challenges, fostering interactive discussions that reveal shared experiences, bottlenecks, and best practices.
- Interviews with Agile experts, senior project managers, and industry leaders provided in-depth qualitative insights into high-level strategic constraints, dependencies, and decision-making challenges in Agile environments.

This phase ensures that the study captures a diverse range of perspectives and experiences, laying the groundwork for further analysis.

Phase 2: Empirical and Conceptual Analysis of Agile Predictability Factors

This phase employed a mixed methods approach to assess the factors influencing Agile predictability, integrating empirical research with theoretical analysis. Due to confidentiality constraints, direct access to proprietary organizational data—such as retrospective reports, Organizational Process Assets (OPAs), lessons learned databases, and Root Cause Corrective Action (RCCA) reports—was restricted. Instead, data was derived through surveys, focus groups, and expert interviews, supplemented by a

comprehensive literature review of Agile methodologies and project performance trends. This approach ensured a well-rounded analysis, combining practitioner insights with established academic frameworks (Creswell & Plano Clark, 2018).

Qualitative Analysis: Insights from semi-structured interviews and focus groups with Agile practitioners provided a nuanced understanding of challenges, success factors, and dependencies affecting project predictability. Thematic analysis (Braun & Clarke, 2006) was employed to identify recurring patterns across responses. This method followed a six-step coding process: (1) familiarization, (2) generating initial codes, (3) searching for themes, (4) reviewing themes, (5) defining and naming themes, and (6) producing the final report.

Quantitative Insights (Survey Data & Secondary Analysis): A structured survey was conducted to gather data on Agile challenges, performance metrics, existing Agile frameworks, predictability including schedule adherence, scope changes, velocity fluctuations, and defect rates. to identify trends and correlations in Agile project performance.

Comparative Framework Development: Agile methodologies, frameworks, and key performance indicators were systematically compared using survey responses, expert feedback, and existing literature. This approach enabled by identifying common pitfalls in Agile project execution across different organizational structures (Kerzner, 2022). The findings were synthesized into a structured framework that support predictable project performance and optimized delivery.

By incorporating direct input from industry professionals while leveraging established research, this phase ensured a robust and evidence-based understanding of Agile predictability. The methodology aligned with ethical research practices and confidentiality requirements, ensuring data integrity while maintaining academic rigor.

Phase 3: Developing the Predictive Agile Delivery Framework

The final phase focuses on developing a structured Predictive Agile Delivery Framework based on the findings from the first two phases. This framework is designed to provide a systematic approach for improving predictability in Agile project delivery by integrating conceptual modeling, Agile performance metrics, and risk mitigation strategies.

- Conceptual Modeling: A structured model is developed to represent Agile constraints, dependencies, and influencing factors. This model visually maps the relationships between key Agile performance indicators, providing clarity on how different elements interact.
- Framework Design: The Predictive Agile Delivery Framework incorporates industry best practices, statistical insights, and risk management strategies to enhance predictability. It provides guidance on improving estimation accuracy, optimizing workflow efficiency, and mitigating variability in Agile project execution.

By following a structured, iterative approach, this phase ensures that the proposed framework is practical to different Agile settings.

The research follows a structured methodology that progresses through the following key stages as defined in Figure 3.3:

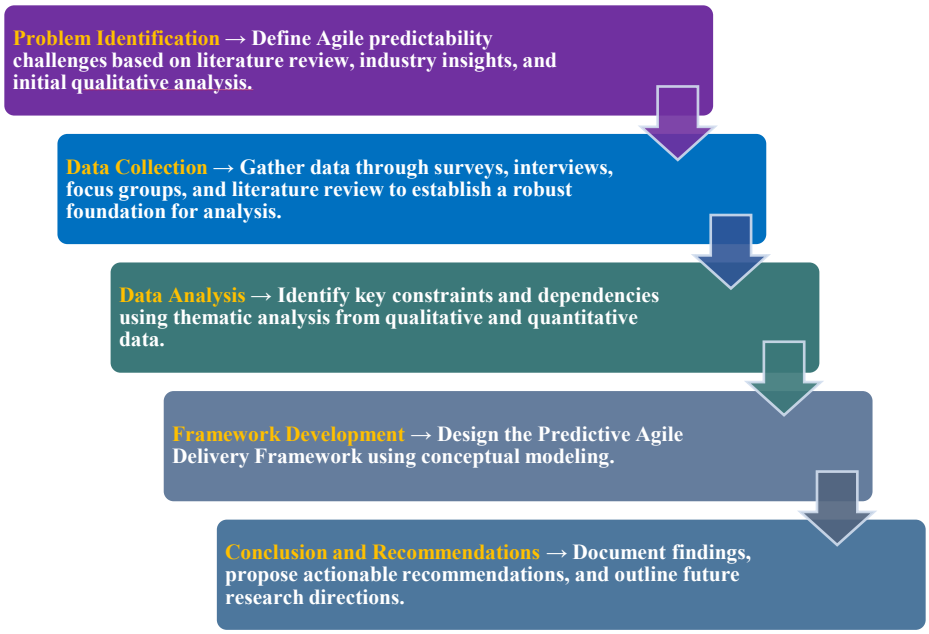


Figure 3.3
Research Framework and Methodology Flow

3.4 Data Collection and Analysis Procedure

Data collection and analysis were critical components of research, ensuring the validity, reliability, and depth of findings. The study utilized both qualitative and quantitative data collection methods to provide a comprehensive understanding of Agile predictability in aerospace projects. Qualitative data collection involved gathering non-numerical insights, often through interviews, focus groups, and thematic analysis of textual data. This method was essential for understanding subjective experiences, behaviors, and perceptions. In contrast, quantitative data collection focused on numerical data, using surveys, statistical analysis, and structured observations to identify patterns and trends (Creswell & Creswell, 2018).

Qualitative Data Collection and Analysis

The qualitative aspect of this research involved semi-structured interviews and focus groups to capture in-depth insights from Agile practitioners and project leaders. Interviews included open-ended questions to explore Agile predictability challenges, factors influencing delivery timelines, and subjective assessments of project success. These interviews provided rich, contextual data, allowing this research to understand the nuances behind project predictability issues.

Focus groups further enhanced understanding by facilitating discussions among Agile team members, revealing common experiences, pain points, and effective strategies. The interactive nature of focus groups allowed participants to build on each other's insights, leading to a more comprehensive exploration of Agile methodologies' real-world applications. Thematic analysis used to identify recurring patterns in qualitative data, applying a structured coding framework to categorize insights based on frequency, emphasis, and relationships between themes (Braun & Clarke, 2006). This approach ensures systematic and rigorous interpretation of qualitative data for this research.

Quantitative Data Collection and Analysis

Quantitative data collection primarily involved surveys distributed to Agile teams across few departments in aerospace organization. These surveys contain structured questions to measure predictability challenges, the impact of Agile practices, and show relation between project management techniques and project outcomes through data.

To ensure comprehensive data collection, surveys incorporated:

- Likert scale questions to assess perceptions of Agile effectiveness and predictability.

- Multiple-choice and ranking-based questions to quantify common challenges and mitigation strategies.
- Demographic and experience-based questions to segment responses based on factors like team size, project complexity, and Agile maturity.

Survey data played a crucial role in identifying trends and relation between Agile methodologies and project predictability.

Ethical Considerations

Ethical considerations were fundamental to the credibility and integrity of this research. Informed consents were obtained from all participants, by clearly outlining the study's purpose, procedures, and for voluntary participation. Confidentiality and anonymity were strictly maintained by securely storing data and ensuring that personally identifiable information is removed from research findings. Participants were also having the right to withdraw at any stage, ensuring their autonomy and comfort throughout the study.

Integration and Conceptual Modeling

To synthesize findings, both qualitative and quantitative data integrated into a conceptual model for “Predictive Agile Delivery Framework”. This model provides a structured framework, offering Agile project teams actionable guidance for improving predictability. A mixed-methods triangulation approach is applied to cross-validate insights from qualitative and quantitative data, ensuring comprehensive analysis and reducing potential biases.

By combining qualitative insights with quantitative validation, this study offered a robust and holistic understanding of Agile predictability. The findings contributed to both academic literature and practical applications in aerospace project management.

3.5 Population and Sampling

To ensure the study captured meaningful insights into Agile adoption within aerospace projects, a purposive sampling strategy was employed. This approach allowed for the deliberate selection of participants with direct Agile project management experience rather than relying on stratified or random sampling methods. Purposeful sampling was particularly relevant for qualitative research as it ensured that participants had deep knowledge of the subject and could provide valuable insights aligned with the study objectives (Morse, 2015a). Selecting respondents capable of articulating perspectives related to the research question was essential for achieving data saturation (Tong & Dew, 2016). Data saturation occurred when additional data collection no longer provided new themes or insights, thereby enhancing the reliability and validity of the findings. According to Morse (2015a), researchers should begin with a small sample, analyze the data for emerging themes, and continue conducting interviews until no new information arises. Overlapping responses helped eliminate potential unknown variables that could otherwise affect the study outcomes, thereby making the research phenomenon clearer and more comprehensible.

To facilitate open communication during interviews, it was crucial to conduct them in a setting that was comfortable for participants (Sivell et al., 2015). Researchers ensured flexibility by accommodating participants' preferences regarding location and scheduling, thereby increasing the likelihood of participation. While purposive sampling provided access to individuals with significant knowledge of Agile adoption, a limitation of this approach was the potential exclusion of certain perspectives that could have enriched the study (Sivell et al., 2015). This research focused on professionals engaged in Agile project management within aerospace projects, ensuring that only individuals with direct Agile experience were included. The total population comprised 399 individuals, categorized into

two main groups: Managers & Leaders and Professionals. Managers & Leaders were responsible for decision-making, strategy, and oversight of Agile implementation, while Professionals were directly involved in Agile execution, including engineering and technical work.

Table 3.1
Population Distribution with Role and Experience

Category	Role	Population	Years of Experience (Avg.)
Managers & Leaders	Director	1	25 years and above
	Associate Director	1	22 years and above
	Senior Managers	3	19 years and above
	Engineering Managers	15	15 years and above
	Associate Engineering Managers	7	10 years and above
Total Managers and Leaders		**27**	
Professionals	Engineers	128	0 to 4 years
	Lead Engineers	93	4 to 8 years
	Senior Lead Engineers	96	8 to 13 years
	Principal Engineers	53	12 to 18 years
	Senior Principal Engineers	2	16 to 22 years
Total Professionals		**372**	
Total Population		**399**	

Sample Size

The population distribution included 27 Managers & Leaders and 372 Professionals, as outlined in Table 3.1: Population Distribution with Role and Experience.

While these experience ranges aligned with industry norms, exceptions existed based on individual competencies.

The required sample size is determined using Yamane's (1967) formula, based on this calculation, approximately 200 respondents is necessary to achieve statistically meaningful results. The sample was proportionally drawn from both groups, with more than 15 participants from Managers &Leaders and more than 180 from Professionals, ensuring a balanced representation.

Yamane's (1967) formula for finite populations:

$$n = \frac{N}{1 + N(e)2}$$

Where:

- N = Total population (399)
- e = Margin of error (5% or 0.05)

Substituting the values:

$$n = \frac{399}{1 + 399(0.05)2}$$

$$n = \frac{399}{1.9975}$$

$$n \approx 200$$

To facilitate data collection, a structured survey distributed via company intranet, and professional networks. The survey incorporated Likert scale questions to assess perceptions of Agile effectiveness, multiple-choice and ranking-based questions to identify key challenges, and demographic and experience-based questions to segment responses. To maintain relevance, inclusion criteria required participants to have direct Agile experience in aerospace projects, while individuals without Agile exposure or those in non-project-related roles are excluded. By focusing on participants with practical Agile

experience, this study aims to provide actionable insights into the relationship between Agile methodologies and project predictability in complex, high-risk environments.

For qualitative data collection, sample sizes were determined by data saturation, ensuring that interviews and focus groups continued until no new themes emerged. Standard qualitative sampling practices suggest that interviews typically involve 10-30 participants, focusing on Agile managers, leaders, and experienced professionals. In addition, focus groups generally consist of 2-5 groups, with 5-8 participants per group (totaling between 10 and 40 participants), allowing for group discussions on Agile adoption experiences.

The final sampling plan consisted of more than 195 survey participants, 18 interviewees, and 4 focus groups with 7 participants per group (totaling 28 participants). This approach ensured a balanced integration of quantitative and qualitative methods, providing both broad numerical insights and in-depth qualitative analysis. The combination of surveys, interviews, and focus groups facilitated a comprehensive understanding of Agile adoption while ensuring data saturation and methodological rigor.

3.6 Research Design Limitation

Every research approach has its limitations, and acknowledging these limitations ensures transparency and critical analysis. The following limitations are acknowledged in this research:

- Sample Size: The sample size for surveys, interviews, and focus groups was not representing the entire population of Agile practitioners in aerospace projects, which limit the generalizability of findings.

- Data Availability: The availability of historical data from aerospace companies using Agile methods was restricted due to confidentiality agreements and proprietary practices.
- Research Scope: This study focused on Agile projects within the aerospace industry, which not fully reflect challenges and practices in other sectors.

3.7 Conclusion

This chapter outlined the research methodology used to investigate Agile predictability challenges. The study follows a mixed-methods approach, combining a literature review, surveys, focus groups, and quantitative data analysis. By integrating qualitative and quantitative analysis, the study provides a comprehensive understanding of Agile predictability constraints and proposed a structured solution to improve delivery outcomes. The next chapter presents the findings and analysis, evaluating key predictability metrics and assessing the effectiveness of the proposed framework.

CHAPTER IV:

RESULTS

4.1 Introduction

This chapter presents the results and findings of the study on predictability challenges in Agile project management within aerospace projects. The research aimed to identify key constraints, dependencies, and factors affecting project predictability, leading to the development of a Predictive Agile Delivery Framework.

The results are derived from qualitative and quantitative analyses, including surveys, interviews, focus groups, and thematic analysis. The data collected from the professionals across various roles in Agile aerospace projects has been analyzed to identify patterns, correlations, and key insights.

The findings are structured as follows:

1. Survey Results – Agile adoption and predictability challenge.
2. Thematic Analysis of Survey data – Common themes from survey data.
3. Interview and Focus Group Data– Agile adoption and predictability challenge.
4. Thematic Analysis – Common themes from qualitative (I&FG) data
5. Mapping Thematic Analysis to Research Questions and Hypothesis
6. Final Hypothesis Analysis
7. Proposed Solution to address Literature Gap for Agile Value and Principal
8. Predictive Agile Delivery Framework – Integration of insights into the proposed Predictive Agile Delivery Framework.

The results offer a comprehensive understanding of Agile predictability issues, supporting the development of structured solutions for improved project performance.

4.2 Survey Results

The survey result collected data through 198 responses to provide a comprehensive view of the challenges, benefits, and overall impact of Agile Project Management (APM) on project predictability in aerospace refer Appendix D for survey data. The survey results. With a strong representation of developers, Scrum Masters, and Product Owners, the data captures varying perspectives from team members and management. However, the low number of Release Train Engineers (RTEs) and Agile Coaches suggests limited enterprise-level Agile leadership, which may impact Agile maturity across the organization.

RQ#1 What are the challenges encountered by teams and management after transitioning to Agile project management?

A major challenge reported was inter-team dependencies, cited by 72% of respondents, which often caused delays and reduced sprint predictability. Scope changes (62%) were another frequent pain point, making it difficult for teams to maintain stable backlogs. Backlog refinement inconsistencies (37%) led to misaligned sprint planning, while 49% of respondents reported unclear ownership, causing bottlenecks in workflow.

Agile's effectiveness also varied based on project type. 36% found Agile well-suited for iterative software development, whereas 22% faced difficulties in safety-critical and hardware-intensive projects. The regulatory landscape posed another obstacle, with 45% stating that Agile was challenging to align with compliance needs, and 24% reporting that it did not sufficiently support regulatory requirements.

Leadership alignment with Agile principles was another challenge, with 34% indicating a disconnect between Agile goals and leadership expectations. Resistance to change (47%) and difficulties in adapting legacy processes (39%) further hindered Agile adoption. Agile training received a 3.34/5 average rating, suggesting moderate effectiveness, but 25% of respondents felt it did not fully address real-world challenges.

Key Observations:

- Dependency resolution (54%) and cross-team risks (36%) are major blockers.
- Agile documentation is deprioritized (41%), leading to compliance tracking difficulties.
- Resistance to change (47%) and leadership misalignment (34%) slow down Agile adoption.
- Agile training impact is moderate (3.34/5), indicating gaps in practical implementation.

RQ#2 What solutions can mitigate the pain points faced during Agile adoption?

To address these challenges, organizations must adopt structured solutions. Standardizing Agile practices across teams (54%) and strengthening dependency management (42%) are among the most recommended improvements. Organizations that implement structured backlog refinement, dependency mapping, and regulatory-aligned Agile frameworks tend to experience better predictability and fewer disruptions.

One key solution is improving backlog management and prioritization (49%), as inconsistent backlog refinement has led to frequent scope shifts. Establishing a cross-functional Agile Program Management Office (APMO) could help in managing dependencies, aligning sprint goals, and improving cross-team collaboration.

Additionally, expanding Agile training programs (38%) is crucial to enhance practical understanding of Agile principles beyond theoretical concepts. Survey findings indicate that while 183 respondents attended "Agile 101" training, only 50 are Certified Scrum Masters (CSM) and 20 Certified Product Owners (CPO), suggesting a gap in advanced Agile proficiency.

PI Planning (47%) was found to help teams anticipate dependencies, although external constraints limited its effectiveness. Agile tools improved sprint predictability for 41%, but mixed results were observed due to backlog instability (36%). Retrospective adherence was inconsistent, with 17% rarely or never conducting them, reducing the opportunity for continuous improvement.

Key Recommendations:

- Implement full fledge scaled Agile frameworks (SAFe) to improve enterprise-wide alignment.
- Use Monte Carlo simulations for better project forecasting and risk management.
- Introduce data-driven Agile analytics tools to predict sprint completion rates based on historical trends.

RQ#3 How does the implementation of Agile impact project predictability?

Project predictability remained a central concern. Agile's impact on project predictability is mixed. Sprint velocity (67%) is the most tracked metric, followed by burndown charts (49%). However, only 27% of respondents measure predictability scores, highlighting a gap in tracking Agile effectiveness.

While 41% reported that Agile improved deadline adherence through better sprint planning, 37% noted that scope changes and dependencies frequently delayed deliverables.

22% found Agile challenging for deadline-driven projects, especially in compliance-heavy environments.

Predictability was assessed using sprint velocity (67%), burndown charts (49%), and predictability scores (27%). However, 52% of respondents cited inter-team dependencies as the biggest challenge, followed by difficulty in aligning sprint goals across teams (46%) and lack of standardized Agile processes (39%).

Predictability scores reflected varied experiences:

- 31 respondents reported "greatly improved" predictability.
- 47 found it "somewhat improved."
- 82 reported that predictability "somewhat worsened."

The survey also indicates that while Agile promotes early risk identification (39%), it struggles with tracking long-term risks (37%), making it difficult to predict outcomes in large-scale aerospace projects.

Key Observations:

- Predictability scores (3.61/5) indicate moderate improvement but not a transformative shift.
- The primary reasons for reduced predictability included poor story estimation (177 responses) and inter-team dependencies (164 responses). These are the biggest blocker to predictable Agile execution.
- Backlog instability (36%) and frequent scope changes (62%) reduce Agile's effectiveness in long-term forecasting.

RQ#4 Why is predictable performance important for any project success?

Predictable performance is crucial for ensuring timely project delivery, managing stakeholder expectations, and maintaining cost efficiency. 41% of respondents found that

Agile helped meet deadlines due to structured sprint planning, while 39% noted that Agile improved risk management by identifying issues earlier. However, 36% found Agile only partially effective due to cross-team risks not being well-managed.

For long-term forecasting, 37% reported Agile was useful, but 41% stated that shifting priorities often disrupted roadmaps. 22% felt Agile was not effective for long-term planning, suggesting a gap in Agile methodologies for strategic foresight.

Agile's impact on cost management was mixed. While 44% found it beneficial in reducing rework, 35% noted that frequent changes sometimes increased costs, reducing cost predictability. This underlines the importance of balancing flexibility with structured planning to achieve stable and predictable project execution.

Similarly, 48% reported improved defect tracking through continuous feedback, but 19% stated that Agile did not significantly improve defect management due to regulatory testing constraints.

Key Insights:

- Agile improves defect tracking (48%), yet regulatory testing cycles remain a bottleneck.
- Long-term forecasting (37%) remains difficult, impacting project budgeting and resource allocation.
- Cost predictability (21%) is not significantly improved, requiring better cost-management strategies.

RQ#5 What constraints and dependencies hinder predictable project outcomes?

Organizational silos (174 responses) and conflicting priorities were cited as major obstacles. The lack of standardized Agile processes (160 responses) contributed to inconsistencies across teams, affecting planning and execution. 89% of respondents

identified poor estimation as the biggest challenge, followed by inter-team dependencies (72%), scope changes (62%), backlog prioritization issues (49%), and dependency tracking (38%) further limited Agile's ability to deliver predictable results.

Resistance to change (47%) and difficulty adapting legacy processes (39%) continued to impede Agile transformation. Additionally, 34% reported leadership misalignment with Agile goals, suggesting a need for greater executive buy-in.

Technical debt was another constraint. While 37% found Agile helped manage technical debt, 41% reported that Agile increased short-term debt due to fast-paced delivery cycles.

Regulatory compliance remains a key hurdle, with 45% finding it difficult to align Agile with certification requirements such as DO-178C and FAA/EASA standards.

Key Challenges:

- Organizational silos (174 responses) hinder Agile efficiency.
- Lack of standardized Agile processes (160 responses) results in inconsistencies.
- Poor story estimation (89%), Inter-team dependencies (72%), scope changes (62%), and backlog prioritization issues (49%).
- Compliance challenges (45%) slow down Agile adoption in safety-critical projects.

4.3 Thematic Analysis of Survey Data

Agile Adoption & Cultural Shift: The transition to Agile in aerospace projects has been met with mixed reactions, particularly due to resistance to change (47%) and difficulty adapting legacy processes (39%). While Agile promotes flexibility and early risk identification, its alignment with regulatory frameworks (45%) remains a challenge.

Additionally, the absence of enterprise-level Agile leadership, as indicated by the low presence of Release Train Engineers (RTEs) and Agile Coaches, has hindered Agile maturity. The effectiveness of Agile training (3.34/5) suggests moderate success, but real-world implementation gaps persist. Leadership misalignment (34%) further complicates Agile adoption, highlighting the need for a structured transformation strategy.

Role Clarity & Accountability Issues: Unclear ownership (49%) emerged as a critical issue, leading to workflow bottlenecks and inefficient execution. The survey revealed inconsistencies in backlog refinement (37%), which contributed to sprint misalignment. The limited number of Certified Scrum Masters (CSMs) and Certified Product Owners (CPOs) further indicated a gap in Agile proficiency, potentially affecting decision-making and execution. Without a well-defined Agile Program Management Office (APMO), managing dependencies and aligning sprint goals across teams remains a challenge.

Challenges with Agile Estimation & Work Breakdown: Poor story estimation (89%) was the most significant challenge, leading to unpredictable sprint outcomes. Backlog instability (36%) and frequent scope changes (62%) exacerbated this issue, reducing the effectiveness of Agile forecasting. Inadequate backlog prioritization (49%) further strained Agile planning, making it difficult to maintain stable workflows. Additionally, while Agile tools improved sprint predictability for 41% of respondents, their impact was limited by ongoing estimation inaccuracies and backlog volatility.

Project Predictability & Velocity Issues: Agile's influence on project predictability was mixed, with only 31 respondents reporting a "greatly improved" predictability score. While sprint velocity (67%) and burndown charts (49%) were the most tracked metrics, only 27% measured predictability scores, revealing a gap in assessing Agile effectiveness. Cross-team dependencies (52%) and difficulty in aligning sprint goals

(46%) were major blockers to consistent velocity. Agile improved early risk identification (39%) but struggled with tracking long-term risks (37%), making it less effective for large-scale aerospace projects.

Agile Meetings & Collaboration Challenges: Although Agile promotes collaboration, retrospective adherence was inconsistent, with 17% of teams rarely or never conducting them, reducing opportunities for continuous improvement. PI Planning (47%) helped anticipate dependencies, but external constraints often limited its effectiveness. Dependency resolution (54%) and cross-team risks (36%) were major blockers to seamless collaboration. Additionally, organizational silos (174 responses) and conflicting priorities further restricted Agile efficiency.

Cross-Team & Cross-Discipline Collaboration: Inter-team dependencies (72%) were a significant hurdle, frequently causing sprint delays. The lack of standardized Agile processes (160 responses) resulted in inconsistencies across teams, making it difficult to synchronize efforts. Dependency tracking (38%) was another weak area, contributing to unpredictable outcomes. Establishing an Agile Program Management Office (APMO) and adopting structured dependency mapping practices could help mitigate these issues.

Customer & External Stakeholder Alignment: Agile's adaptability to stakeholder needs was inconsistent. While 41% of respondents found Agile effective in meeting deadlines, shifting priorities (41%) often disrupted roadmaps, making long-term planning difficult. Compliance requirements posed additional challenges, with 45% finding Agile difficult to align with industry regulations such as DO-178C and FAA/EASA standards.

Workload Management & Stress: The fast-paced nature of Agile sometimes increased technical debt, with 41% of respondents reporting a short-term rise in unresolved issues. Frequent scope changes (62%) added to workload instability, impacting team

morale and stress levels. Additionally, the lack of standardized processes and clear role accountability often led to overburdened teams, increasing burnout risks.

Agile Suitability for Different Work Types: Agile was most effective in iterative software development (36%) but faced challenges in safety-critical and hardware-intensive projects (22%). The regulatory landscape further complicated Agile implementation, with 24% reporting that it did not sufficiently support compliance requirements. For long-term forecasting, only 37% found Agile useful, while 22% felt it was ineffective for strategic planning. This suggests that Agile methodologies need better integration with structured project management frameworks for non-software projects.

Leadership & Organizational Support: Leadership alignment with Agile principles was a recurring concern, with 34% indicating a disconnect between Agile goals and leadership expectations. Resistance to change (47%) and difficulty in adapting legacy processes (39%) further slowed Agile adoption. Additionally, the low presence of enterprise-level Agile roles such as Release Train Engineers (RTEs) and Agile Coaches suggested that leadership support for Agile transformation was insufficient. Greater executive buy-in and structured Agile scaling frameworks, such as SAFe, are needed to improve enterprise-wide adoption and predictability.

4.4 Interview and Focus Group Data

The interview and focus group data were successfully collected, providing valuable insights into the factors that influenced Agile adoption, project predictability, and performance challenges in aerospace projects. The data included perspectives from industry professionals, capturing real-world challenges, dependencies, and constraints that affected project outcomes. This qualitative dataset served as the foundation for identifying key themes, patterns, and emerging trends (see Table 4.1 for a summary of collected data).

In the next section, a thematic analysis is conducted to systematically examine the data, categorize recurring themes, and derive meaningful interpretations.

Table 4.1
Interview and Focus Data

Interviewer and Focus Group Data	**Challenges that faced by team and management after transitioning to Agile Project Management.**	**Impact of implementation of Agile Project Management on predictability of the project performance.**	**Importance of predictable performance for any project.**	**Constraints and dependency that can hinder project performance typically found in aerospace long term projects.**
I&FG#1	Agile itself has well defined roles and responsibilities. When taken as a whole, it appears to solve multiple problems with the entire software development process. What agile itself does not deal with is when anyone on the team is not meeting the expectations of their role. What happens when a SM cannot manage a scrum team well. What happens if a PO does not do everything they are supposed to. And a big one, Agile assumes (it's the primary driver for its usage) that the development team is the full of experts who are the "best"	Taken at face value, I think Agile will very much help the predictability of project performance. Better than most paradigms I've experienced. However, as noted in #1, not being able to execute agile as agile is expecting to be executed can hurt predictability in such a way that the benefits Agile brings can be difficult to find. I do believe though that even with many	Extremely useful. I think an important part to discuss here is that the individual contributor tends to not understand the benefits of predictable performance - and they see any "extra" effort as wasted or sometime stifling to their day-to-day work.	#1 above I believe is the primary challenge I encounter - namely the ability of everyone at every part of the release train above to execute ALL aspects of their role at a high level. Additionally, when a team tries to be agile, but doesn't fully commit, or worse does things that are actually anti-agile. I don't believe that long vs short term projects have any impact. Finally, in the aerospace industry, there is SO much work that is focused on things outside of specific features. It's very easy to understand the concepts of agile when discussing in terms of features in the traditional sense, but once you add in the extremely large amount of work that doesn't

	individuals to determine things like story size, how to technically solve the problem, and that they will have an inherent desire to do the right thing and meet commitments. There isn't lots of guidance from Agile on how to handle when these types of roadblocks are hit.	challenges in a release train where the team isn't meeting the expectations of their role, I have still witnessed a significant increase in predictability - it's just that the predications indicate that the work cannot be done as fast or as cheap as the business wants.		have the strong connection to the user as a feature does it breaks down. For example, the concept of defining stories from the perspective of a persona is very simple when just talking about adding a feature to an application. But the team will be challenged with how to write a story for a SOI audit, or how to write a story for a SAS etc.
I&FG#2	Splitting stories, appropriately sizing work, communicating WIW	Better communicatio n with team and stakeholders (PE, TPM). More frequent updates to status to assess on track/at risk/behind	Past performance informs future performance. If we don't know how we did, we can't accurately forecast and plan what we can do. If we don't operate predictably, we can't execute to that plan	Scope creep, customer/certification reviews and approvals, process and regulatory environment
I&FG#3	During our program's initial Agile transition, we were told to expect a few months of less efficient work while we got used	The predictability of the program became better after implementing Agile. Prior to	Predictable performance proves that we are reliable to our customers. We are able	The changing of requirements late in development can be very challenging. I understand this is one of the 12 Principles, to Welcome changing

	to the new process. My team from prior to Agile was essentially the same, as the "software" team stayed mainly intact. I believe due to our team staying the same and keeping the same type of work, we actually transitioned very well. As the SW team was able to focus on immediate expectations (1 sprint) versus various long term work (whole build), we could easily implement and review that same work. All in all, for the software team specifically, Agile was a great move for us. I think the verification teams experienced more challenges, as most were not used to short term (2 week) testing + reviews.	Agile, builds would be months long and implementation work would be open for weeks at a time. I remember some instances of implementing software, and then the reviewer wouldn't get to the review for a month. By the time they looked and added any findings, I would often have to completely review the PR/update again in order to re-familiarize myself with the work again. After Agile, and especially after implementing the 6 week build cycle, we were able to focus completely on a specific amount of work. That work would be done within a	to meet their needs with quality and accountability. This helps improve our appearance with the customer for the current program, but also can help us improve our standings with any future programs.	requirements, but it definitely can cause some headaches. Especially in this program, there are lots of requests from the customer towards the very end of the delivery. It can be difficult for our team to provide the updates and verification for these changes when the customer doesn't meet our deadlines for providing these changes.

		sprint, and verified within the build, therefore if any software issues were found in testing, it could be fixed immediately (instead of finding an issue after the software freeze). This greatly improved the predictability and understanding of the work and changes going into each build.		
I&FG#4	During my time at Collins, I did not personally witness the actual transition to Agile methodology. However, I started my career within agile at Collins. Overtime, I have naturally witnessed several challenges that I think are prominent on a weekly basis. One challenge is determining the basis for measurement of a story point across all of the teams for consistency. I think driving towards a	I think Agile provides more visibility on project performance relative to cost and schedule. I believe that Agile helps to track against deadlines and cost much more effectively than a traditional waterfall approach which could be captured in Microsoft	Predictable performance for any project is important as it allows for quicker planning, pivoting, and mitigation of issues on a quicker timeframe. You are almost forced to consistently keep track of schedule, cost, performance with	External dependencies (waiting on approvals from customer, added scope from customer, etc), issues with internal tools/equipment (test stands aren't cooperating, issues with the systems rigs/labs, etc), technical findings during implementation leading to larger effort, onboarding new engineers to highly complex systems and software and expecting the same outcomes as seasoned engineers

	consistent scale across all teams will help to drive better understanding of what teams are voting towards, helping new hires grow on some form of consistent basis, and help project engineers better track metrics relative to actuals and the bid. I think another challenge is determining total capacity in a consistent way without facing any type of discrepancies or ambiguities.	Project for example.	predictability built in.	
I&FG#5	The single largest challenge faced is getting teams to break down stories into small vertical slices. After that I would say getting teams to conduct retrospective where they truly focus on improving as a team.	The first 6-9 months the predictability was impacted as the team formed.	Project execution is the single largest driving factor for a project.	Last minute changes driven by the customer tend to be the biggest factor in hindering project performance.
I&FG#6	I think one of the biggest challenges we face regularly is the vast diversity in projects and products that the release train manages. The underlying assumptions around cross-functional and fully capable teams	Predictability is probably increased in the sense that you can generally see the progress of scope completion and the plan for the remainder, and	In my mind as a contributing engineer, predictable performance on a macro level is important for the purpose of making corrections	The projects are long term which means there is a lot of history and every project is unique in different ways. It is inevitable that there is team makeup churn through the lifecycle. Maintaining a constant pipeline of knowledge transfer to younger engineers is very

	are not true across the release train because there are just so many different types of work going into the generic hopper. Problems arise with quality and efficiency, as well as constant jumping around between products.	can more or less predict how it will pan out if nothing changes. Allowing corrections along the way. I think it comes at the expense of quality and efficiency in the implementation I have seen.	and adjusting a plan as needed. But should not come at the expense of quality and technical expertise of the team working on it.	difficult. Made even more difficult when projects have completely different sets of history and knowledge required and a team is jumping around to different projects.
I&FG#7	Developing priorities and structures and communication to facilitate moving work around.	initially there was chaos, but over time (3+ years), it seems predictability has improved, though we are struggling still to meet our predictability goal.	Very, very high	Customer-provided technical requirements, funding decisions, wrong assumptions about the work made by team, expansion of scope (discovery of tech debt) during execution, changing certification expectations
I&FG#8	It can sometimes be hard to align with customer/external schedules. Agile allows for quick implementation of features; however delayed feedback from the customer can cause additional challenges.	There is still some unpredictability in Agile, however it focuses on the shorter timescale. So roadblocks, issues, and concerns are raised much sooner than traditional project management.	Predictable performance is important for current execution and future bid pursuits.	Integration of large, complex systems. There are many moving parts that all need to come together seamlessly and work together. Unclear scope definition can create bottlenecks and sometimes limit teams' abilities to implement features.

I&FG#9	I haven't personally experienced a transition to Agile, but I can speak to the current challenges of Agile Project Management. Some challenges can be caused by simply by differences between the theoretical Agile Methodology and the practical, likely hybrid version of the Agile Methodology. Key assumptions and dynamics may not be applicable at all, driving challenges around predictability/velocity, and work prioritization.	Agile facilitates projects staying on schedule and identifying new scope early on in the life cycle. However, a consideration which maybe not tied to performance is the financial impact of short term predictability over long term project schedules and budgets. Concepts like swarming that tend to improve predictability often cause increase in project spending, affecting overall performance.	Predictability allows the company to deliver high quality products to customers within project schedule. When done right it can serve as a solid estimate for future projects to plan around.	There are several constraints the can hinder project performance. Most of them I think can be tied to a hybrid agile methodology being taken. Assumptions like agile teams being cross functional, able to dedicate full time to the team, and long lived are often not the case. Causing churn and constant re-adjustment of team dynamics and increased inter-team dependency. Additionally, another constraint exists around technical expertise. If everyone in a team could work every tasks performance would be a lot better. However, in reality technical expertise becomes a significant bottleneck where specific individuals are required to maintain performance.
I&FG#10	One of the biggest challenges we faces was getting used to the habit or breaking up the work into small enough pieces so each piece could be accomplished inside the sprint timeline. This in conjunction with estimating the scope size using	I'm not sure if I can say that has made the project more predictable. My intuition would be that it has, but I don't necessarily have the data to back me up. One thing it	Predictable performance is important for the health of the organization for a few reasons. Customer relationships will suffer from having unpredictable	Constraints include resources (computer, knowledge, humans, technology), cumbersome documentation. Dependencies include matching the appropriate amount of scope approved to work in order to keep everyone engaged while not having too much

	story points was difficult for the teams to get used to. One other related challenge is the discipline of writing well-defined stories that capture the scope clearly, with a definition of done that is easily verifiable. We still do not have great discipline in this area and people are needed to champion this behavior.	did was break apart the planning and distributed the responsibility to the teams that own the work. This is both good and bad. Having one highly knowledgeable scope and estimate the size of the work tends to lead to fewer scope misses from the planning level which can help to insure that those misses aren't introduced in the implementation phase. The problem with that approach is that the "devil is in the details". The team that own the work, IF they have done their due diligence in planning have a better idea of what their team is capable of achieving in a given amount of time. But the distribution	timelines on our work. These expectations can be managed, but it's important to meet the commitments make. Predictable performance is also important for our workforce. The stress of an upcoming deadline that no one is prepared to meet successfully is unpleasant for everyone involved. The stress can cause burnout, turnover, and have impacts on employees careers. Missing deadlines–or working overtime to meet them–leads to lower job satisfaction overall.	which necessitates higher more people. Another dependency is getting proper engagement from Cert authorities throughout the development process. Documents reviewed and cleared in early stages will be re-reviewed in later stages and deemed insufficient can lead to massive re-work for the project.

		can also lead to more scope misses in the planning phase since the teams planning the work may "not know what they don't know" which means they don't ask questions for the details they don't know they are lacking.		
I&FG#11	NA, I was not here when the transition took place.	Agile can increase the intermediate predictability and help identify issues that will cause schedule creep earlier than traditional project management.	Predictability is arguably the most challenging component to managing any project in the engineering domain. It will establish trust with the customer and will provide tremendous profitability for the company. .	Whenever customer input is needed to proceed to the next step of the process. Also if there are teams that are not working in the same department or organization this will also cause issues.
I&FG#12	In this case, the most challenging aspect of the transition in my opinion was work assignment. Work had to be divided such that it fits the agile framework and for this case, two week periods	For the projects I have been involved so far, agile framework has worked in our favor to breakdown work into smaller chunks, i.e.,	I think predictable performance help planners allocate money, stay on top of release train goals and meet long term	Some aerospace projects that don't fit into agile can have constraints such as reviews and SME inputs especially for new type of work that really can create difficulties in terms of planning. Projects that involve outside dependencies such as

	called sprints. Work allocation also involves voting for stories on a scale to measure complexity and time it would take to be completed. This becomes challenging because some work is so large that fitting into a sprint is a big challenge, breaking it up into smaller chunks is unreasonable and makes planning a bit difficult. Such kind of work, I have observed, will fit into two sprints or more and there agile is not very favorable for such activities.	work packages into stories that accomplish objectives of the work package over a period of time; usually a few sprints. It is important to note that predictability of timeline for completion can be well understood with agile which from a Project Engineering perspective, it becomes easier to allocate funding and manpower.	commitments. Predictable performance also helps with long term planning and staying organized.	certification work can also fall into the category of work that doesn't fit the agile framework.
I&FG#13	Discipline wise project management and integration of the stories from various disciplines (SW, EE, ME, PM, etc)	In certain disciplines the impact of implementation was on the positive side as they were able to break their activities effectively using stories. Whereas for certain disciplines where there are long term goals. For example the Class B/A product	PPP is very important as it keeps monitor on the current status of the project and avoids any overruns including cost and schedule.	External factors - Customer review and feedback on the documents, DER feedback, SCM, etc. Internal factors - collaboration between Engg and operations.

		design, procurement and testing it was challengeable.		
I&FG#14	Pressure to complete the given task with 10 days;. It's always meeting a deadline. Deadline after deadline gives stress, specially when you're loading to your full capacity with no buffer.	Faster delivery of result	to foresee the potential risks that may come during the course of project	Dependency on the client's response every time, delay of which sometimes leads to re-work
I&FG#15	If engineers couldn't able to groom the Scope of the work due to lack of knowledge/efficiency results in incorrect estimation and while implementation it will take more efforts and engineer needs to stretch to meet the milestone along with the planned work in the sprint and PI.	Agile gives us the clear vision on the project performance due to proper planning we can predict the project performance, but if we got any walk in work that prediction will get changed.	we should plan every project with positive prediction only to get it success.	no idea
I&FG#16	Team: On individual level planning, evaluating and prioritizing the task within the provided time box initially was tough. Learning opportunities are limited and dependency more	Impact: Agile KPIs and ceremonies are key to predict project performance. Keeping Agile principle in mind surely enhance the	Importance: Helps us understand where we have to put our efforts to succeed and deliver the product faster with quality. How shortcomings,	Constraint and dependency: Scope, Time, cost, resource, quality

	on SME due to more focus on given task schedule rather than exploring the task.	project performance.	ownership can be handled. planning and execution is key to success for any project.	
I&FG#17	1. Good impact on engineers on transitioning to Agile as a engineer can take up new tasks and improve there knowledge	Able to finish the work on time as it has time constraint and as we plan accordingly to finish the work.	Performance is increasing as the team is learning new things in each sprint, engineers skills are developing.	No constraints and we have dependency we are communicating with other agile teams and getting the things done.
I&FG#18	1. Correlating what was bid at the beginning of the program vs the actual effort spent. 2. Difficulty in accounting for support activities	1. Helps in tracking day-day activities easily and resolve dependencies 2. Accommodates changes relatively easily 3. Difficulty in assessing the actuals to the BOE or bid estimates.	It gives clarity on the project. Helps plan things in terms of resources, schedule, etc. It also helps in estimates for future projects also. Helps to look for optimizations. , under utilization, burn-out.	changing requirements/design, availability of hw/sw resources, inter/intra team dependencies, maintaining the team in the long run
I&FG#19	FCS had been working on Waterflow project execution from quite some but when we transitioned to Agile 2-3 years ago we had to deal with lot of challenges. - Understanding	Agile Project Management has a huge impact on Predictable Project Performance considering. We do rigorous planning	Its very important for a project for Predictable Project Performance as it allows us understand ahead of time about delivery in	There are several constraints which usually impacts a aerospace projects eg: - Dynamic requirements change requests, since the project lengths are usually very long ~10 years from scratch to entry into service. There are several additional

	Agile: Since it was quite new to us then, many of the Team members were having lack of clarity about Agile and its difference from Waterflow. - Meeting timing: Many Team members were facing challenges in the number of meetings like DSU, Backlog Grooming , Retro, planning extra. There were some resistance from the Team on the number of meetings we do in Agile in comparison to Waterfall. - Lack of understanding on Technical : Agile requires Team to be cross function, how ever because of Team structures we had on waterfall based on Type of Application (MNT, PLT, AF) Team took some time to be independent as the Team had limited overall understanding on the entire task of 777X considering its size.	through out the program. Entire Team dedicatedly meet and plan for ~ 2 days every quarter with the capacity, planned vacations and the chunk of task we have along with dependencies and risk. As we go into sprint execution which is also if a short span(2 weeks) it allows us to better commit what is achievable and produce the expected outcome, since we meet every Sprint on the kickoff day. As an outcome we get: - Very predictable velocity. - Better Quality considering we factor in risks and dependencies at the beginning of the PI and Sprint to deliver better outcome. - Kanban board	terms of timeline, cost and Quality. If we sense there is any issues in these then we get enough time to course correct for PPP.	requests from Customer for changing requires which impacts PPP. - Since in aerospace projects we deal with life critical products we needs to be measure Quality very closely. Even minor issue impacts the PPP hugely.

		what we use for Agile execution that also allow us to better track and predict the outcome.		
I&FG#20	Challenges on providing training to all the engineers about Agile and get up to speed on the Agile philosophy.	In short time, regularly delivering piece of working Software to customer instead of waiting long term to deliver complete working software. Cost will be less incase any changes are required to Agile project management.	Better planning, stakeholder management and resource allocation and overall success of project.	SME's leaving or moving out of projects or organization and Not properly maintaining project trainings. Crucial data history need to maintain properly in the projects which runs for a long term..
I&FG#21	Challenges faced by the Team: 1. Spending more time in meetings than the waterfall method. 2. Reaching out to a member from another team has to be requested formally through SM/PO 3. Maintaining the quality while meeting the sprint commitments, as there will be a prefixed notion of 100% on time sprint closure in mind when you are new	1. The Project Management team is able to deliver frequent and smaller delta builds with definitive content, as per the customer demands. 2. Increase in customer satisfaction with the frequent builds and the content that has utmost priority. 3.	1. If the PM Team can predict their Team's performance based on the complexity of the content and the strength of their Team, they can give better estimates to the customer which helps the PM Team to achieve the milestones on time. 2. The	Aerospace projects are very long term projects that span for years, which is typically distributed to multiple Groups. This demands a lot of coordination amongst all the groups that are working on the Project, due to which there is a lot of dependency amongst the groups that are working on the project. Hence, each group should be able to give appropriate commitments and will have a constraint to meet that commitment in order to get the

	to agile. Challenges faced by the Management: 1. Getting the Program Backlog Ready when there is a lot of dynamic content waiting for Customers approval 2. Sometimes, with most of the focus on meeting the sprint commitments, quality might be at stake. 3. Imbibing the thought of 'Agile' into the Engineer's minds, after working on waterfall method for ages.	Sometimes, with most of the focus on meeting the sprint commitments, quality might be at stake. 4. Distributed Knowledge across teams, in contradiction to pre-agile days where the knowledge on a specific domain is limited to a specified team. This helps the PM Team to predict that the work gets done irrespective on a specified team/engineer s availability.	Teams can also ensure that they are not overcommitti ng or undercomittin g to maintain the load balance among engineers. 3. The Project Management team is able to deliver frequent and smaller delta builds with definitive content, as per the customer demands. 4. Increase in customer satisfaction with the frequent builds and the content that has utmost priority.	depending team to continue with their work which is dependent on our deliverable, failing which the costs might go beyond the expectations.
I&FG#22	to create a true cross functional team is still a big challenge, cross team collaboration is another big challenge, specially when there are teams working in different time zones and locations. experienced persons moving to leadership roles	Predictability is good for project performance , but we have to ask the question in a release train how many people can actually predict? APM vastly works on	we have a predict our performance with respect to the kind of resource we are having along with the timeline in sight. Predictable performance helps to plan better for the	ever evolving PRs, Research and development, introduction of new type of work every cycle. estimated work that cannot be complete in 1 sprint cycle or takes complete PI to complete

	(SM,PO,RTE)creates experience vacuum, Agile is good where same work is done again and again, like in operations, there grooming and estimation is almost next to perfect but in R&D world estimation doesn't work as it works in operations. Tailoring Agile only makes it look good from outside, but from inside its not achievable and we have to break stories to achieve our DoD.	Predictability, but it has to be aligned with better estimation and quality of work. if any of these are not proper, the Predictability goes into a toss.	resources in use as well as helps in avoiding any roadblock or bottleneck which may occur in future.	
I&FG#23	Estimating upcoming work	Collaboration, continuous improvement	To gain customer satisfaction	Availability of shared resource, Maintaining Quality
I&FG#24	1.Management didn't provide enough time for the team to fully understand Agile, as the work was being carried out in parallel with the transition. 2.The team is integrating Agile with the previous lifecycle and continuing to work within both frameworks.	Predictability is constrained here because, with a 3-month PI, we can only reliably predict metrics and performance up to that 3-month period. Beyond that timeframe, the ability to predict project performance becomes less certain due to the flexibility of Agile	1. We can shape our team 2. zero commitment miss 3. Work with confidence 4. Defined milestones 5. planned timelines 6. Tracking Capacity 7. Giving correct SPs for work 8. Using previous metrics and forecasting	1. Cost and quality constraints 2. Time and Scope constraints 3. Settled Employee Skills will not improve 4. Changing teams and their trainings 5. Limited Technology as we cannot use new ones 6. Demand may decrease 7. Decisions/Communications will not be on spot

		practices, evolving requirements, and ongoing changes in priorities.	next 9. Reduced risks	
I&FG#25	1.Frequent communication among agile teams 2. Requires lot of co-ordination among the agile teams 3. The exact concept of work is not being carried to the individual team after the work is broken into stories(example of cert findings)	1. Enable agile teams to continuously monitor and evaluate the work progress through regular iterations and feedback loop 2. It gives more transparency and enables the stake holder to gain a clear understanding of project status and potential risks	1. Timely forecasts provide sufficient time to course-correct 2. Increases the achieving targets and reducing outcome variance. 3. We can also check the how team work flow was improved.	1.Dependency on the suppliers and from other stake holders 2.Issues that identifies during the certification 3. Frequent change in the scope of the work
I&FG#26	Major challenge we faced was no harmonized story point definitions across different agile teams. Took a while to get used to how a team estimates their items after getting started with it	Biggest change with Agile Project Management was that individuals who actually work on it also get to voice their concerts, thereby resulting in early risk identification and mitigation strategies. This significantly reduced the	Predictable performance helps in maintaining OTD, helping plan for learning curves, reduces inter team dependencies, reduces stretching at finish lines and many more	Typical long term projects would have been developed on older technologies and hardware's and this results in constantly updating the processes mid cycle, won't be able to adapt new and better technologies until a new project starts, there by increasing the upskilling gap.

		chances of spillovers, increasing predictability		
I&FG#27	There were initial challenges with getting the team to buy into the Agile Project Management approach. The primary concerns were that it would be impossible to break up large tasks into small enough scope to implement with a 2 week sprint. Agile also drives more visibility and accountability into what individuals are working on. Some individuals didn't want their work to be discussed with the team on a daily basis. At the beginning, teams were also sometimes overly optimistic and it led to teams committing to less work instead of finding ways ways to improve.	Agile Project Management has made the project much more predictable and there is greater visibility to issues when they come up. Agile drives more conversations between the teams and stakeholders with a better ability to adapt when issues do come up. While the customer isn't integrated into the Agile framework, there is still improved deliveries to the customer and a more frequent deliveries that improve predictability.	Predictable performance is extremely important to building trust with the customer and providing stability for the development team. Unpredictable teams are consistently starting "fires" that require special attention to resolve issues and the team get in the habit of fighting fires rather than trying to stop them in the first place. Being predictable allows project managers to create realistic estimates and schedules. This builds trust and leads to more business with the customer. Predictable	In aerospace projects, customers consistently face challenges of changing scope and aren't able to predictably share scope beyond the next 2-4 weeks. This is a constraint in project planning and makes it difficult to avoid rework or replanning. Agile and a rolling wave planning approach help mitigate this constraint. The regulatory environment is also consistently changing across avionics through lessons learned and shared information between all companies. This also leads to rework and additional scope that likely wasn't in the initial plan. Long term projects often have constraints between customers and suppliers in terms of layered testing. It often takes a long time for customers to find new issues and flow information back to the development team.

			performance also improves work-life balance and means the team is fully in control of their decisions and development.	
I&FG#28	When agile has the desired effect, it must be a mindset rather than a rulebook. There are many tools and structures that have been tried and tested of ways to implement agile methodologies but there's no one perfect method that will yield perfect project performance every time. Agile has so much more to do with people. To implement agile well, the team and leadership must get to know each other and the work and determine the best tools and structures to fit that specific team. The mindset of agile (continuous improvement, tight coordination and communication, honest feedback, etc) must be internalized by the team and leadership together. It's a culture change that	Smaller increments of planning and retrospection allow the team to provide more accurate estimates and plans and correct concerns more rapidly, keeping them from snowballing into larger problems.	It is only by accurate estimates and performing predictably to those that a project can find the best balance of providing low enough prices to their customers to win business while having high enough prices to cover real costs and expected profit.	An error factor can be compounded over time. If performance is 5% off expectations and that is found and corrected after a few weeks, impact to cost is minor. If performance is 5% off expectations and that's only discovered at the next major project milestone months or years later, the cost impact could be drastic.

	must happen, not a process change. Culture/mindset change is much more challenging and difficult than process change and results are harder to see than the cost of the effort to get there.			
I&FG#29	Dependencies on working parallelly across SDLC tasks will be challenging to meet the sprint deadlines.	Agile Project Management improves predictability of the project performance and helps to foresee the big plan in smaller executable chunks.	Predictable performance creates an improvement scale in terms of both program execution and performance.	Repetition of SDLC tasks from scratch for minor requirement update and having enormous documentation to be done in a manual way is hindering performance.
I&FG#30	i) Vacations(Planned or unplanned) conflict with Sprint deadlines. ii) Planning of story is not predictable. iii) Story Points estimates may not match with work done to NWA.	i) Definitely visibility at all levels increased which is very much helpful to Management ii) Responsibility is clear on engineers	Align with Project needs and Agile commitments Predictability of achievement is eased.	Planned storied on tasks will vary with project hindernesses.
I&FG#31	Productivity decreased, Team members started deciding their own sweet time to complete task, unknown reasons for not doing task on time.	Predictability increased, but cycle time also increased. Engineers are not over burden.	Can estimate future work. See the improvement points. Velocity increases with known issues solving.	Knowledge of engineer. Many faster and smarter ways of doing supporting task KT do not happens. Not able to find talented people when need is there. Schedule is a challenge. If Customer have delays, need to move

				resources to other task and we loose track.
I&FG#32	Understanding and adopting to Agile methodology. Such as, shorter standup meetings(~15min) compared to conventional team meetings(>30min usually) where team gets to discuss about work progress and concerns at length. Where as in Agile, scrum standup meetings are kept short and team have focused discussion. Sometimes, team would be unclear about what are the roles of Scrum Master and Product Owner.	With Agile, Team will be educated about what to be completed in a sprint. Since it becomes convenient to foresee if a task can be completed or not within 2 weeks (i.e., in one sprint), it becomes practical to predict and re-plan the things to improve project performance. Team will have clarity on what to be done, as the tasks are divided into smaller chunks, and that reduces dependency. Development, Verification and other activities can go in parallel.	It is important to have an understanding about a project performance. This will help in performing readjustments in execution - time-wise and budget-wise.	1. One individual playing multiple roles at a time 2. Frequent changes in scope of work 3. Frequent changes in program schedule 4. Not following rules and procedures during program execution. Not doing the right thing. Team sometimes rush to close out peer reviews and that leads to missing out important details in artifacts.

4.5 Thematic Analysis – Common themes from qualitative (I&FG) data

Agile transition introduces structural, cultural, and operational challenges that impact both teams and management. While Agile offers a flexible framework for iterative

development, the transition often reveals gaps in practical implementation. This analysis explores common challenges teams face after transitioning to Agile and presents solutions that enhance adoption and improve project predictability.

One of the primary challenges teams encounters is the ambiguity in roles and responsibilities. Agile assumes that team members inherently understand and fulfill their roles, but gaps often emerge. If the Scrum Master fails to manage the team properly or the Product Owner does not define priorities clearly, misalignment occurs. Additionally, cross-functional collaboration becomes difficult due to expertise gaps, especially in projects requiring the integration of software, hardware, and testing. Another major issue is story sizing and work estimation. Teams often struggle to break down large user stories into manageable increments that fit within a sprint. Inconsistent use of story points across teams further complicates velocity tracking, making project forecasting unreliable. This issue is more pronounced in R&D projects, where work does not fit neatly into Agile sprints, leading to fragmented workflows and unclear progress metrics.

Work prioritization and dependency management also present significant hurdles. Aligning Agile sprints with customer feedback cycles can be challenging, and delayed input often leads to rework. Teams working on multiple products within a release train face frequent context-switching, reducing efficiency. Additionally, Agile's process-heavy nature introduces an increased meeting load, which some teams feel creates more overhead than their previous Waterfall approach. With frequent meetings such as Daily Stand-ups, Sprint Planning, Backlog Grooming, and Retrospectives, available development time gets reduced, impacting productivity.

Cross-team coordination is another critical challenge, especially in complex, multidisciplinary projects involving software, electrical, mechanical, and testing components. Collaboration difficulties are further exacerbated for teams operating in

different time zones. Moreover, predictability and performance measurement become difficult as initial project bids rarely align with actual effort, and teams either overcommit or under commit due to estimation uncertainty. The cultural shift required for Agile adoption also presents resistance, as it demands a transition from individual ownership to collective responsibility. Team members accustomed to long-term Waterfall planning often find Agile's frequent iterations disruptive.

During the early transition phase (typically the first 6–9 months), teams experience execution issues as they fail to follow Agile principles correctly. Planning and retrospectives initially slow down delivery and breaking down large work packages into smaller user stories proves challenging. In some cases, teams prioritize sprint commitments over long-term refinement, leading to quality trade-offs. Additionally, frequent deliveries and swarming activities increase project costs. However, teams that persist with Agile often see improved predictability after three or more years of adaptation.

To mitigate these pain points, organizations must take proactive steps to strengthen Agile roles and responsibilities. Providing targeted training for Scrum Masters, Product Owners, and teams ensures that they are well-equipped to handle Agile execution. Peer reviews and leadership coaching can further improve team facilitation and backlog prioritization. Standardizing story sizing and estimation techniques is crucial for improving consistency across teams. Establishing a unified story point system and using reference stories from past sprints can help teams make more accurate estimations.

Sprint planning and work breakdown strategies also need refinement. Teams should apply the INVEST principles (Independent, Negotiable, Valuable, Estimable, Small, Testable) when defining user stories. A structured work decomposition framework helps in identifying the smallest deliverable unit of work for each sprint. Reducing unnecessary meeting durations through strict timeboxing and agenda-driven discussions can help

optimize Agile processes. Additionally, asynchronous backlog grooming can minimize sprint planning overhead, freeing up more time for development.

For better cross-team coordination, Agile Release Trains (ARTs) should be implemented with clear alignment mechanisms between teams. Collaboration tools such as JIRA, Confluence, and Miro can help track dependencies effectively. To manage external dependencies and work prioritization issues, organizations should ensure early stakeholder involvement to align sprint goals with feedback cycles. Introducing buffer sprints or "hardening sprints" can help absorb customer input delays without disrupting overall timelines.

Driving an Agile mindset and cultural shift is essential for long-term success. Rather than focusing on rigid rule-following, teams should prioritize adaptability. Continuous learning and retrospectives should be emphasized to encourage team-driven improvements. By addressing these challenges, Agile teams can significantly improve predictability, which is critical for project success.

Agile's ability to enhance project predictability comes from its faster feedback loops, which reduce deviations from customer expectations. Velocity tracking and burndown charts provide better short-term forecasting, while incremental delivery ensures earlier value realization. However, Agile does not eliminate unpredictability; it confines it to smaller timescales. Challenges such as high sprint velocity variability, frequent scope changes, and external dependencies still impact long-term forecasting. Despite these issues, Agile offers greater progress visibility and allows for proactive risk mitigation.

Predictability is a key factor in ensuring project success. It enhances stakeholder confidence by providing clear timelines and enables better risk management. Additionally, improved forecasting allows for optimal resource allocation, reducing cost overruns and strengthening customer trust. While Agile does not always result in faster or cheaper

project execution, it provides teams with a realistic view of achievable outcomes, making planning more effective.

Several constraints and dependencies hinder predictable project outcomes. Organizational constraints such as rigid legacy processes and lack of executive buy-in slow down Agile adoption, leading to inconsistent implementation across business units. Technical constraints, including complex system integrations and dependencies on external vendors, introduce uncontrollable delays. Human factors such as knowledge gaps among new Agile adopters and high attrition rates further impact estimation accuracy and execution consistency. Additionally, Agile's inherent limitations in long-term forecasting create challenges, particularly when dealing with unclear initial requirements and frequent scope shifts.

Despite these constraints, Agile provides greater transparency and control, allowing teams to make real-time adjustments. By addressing challenges related to roles, estimation, dependencies, and predictability, organizations can improve Agile adoption and enhance project performance. Standardized estimation techniques, improved planning, better cross-team coordination, and a flexible Agile mindset collectively contribute to making Agile a valuable framework for modern project execution.

4.6 Mapping Thematic Analysis to Research Questions and Hypothesis

Explained in Table 4.2 and Table 4.3:

Table 4.2
Survey Data Mapping

Broader Theme	Research Question (RQ) Addressed	How It Answers the Question	Supporting Hypothesis
Agile Adoption & Cultural Shift	RQ#1: Challenges in Agile Adoption RQ#2: Solutions to Pain Points RQ#3: Impact on Predictability	Resistance to change (47%) and leadership misalignment (34%) hinder adoption. Agile training (38%) and leadership buy-in improve adoption.	**Supports H_1** – Cultural resistance affects predictability.
Role Clarity & Accountability Issues	RQ#1: Challenges in Agile Adoption RQ#2: Solutions to Pain Points RQ#3: Impact on Predictability	Unclear ownership (49%) causes bottlenecks; structured Agile roles (APMO, RTEs) improve execution.	**Supports H_1** – Role ambiguity impacts predictability.
Challenges with Agile Estimation	RQ#1: Challenges in Agile Adoption RQ#2: Solutions to Pain Points RQ#3: Impact on Predictability	Poor estimation (89%) and backlog prioritization (49%) create unpredictability. Monte Carlo simulations improve forecast accuracy.	**Supports H_1** – Estimation issues reduce predictability.
Project Predictability & Velocity Issues	RQ#3: Impact on Predictability RQ#4: Importance of Predictable Performance RQ#5: Constraints & Dependencies	Cross-team dependencies (52%) and sprint misalignment (46%) impact velocity; structured dependency resolution improves forecasting.	**Supports H_1** – Dependencies hinder predictability.
Agile Meetings & Collaboration Challenges	RQ#1: Challenges in Agile Adoption RQ#2: Solutions to Pain Points RQ#3: Impact on Predictability	Inconsistent retrospectives (17%) reduce improvement opportunities. Improved Agile rituals (PI Planning, retrospectives) enhance predictability.	**Supports H_1** – Collaboration affects predictability.

Cross-Team & Cross-Discipline Collaboration	RQ#1: Challenges in Agile Adoption RQ#2: Solutions to Pain Points RQ#3: Impact on Predictability RQ#5: Constraints & Dependencies	Siloed teams (72%) delay execution. Implementing SAFe and structured PI Planning improves cross-team coordination.	**Supports H_1** – Silos reduce predictability.
Customer & External Stakeholder Alignment	RQ#1: Challenges in Agile Adoption RQ#2: Solutions to Pain Points RQ#3: Impact on Predictability RQ#4: Importance of Predictable Performance	Agile flexibility (41%) leads to frequent changes, disrupting long-term planning. Aligning Agile with regulatory compliance improves stability.	**Supports H_1** – Stakeholder misalignment affects predictability.
Workload Management & Stress	RQ#1: Challenges in Agile Adoption RQ#2: Solutions to Pain Points RQ#3: Impact on Predictability RQ#5: Constraints & Dependencies	Frequent scope changes (62%) increase stress and workload instability. Structured backlog refinement helps maintain stable workloads.	**Supports H_1** – Workload instability affects predictability.
Agile Suitability for Different Work Types	RQ#3: Impact on Predictability RQ#4: Importance of Predictable Performance RQ#5: Constraints & Dependencies	Agile is effective for iterative software (36%) but struggles in hardware projects (22%). Hybrid Agile-Waterfall improves predictability for compliance-heavy projects.	**Partially Supports H_1** – Predictability depends on work type.
Leadership & Organizational Support	RQ#1: Challenges in Agile Adoption RQ#2: Solutions to Pain Points RQ#3: Impact on Predictability RQ#4: Importance of Predictable Performance	Leadership misalignment (34%) limits Agile maturity. Strong executive alignment and Agile scaling frameworks improve project predictability.	**Supports H_1** – Leadership alignment is critical for predictability.

Table 4.3
Interview and Focus Group Data Mapping

Broader Theme	Research Question (RQ) Addressed	How It Answers the Question	Supporting Hypothesis
Role Ambiguity & Responsibility Gaps	RQ#1: Challenges in Agile transition	Teams struggle with unclear roles, leading to misalignment and inefficiencies.	Supports H_1 (Agile impacts predictability by influencing team performance).
Estimation & Story Sizing Issues	RQ#1: Challenges in Agile transition	Inconsistent story sizing and velocity tracking hinder accurate forecasting.	Supports H_1 (Poor estimation reduces predictability).
Work Prioritization & Dependencies	RQ#1: Challenges in Agile transition	Misalignment with feedback cycles and high dependency complexity slow progress.	Supports H_1 (Dependencies create unpredictability).
Increased Meeting Overhead	RQ#1: Challenges in Agile transition	Teams feel Agile introduces excessive process overhead, affecting efficiency.	Supports H_1 (Too many meetings reduce execution time).
Cross-Team Coordination Challenges	RQ#1: Challenges in Agile transition	Lack of alignment between teams delays deliverables and affects dependencies.	Supports H_1 (Collaboration issues lead to unpredictability).
Cultural Resistance to Agile	RQ#1: Challenges in Agile transition	Shift from individual to collective ownership causes resistance and slower adoption.	Supports H_1 (Adoption challenges slow down performance gains).
Structured Role Training & Coaching	RQ#2: Solutions to Agile adoption challenges	Training Scrum Masters and Product Owners improves execution and clarity.	Supports H_1 (Better training improves project performance).
Standardized Estimation Practices	RQ#2: Solutions to Agile adoption challenges	Using reference stories and INVEST principles enhances estimation accuracy.	Supports H_1 (Accurate estimation improves predictability).
Optimized Sprint Planning & Meetings	RQ#2: Solutions to Agile adoption challenges	Reducing meeting durations and timeboxing improves efficiency.	Supports H_1 (Less overhead enables better delivery).
Enhanced Dependency Tracking	RQ#2: Solutions to Agile adoption challenges	Agile Release Trains and collaboration tools improve cross-team alignment.	Supports H_1 (Better coordination enhances predictability).

Stakeholder Alignment & Feedback Loops	RQ#2: Solutions to Agile adoption challenges	Early stakeholder involvement ensures work prioritization aligns with business needs.	Supports H_1 (Early feedback improves sprint outcomes).
Agile's Faster Feedback Loops	RQ#3: Agile's impact on project predictability	Frequent iterations reduce uncertainty and allow quick course corrections.	Supports H_1 (Shorter cycles lead to better predictability).
Velocity & Burndown Chart Tracking	RQ#3: Agile's impact on project predictability	Real-time progress tracking enables more accurate forecasting.	Supports H_1 (Data-driven metrics improve predictability).
Limited Long-Term Forecasting	RQ#3: Agile's impact on project predictability	Agile confines unpredictability to smaller timeframes rather than eliminating it.	Supports H_0 (Agile does not eliminate unpredictability completely).
Customer Confidence & Risk Management	RQ#4: Importance of predictable performance	Predictable execution builds trust and reduces cost overruns.	Supports H_1 (Predictability is a key success factor).
Resource Optimization & Cost Control	RQ#4: Importance of predictable performance	Improved forecasting ensures better allocation of workforce and budget.	Supports H_1 (Better planning leads to optimized resource use).
Legacy Processes & Organizational Constraints	RQ#5: Constraints affecting predictability	Rigid traditional processes slow Agile adoption and create inconsistencies.	Supports H_1 (Legacy systems reduce agility).
Technical Constraints & External Dependencies	RQ#5: Constraints affecting predictability	Vendor dependencies and integration complexities introduce uncontrollable delays.	Supports H_1 (External factors reduce Agile predictability).
Human Factors & Knowledge Gaps	RQ#5: Constraints affecting predictability	High attrition and insufficient Agile training lower estimation accuracy.	Supports H_1 (Team experience impacts Agile success).
Frequent Scope Changes	RQ#5: Constraints affecting predictability	Shifting priorities disrupt sprint commitments and project timelines.	Supports H_0 (Agile does not prevent scope creep).

4.7 Final Hypothesis Analysis

The thematic analysis from, survey data, interviews, and focus groups strongly support the Alternative Hypothesis (H_1): Transitioning to Agile Project Management significantly impacts the predictability of project performance in aerospace projects. Across all data sources, a clear pattern emerged—Agile adoption introduces both challenges and benefits that affect project predictability. While Agile creates estimation difficulties, backlog instability, and dependency risks, it also strengthens risk identification, sprint planning, defect tracking, and real-time monitoring. Participants emphasized that structured Agile scaling frameworks such as SAFe, along with effective dependency management and Monte Carlo forecasting, help mitigate these challenges and enhance predictability.

Given this evidence, the Null Hypothesis (H_0) is rejected, as Agile adoption demonstrably impacts project predictability. However, its effectiveness is influenced by several key factors, including estimation accuracy, leadership support, cross-functional collaboration, and the suitability of Agile for different project types. Thematic analysis of qualitative responses revealed that Agile's iterative approach fosters frequent feedback loops, enhances stakeholder alignment, and enables proactive risk mitigation. Survey results highlighted those Agile practices such as retrospectives, burndown charts, and velocity tracking improve decision-making, leading to greater predictability in project execution. Additionally, standard estimation techniques—including reference stories, INVEST principles, and historical velocity data—were frequently mentioned as essential tools for refining sprint planning and reducing variability in delivery timelines. Furthermore, Agile's ability to prioritize high-value work, optimize resource allocation, and maintain financial control contributes to improved cost predictability, as noted in multiple focus group discussions.

However, limitations were also identified in survey responses and interview discussions, where respondents pointed out that Agile does not completely eliminate unpredictability. Scope creep and evolving regulatory requirements remain persistent challenges, particularly in aerospace projects with long development cycles and stringent compliance needs. External dependencies—such as supplier delays, integration complexities, and regulatory approvals—introduce risks beyond Agile's control. Long-term forecasting emerged as a major concern, with focus group participants noting that while Agile enhances short-term adaptability, it may not always provide accurate multi-year forecasts, which are crucial for strategic aerospace planning. Additionally, human factors such as high attrition rates, resistance to change, and gaps in Agile expertise were frequently cited as contributors to execution variability.

Conclusion

Based on the thematic analysis, survey data, interviews, and focus groups, it is evident that Agile significantly impacts project predictability, though its effectiveness depends on organizational maturity, implementation strategy, and project complexity. Agile provides data-driven decision-making tools, real-time tracking mechanisms, and structured risk management practices, which contribute to greater predictability. However, external dependencies, long-term forecasting challenges, and human factors introduce unpredictability that Agile alone cannot fully address. Therefore, while Agile adoption improves predictability in aerospace projects, complementary strategies, such as advanced risk management, enhanced cross-team coordination, and predictive analytics, are necessary to achieve consistent long-term forecasting and project stability.

4.8 Proposed Solution to address Literature Gap for Agile Value and Principle

Gap#1: Agile methodologies have transformed project management by emphasizing flexibility, collaboration, and customer responsiveness. However, a critical challenge remains—predictability. While Agile promotes adaptability, excessive uncertainty can lead to inefficiencies, missed commitments, and difficulties in risk management. The literature review has already identified this gap, highlighting that while Agile enhances responsiveness, it often lacks mechanisms to ensure reliable forecasting, risk mitigation, and steady progress. This gap is particularly evident in large-scale Agile implementations, where balancing flexibility with organizational constraints becomes a challenge. Addressing this requires an additional Agile value: **Predictability Over Uncertainty**, refer Figure 4.1. Predictability in Agile does not mean rigid adherence to fixed plans but rather the ability to forecast outcomes, manage risks, and maintain steady progress without sacrificing adaptability.

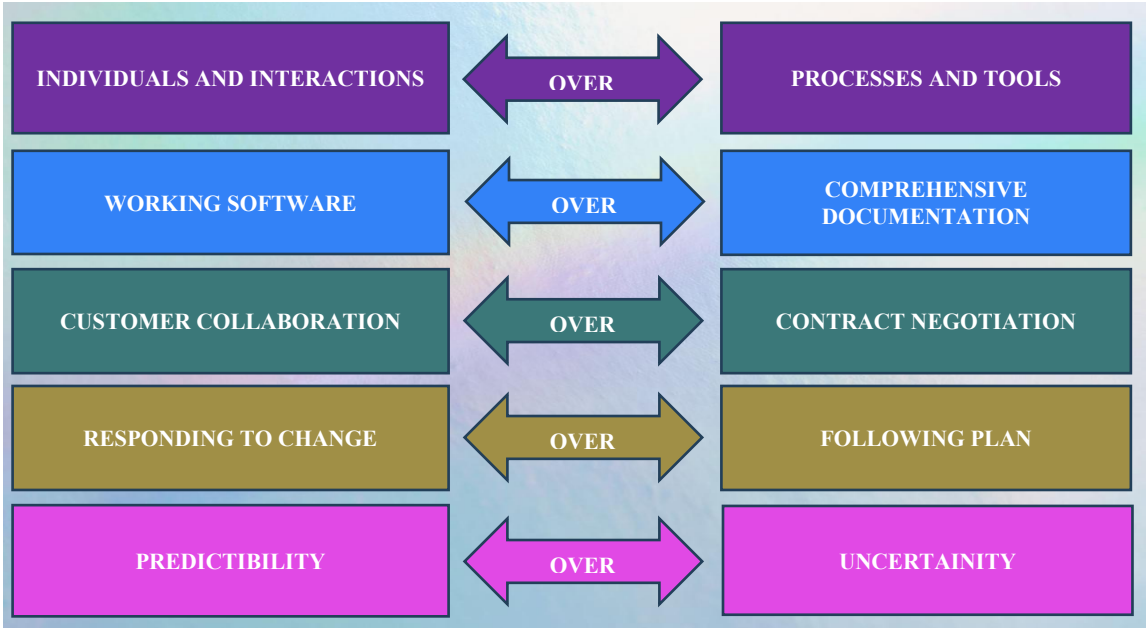


Figure 4.1
Additional Agile value (Predictability Over Uncertainty)

The inclusion of Predictability as a Fifth Agile Value is essential because it ensures that agility does not devolve into chaos while preserving transparency, reliability, and stakeholder confidence. It complements Agile's core principles by maintaining customer focus while providing a stable foundation for risk-aware decision-making. Furthermore, it aligns with the need for data-driven forecasting and sustainable delivery practices, addressing the gap identified in Agile at scale. By embedding predictability into Agile methodologies, organizations can achieve optimized project delivery, improved risk management, and enhanced stakeholder trust, strengthening Agile's effectiveness in complex and dynamic environments like Aerospace organization.

Gap#2: Agile principles emphasize adaptability, customer collaboration, and frequent delivery, but they do not explicitly address predictability in delivery cadence. While Agile encourages flexibility and responsiveness, teams often struggle with balancing adaptability with consistent and reliable delivery expectations. This gap can lead to unpredictable workflows, missed commitments, and stakeholder uncertainty—especially in complex, large-scale projects where dependencies and risks must be managed effectively.

The 12th Agile Principle—which states that teams should "reflect on how to become more effective and adjust behavior accordingly"—promotes continuous improvement but lacks an explicit focus on flow efficiency and predictability. Reflection alone does not ensure that teams actively optimize their processes to enhance stable throughput and delivery confidence. Without a structured emphasis on predictability, teams may focus only on short-term adaptability while overlooking long-term process stability.

Proposed Solution: Enhancing the 12th Agile Principle

To bridge this gap, study propose expanding the 12th Agile Principle to incorporate efficiency and predictability while preserving its original intent. The revised principle is as follows:

"At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly, improving efficiency, enhancing predictability, and sustaining consistent delivery while retaining Agile flexibility."

This revision ensures that teams not only adapt to change but also proactively optimize workflows to maintain a consistent, predictable cadence.

4.9 Predictive Agile Delivery Framework

Introduction

The Predictive Agile Delivery Framework (PADF) is a structured approach that integrates Agile methodologies with DevSecOps to enhance predictability, security, and delivery efficiency in aerospace projects. This framework is designed to address compliance constraints, risk management, and cross-team coordination while ensuring continuous integration, deployment, and security validation.

Unlike traditional Agile implementations that focus primarily on incremental delivery, PADF emphasizes:

- Backlog prioritization, story estimation and risk & dependency assessment as key drivers for sprint execution.
- Security and compliance integration throughout the Agile lifecycle.
- Cross-functional team alignment using Scaled Agile Framework (SAFe) while maintaining both speed and quality.

By embedding predictive planning, security enforcement, and automated compliance validation within Agile processes, PADF enables aerospace organizations to optimize project delivery without compromising safety, regulatory adherence, or system reliability.

The Three Pillars of Predictive Agile Delivery Framework (PADF)

PADF is built on three foundational pillars, each designed to optimize Agile execution, enhance security, and improve predictability. These pillars serve as the guiding principles for seamless project delivery in aerospace engineering, refer Figure 4.2:

1. Velocity Navigator (Agile Execution for Iterative Development) – This pillar ensures incremental, high-quality software delivery through well-structured Agile execution. It leverages Scrum, SAFe, and hybrid Agile models tailored for aerospace projects, ensuring teams work collaboratively while maintaining regulatory compliance.
2. Fortified Pipeline (DevSecOps for Continuous Integration, Security, and Compliance) – Security and compliance are seamlessly embedded into the development and deployment processes. Automated testing, security validation, and regulatory audits are integrated into every stage to ensure a robust software pipeline.
3. Precision Compass (Predictability through Risk-Based Prioritization) – This pillar enhances predictability by focusing on risk-based backlog prioritization along with value based. Teams are guided by data-driven decision-making, ensuring that high-impact, low-risk deliverables are tackled first, improving sprint accuracy and project timelines.

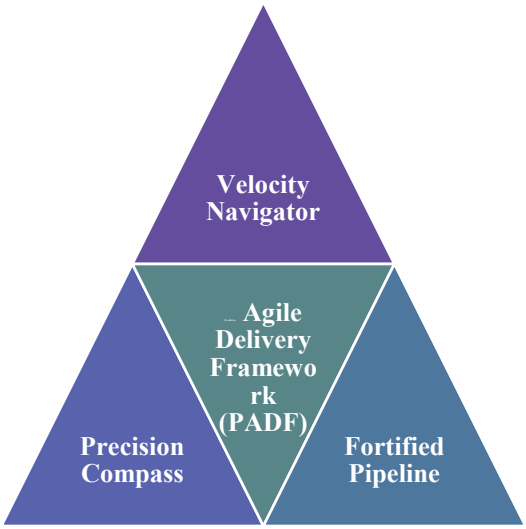


Figure 4.2
Three Pillars of Predictive Agile Delivery Framework (PADF)

PADF Process Flow

The Predictive Agile Delivery Framework (PADF) consists of six key phases (as described in Figure 4.3) that ensure a seamless, secure, and predictable delivery pipeline. Each phase plays a crucial role in structuring the Agile execution process, embedding compliance, and enhancing project predictability. The six phases include Agile Planning & Risk-Based Prioritization, Secure Agile Development & Integration, Continuous Integration & Compliance Validation, Continuous Deployment & System Integration, Continuous Monitoring & Security Enforcement, and Agile Feedback Loops & Predictability Metrics.

Each phase consists of a set of key activities, inputs, outputs, and expected outcomes, refer Table 4.4 ‘Predictive Agile Delivery Framework (PADF) Phases’.

Table 4.4
Predictive Agile Delivery Framework (PADF) Phases

Phase	Key Activities	Inputs	Outputs	Outcomes
Phase 1: Agile Planning & Risk-Based Prioritization	Conduct backlog refinement with risk assessment using Weighted Shortest Job First (WSJF). Align user stories with compliance requirements and system architecture. Perform security and threat modeling.	Product backlog, story estimation, risk and dependency assessment data, compliance standards.	Risk-prioritized backlog, sprint goals with dependency resolution.	Well-defined backlog with security and compliance risks addressed before development begins.
Phase 2: Secure Agile Development & Integration	Implement secure coding practices, conduct code reviews, automate unit testing and static analysis, ensure collaboration between developers, testers, and security teams.	Risk-prioritized backlog, secure coding guidelines, automated testing tools.	Security-validated code, early identification of risks and vulnerabilities.	Reduction in security defects, improved software reliability.
Phase 3: Continuous Integration & Compliance Validation	Automate build verification testing (BVT), conduct compliance validation against FAA, EASA, DO-178C, maintain immutable audit trails.	Code repository, regulatory requirements, test automation tools.	Incremental regulatory sign-offs, audit-ready compliance logs.	Reduced certification delays, improved compliance tracking.
Phase 4: Continuous	Deploy incremental	Security-validated code,	Feature validation	Faster software validation

Deployment & System Integration	software builds in controlled environments, perform automated system integration testing, enforce zero-trust security policies.	test environment, deployment automation tools.	reports, integrated software components.	cycles, minimized integration failures.
Phase 5: Continuous Monitoring & Security Enforcement	Implement real-time anomaly detection, establish rollback mechanisms, maintain compliance logs.	Deployed system, security monitoring tools, audit logs.	Incident detection reports, automated rollback logs.	Proactive risk mitigation, improved system performance stability.
Phase 6: Agile Feedback Loops & Predictability Metrics	Conduct retrospectives, track Agile predictability metrics, refine backlog based on feedback.	Sprint reports, defect data, velocity metrics.	Optimized backlog, refined Agile processes.	Improved Agile execution, enhanced predictability in project delivery.

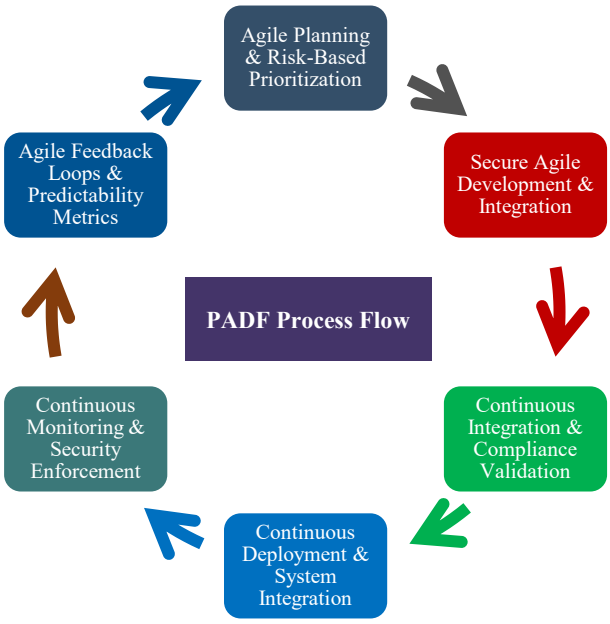


Figure 4.3
Predictive Agile Delivery Framework (PADF) phases

Phase 1: Agile Planning & Risk-Based Prioritization-Table 4.5

Table 4.5
Agile Planning & Risk-Based Prioritization

Key Activities	Inputs	Outputs	Outcomes
Conduct backlog refinement with risk assessment using Weighted Shortest Job First (WSJF)	Product backlog, story estimation, risk and dependency assessment data, compliance standards	Risk-prioritized backlog	Well-defined backlog with security and compliance risks addressed
Align user stories with compliance requirements and system architecture	Agile release plans, compliance guidelines	Sprint goals with dependency resolution	Reduced rework and better regulatory alignment
Perform security and threat modeling	Security guidelines, threat assessment tools	Threat-mitigated backlog	Early risk identification and mitigation

Phase 2: Secure Agile Development & Integration-Table 4.6

Table 4.6
Secure Agile Development & Integration

Key Activities	Inputs	Outputs	Outcomes
Implement secure coding practices and conduct static analysis	Secure coding guidelines, automation tools	Security-validated code	Reduction in security defects
Automate unit testing and peer reviews	Test cases, coding standards	Early identification of defects	Improved software reliability
Collaborate between developers, testers, and security teams	Agile team feedback, CI/CD tools	Security-enhanced Agile workflows	Minimized vulnerabilities

Phase 3: Continuous Integration & Compliance Validation-Table 4.7

Table 4.7
Continuous Integration & Compliance Validation

Key Activities	Inputs	Outputs	Outcomes
Automate build verification testing (BVT)	Code repository, test automation tools	Incremental regulatory sign-offs	Reduced certification delays
Maintain immutable audit trails for compliance	Regulatory requirements, compliance logs	Audit-ready compliance reports	Improved compliance tracking
Conduct security validation at every CI/CD stage	Security policies, testing tools	Early vulnerability detection	Secure software releases

Phase 4: Continuous Deployment & System Integration-Table 4.8

Table 4.8
Continuous Deployment & System Integration

Key Activities	Inputs	Outputs	Outcomes
Deploy incremental software builds in controlled environments	Security-validated code, deployment automation tools	Feature validation reports	Faster validation cycles
Perform automated system integration testing	Test plans, compliance reports	Integrated software components	Minimized integration failures
Enforce zero-trust security policies	Security frameworks, DevSecOps tools	Hardened deployment environments	Increased system resilience

Phase 5: Continuous Monitoring & Security Enforcement-Table 4.9

Table 4.9
Continuous Monitoring & Security Enforcement

Key Activities	Inputs	Outputs	Outcomes
Implement real-time anomaly detection	Deployed system, monitoring tools	Incident detection reports	Proactive risk mitigation
Establish rollback mechanisms for failed deployments	System logs, rollback plans	Automated rollback logs	Improved system stability
Maintain compliance logs for audits	Security monitoring tools, compliance checklists	Continuous compliance validation	Sustained regulatory adherence

Phase 6: Agile Feedback Loops & Predictability Metrics-Table 4.10

Table 4.10
Agile Feedback Loops & Predictability Metrics

Key Activities	Inputs	Outputs	Outcomes
Conduct retrospectives to analyze sprint execution	Sprint reports, defect data	Optimized backlog	Improved Agile execution
Track Agile predictability metrics	Velocity reports, risk assessment	Refined Agile processes	Enhanced predictability in delivery
Adjust backlog prioritization based on feedback	User feedback, sprint analysis	Updated backlog	Greater responsiveness to project needs

Implementation Strategy for PADF Adoption in Aerospace

For successful adoption of PADF, a structured step-by-step implementation strategy is required. Each step is creatively named to reflect its core purpose:

1. Blueprint Alignment (Agile & DevSecOps Integration Strategy) – Define how Agile sprints map to DevSecOps pipelines and compliance checkpoints. Establish risk-based backlog prioritization methods to focus on high-impact items.
2. Compliance Guardian (Automated Compliance & Security Validation) – Embed automated compliance and security validation into CI/CD pipelines. Introduce traceability matrices linking Agile work items with regulatory artifacts.
3. Synchronized Momentum (Scaling Agile Across Engineering Domains) – Implement SAFe Agile Release Trains (ARTs) to enable multi-team collaboration. Conduct cross-functional sprint planning to align software, hardware, and compliance teams.

4. Governance Sentinel (Agile Governance & Predictability Tracking) – Deploy enterprise dashboards to track velocity, backlog health, and risk exposure. Conduct quarterly Agile audits to identify execution gaps and areas for improvement.
5. Evolutionary Loop (Continuous Improvement & Agile Maturity Enhancement) – Train teams on DevSecOps best practices, refine backlog prioritization based on sprint performance, and establish long-term feedback loops for iterative optimization.

CHAPTER V:

DISCUSSION

5.1 Discussion of Results

The results of this study highlight significant challenges and opportunities in implementing Agile methodologies in aerospace projects. While Agile is widely recognized for its flexibility and iterative development approach, its adoption in the aerospace sector presents specific hurdles related to predictability, dependency management, compliance, leadership alignment, and cost forecasting. The findings indicate that Agile methodologies, in their current form, struggle to meet the rigorous demands of highly regulated aerospace environments, leading to inefficiencies in project execution and delivery.

One of the key findings of this research is that predictability remains a major challenge, with 67% of respondents stating that backlog instability and estimation errors frequently lead to delays and budget overruns. The inherent uncertainty in Agile estimations, particularly in complex aerospace projects, contributes to inconsistent sprint performance and fluctuating project timelines. Additionally, dependency management emerged as a major bottleneck, with 72% of participants emphasizing the difficulty of coordinating cross-functional teams and interdependencies in large-scale Agile projects. Traditional Agile frameworks such as Scrum and SAFe do not fully address these dependencies, resulting in inefficiencies and rework.

Furthermore, compliance and regulatory alignment pose significant barriers to Agile adoption in aerospace. Nearly 45% of respondents indicated that Agile teams struggle to meet regulatory requirements such as DO-178C, FAA, and EASA standards. Agile's emphasis on flexibility often conflicts with the structured and documentation-

heavy approach required by aviation regulators. Leadership alignment was also identified as a concern, with 34% of respondents stating that Agile transformation efforts often lack executive sponsorship, leading to fragmented implementation across departments. Lastly, cost predictability emerged as a key challenge, with 21% of respondents highlighting budget overruns due to scope creep, poor estimation techniques, and unplanned rework.

These findings indicate that while Agile can offer significant benefits, its implementation in aerospace projects must be carefully structured. The discussion that follows provides an in-depth analysis of each research question based on these findings.

5.2 Discussion of Research Questions

Agile predictability is a critical factor influencing project performance in the aerospace industry. The study found that a lack of predictability in Agile teams often leads to missed deadlines, budget overruns, and increased risk exposure. One of the primary reasons for this unpredictability is backlog instability, with 89% of participants indicating that frequent changes in backlog priorities disrupt sprint planning and execution. Additionally, inaccurate estimation techniques lead to discrepancies between planned and actual work completed, further exacerbating project delays.

The discussion highlights that improving Agile predictability requires adopting historical data-driven forecasting techniques that leverage historical data and real-time project metrics to refine backlog estimations. Monte Carlo simulations, for instance, can provide probabilistic predictions of project timelines, helping teams anticipate risks before they escalate. Moreover, implementing structured Agile estimation frameworks, such as the SAFe Predictability Model, can improve backlog stability by reducing mid-sprint scope changes.

To address these issues, organizations should incorporate dependency mapping and risk management frameworks to identify potential bottlenecks early in the project lifecycle. This will help teams proactively manage risks and reduce unexpected delays. Adopting structured Agile estimation models and model-based project forecasting tools is recommended to enhance predictability and improve overall project performance in aerospace environments.

Dependency management is another most significant challenges faced by Agile teams in aerospace projects. The study found that 72% of respondents struggle with inter-team dependencies, which frequently lead to delays and inefficiencies. Large-scale Agile frameworks, such as SAFe and LeSS, provide some mechanisms for dependency management, but they often fall short in addressing the complexity of aerospace projects, where multiple teams work on interrelated subsystems.

A critical discussion point is the need for an Agile Program Management Team (APMT) that can oversee cross-team coordination and ensure alignment between various workstreams. The APMO would act as a central body responsible for identifying dependencies, resolving conflicts, and ensuring smooth collaboration between teams. Additionally, integrated Agile planning techniques such as multi-team PI planning can help teams proactively identify and address dependencies during early planning stages.

Another promising approach is the use of Digital Twin and historical data-based models for dependency resolution. Model-driven dependency mapping tools can analyze project workflows and suggest optimal dependency resolutions, reducing the risk of bottlenecks. By implementing these solutions, aerospace organizations can significantly improve Agile dependency management, leading to smoother project execution, reduced rework, and enhanced team collaboration.

Compliance with aviation regulations is a major concern for Agile teams in aerospace projects. The study found that 45% of respondents experience challenges aligning Agile practices with regulatory requirements, such as DO-178C, FAA, and EASA standards. The primary issue is that Agile promotes flexibility and iterative development, whereas regulatory standards require structured processes, extensive documentation, and strict validation protocols.

The discussion emphasizes the need for compliance-aware Agile frameworks that integrate regulatory requirements without disrupting Agile workflows. One approach is to implement automated documentation and auditing tools that generate compliance artifacts in real time, ensuring that Agile teams meet regulatory obligations without excessive manual effort. Additionally, incremental certification approaches, where regulatory authorities are engaged throughout the Agile development process, can help streamline compliance without disrupting iterative workflows.

A key recommendation is the adoption of Agile-compliant regulatory frameworks, such as SAFe for regulated industries, which incorporates compliance requirements into the Agile process. Furthermore, integrating continuous verification and validation (V&V) pipelines can help ensure that Agile deliverables meet regulatory standards without requiring excessive rework. Organizations should also invest in Regulatory Liaisons within Agile teams to ensure that compliance considerations are addressed early in the development lifecycle.

By implementing these solutions, aerospace companies can reduce compliance-related roadblocks while maintaining the flexibility of Agile. This will enable faster product development cycles while ensuring that safety and regulatory standards are upheld.

Leadership alignment plays a crucial role in the success of Agile transformation efforts within aerospace organizations. The study found that 34% of respondents believe a

lack of executive sponsorship is a major barrier to Agile adoption. Without strong leadership support, Agile initiatives often fail to gain traction, leading to fragmented implementation and resistance from teams.

A key discussion point is that Agile transformation requires a top-down and bottom-up approach where executive leadership provides clear strategic direction, while teams have the autonomy to execute Agile practices effectively. Leadership training programs focused on Agile principles can help executives understand how Agile fits into the broader organizational strategy. Additionally, Agile Transformation Teams (ATT) can facilitate alignment between leadership and Agile teams by bridging communication gaps and ensuring consistent implementation across departments.

Organizations should also focus on establishing clear success metrics for Agile transformation, such as improved time-to-market, defect reduction, and increased employee engagement. By demonstrating the tangible benefits of Agile, leadership buy-in can be strengthened, leading to more successful and sustained Agile adoption.

Cost predictability remains a significant challenge for Agile teams in aerospace projects, with 21% of respondents citing budget overruns as a frequent issue. The root cause of this challenge lies in poor estimation techniques, scope creep, and unexpected rework due to changing requirements.

A major discussion point is the need for structured financial forecasting models that align with Agile development cycles. Traditional cost estimation methods are not well-suited for Agile, where work is planned in short iterations. Instead, data-driven cost prediction models, which analyze historical project data and real-time spending patterns, can provide more accurate financial forecasts.

Additionally, organizations should adopt Earned Value Management (EVM) for Agile, a hybrid approach that combines Agile metrics with traditional cost-tracking

mechanisms. This allows project managers to monitor budget consumption in real time and take corrective actions before overruns occur. Feature-based funding models, where budgets are allocated based on deliverable value rather than rigid upfront planning, can also help improve cost predictability.

By leveraging data-driven financial forecasting and hybrid cost-tracking approaches, aerospace organizations can enhance budget predictability, leading to more efficient resource allocation and reduced financial risks.

CHAPTER VI:

SUMMARY, IMPLICATIONS, AND RECOMMENDATIONS

6.1 Summary

The study identified several critical factors affecting Agile adoption in aerospace projects, including predictability challenges, dependency management issues, compliance barriers, leadership alignment, and cost unpredictability. The findings highlight that while Agile offers significant benefits, such as improved collaboration and faster development cycles, its implementation in aerospace requires structural modifications to address predictability, regulatory and complexity-related challenges.

To enhance Agile predictability, data-driven forecasting models and structured estimation frameworks should be adopted. Multi-team dependency management through Agile Program Management Team (APMT) and Model-based dependency mapping can help resolve bottlenecks. Compliance barriers can be mitigated through automated documentation tools and continuous verification pipelines. Leadership alignment can be improved with Agile training for executives and clear success metrics, while hybrid cost-tracking methods using Earn value Management Systems can enhance financial predictability.

These findings provide actionable insights for aerospace organizations looking to scale Agile effectively while addressing industry-specific constraints.

6.2 Implications

The implications of this study extend to both academic research and industry practice. From an academic perspective, the findings contribute to the growing body of knowledge on Agile implementation in regulated industries, specifically in aerospace. This

research provides empirical evidence on the challenges and solutions for Agile predictability, dependency management, compliance, leadership alignment, and cost forecasting, which can serve as a foundation for future studies.

For industry practitioners, the study offers practical recommendations to improve Agile adoption in aerospace. The integration of data-driven predictive analytics, automated compliance tools, structured dependency management frameworks, and leadership engagement strategies can significantly enhance project performance. Additionally, the findings suggest that aerospace companies should invest in hybrid Agile models that blend Agile flexibility with structured governance mechanisms to balance innovation with regulatory compliance.

Overall, this study highlights the need for a tailored approach to Agile in aerospace, ensuring that Agile principles are adapted to fit the industry's unique constraints while maximizing their benefits.

6.3 Recommendations for Future Research

Based on the study's findings, few key recommendations are proposed:

1. Adopt data-driven Agile forecasting tools to improve backlog estimation, predict project timelines, and enhance cost predictability.
2. Establish an Agile Program Management Teams (APMT) to oversee dependency management and ensure seamless cross-team coordination.
3. Implement automated compliance tools that generate regulatory documentation in real time to align Agile workflows with industry standards.

4. Enhance leadership engagement through Agile training programs and the establishment of Agile Transformation Teams (ATT) to bridge the gap between strategy and execution.
5. Leverage hybrid financial tracking models, such as Earned Value Management (EVM) for Agile, to improve cost control and budget predictability.
6. Utilize multi-team PI planning and digital twin models to improve dependency resolution and reduce bottlenecks in large-scale Agile projects.
7. Develop a tailored Agile framework (Predictive Agile Delivery Framework) for aerospace that incorporates structured governance while maintaining Agile flexibility.

These recommendations will enable aerospace organizations to effectively navigate Agile transformation challenges, improve project predictability, and ensure compliance with regulatory standards while maintaining efficiency.

6.4 Conclusion

This study explored the challenges and solutions for Agile predictability, dependency management, compliance, leadership alignment, and cost forecasting in aerospace projects. The findings indicate that while Agile provides significant advantages in terms of flexibility and iterative development, its application in aerospace requires modifications to address regulatory and complexity constraints.

By integrating data-driven forecasting, structured dependency management, automated compliance tools, and leadership engagement strategies, aerospace organizations can overcome existing barriers and enhance Agile adoption. The research

emphasizes the importance of balancing Agile flexibility with structured governance models to ensure project success in highly regulated environments.

Ultimately, the study concludes that Agile can be successfully scaled in aerospace projects if it is adapted to industry-specific challenges. The recommended approaches provide a roadmap for aerospace companies seeking to improve project performance, enhance predictability, and optimize delivery timelines while ensuring compliance with safety and regulatory requirements. Future research should focus on refining hybrid Agile models and exploring data-driven automation techniques to further enhance Agile efficiency in aerospace.

APPENDIX A

SURVEY COVER LETTER

Hello Everyone,

I hope you all are doing great. I am conducting a survey on "Impact of Agile Approach on Project Predictability and Performance in Aerospace Project" to undersatand current situation and propose solution based on the data to enhance outcome of project with improved predictibilty.

Your insights are invaluable in helping me to exmine current Agile practices and find the gaps in current practices. The survey will take approximately 30 – 40 minutes to complete, and all responses will remain confidential and anonymous.

Please complete the survey by 25th December 2024 using the Survey Link.

Your participation is voluntary, but I sincerely appreciate your time and input. If you have any questions, feel free to contact me at sunaina@ssbm.ch.

Thank you for your support!

APPENDIX B

SURVEY QUESTIONS

Impact of Agile Approach on Project Predictability and Performance in Aerospace Projects

Section 1

Participant Information

1. What is your Role in the Organization? *

- ◯ Senior Leader
- ◯ Manager
- ◯ Agile Coach
- ◯ Release Train Engineer (RTE)
- ◯ Product Owner (PO)
- ◯ Scrum Master (SM)
- ◯ Team Member (Developer/Self Organized Team)
- ◯ Other

2. How many years of experience you have in the organization? *

- ◯ 0-3
- ◯ 4-8
- ◯ 9-12
- ◯ 13+

3. What is the Agile maturity level of your Organization? *

- ◯ Beginner (New to Agile)
- ◯ Intermediate (Some Agile adoption)
- ◯ Advanced (Fully Agile organization)

4. What is the size of your typical project team? *

- ◯ 1–5 members
- ◯ 6–10 members
- ◯ 11–15 members
- ◯ More than 15 members

5. How complex are the projects you work on? *

- ◯ Low complexity
- ◯ Medium complexity
- ◯ High complexity
- ◯ Very high complexity

6. What level of involvement do you have in Agile decision-making? *(Scale: 1 = No Involvement, 5 = Highly Involved)* *

☆ ☆ ☆ ☆ ☆

7. How long has your team been practicing Agile? *

- ◯ Less than 1 year
- ◯ 1–3 years
- ◯ 3–5 years
- ◯ More than 5 years

Section 2

Agile Frameworks & Methodologies Used

8. Which Agile frameworks/methodologies does your organization follow? *(Select all that apply)* *

- [ ] Scrum
- [ ] Kanban
- [ ] SAFe (Scaled Agile Framework)
- [ ] LeSS (Large Scale Scrum)
- [ ] Disciplined Agile (DA)
- [ ] Hybrid (Agile + Waterfall)
- [ ] Other

9. If your organization uses a hybrid model, what percentage of project management follows Agile vs. Traditional methods? *

- ( ) Mostly Agile (80% Agile, 20% Traditional)
- ( ) Balanced (50% Agile, 50% Traditional)
- ( ) Mostly Traditional (20% Agile, 80% Traditional)

Section 3

Agile Training & Certifications

10. Have you received formal Agile training or certifications? *(Select all that apply)* *

- [ ] Agile 101
- [ ] Certified Scrum Master (CSM)
- [ ] Certified Product Owner (CPO)
- [ ] SAFe Agilist (SA) / SAFe Program Consultant (SPC)
- [ ] PMI-ACP (Agile Certified Practitioner)
- [ ] None

11. How confident are you in your Agile knowledge and application? *(Scale: 1 = Not Confident, 5 = Highly Confident)* *

☆ ☆ ☆ ☆ ☆

Section 4

Organizational Support for Agile

12. Does your organization provide formal Agile training and mentorship? *

- ◯ Yes, extensive training is provided
- ◯ Some training, but mostly self-learning
- ◯ No formal training, learning is ad-hoc
- ◯ No Agile training at all

13. Does leadership actively support Agile transformation? *(Scale: 1 = Not Supportive, 5 = Highly Supportive)* *

☆ ☆ ☆ ☆ ☆

14. What leadership behaviors have the most impact on Agile adoption? *(Select all that apply)* *

- ☐ Encouraging experimentation & iterative improvements
- ☐ Empowering teams to make decisions
- ☐ Setting clear priorities & business goals
- ☐ Removing organizational blockers to Agile implementation

15. What organizational barriers make Agile adoption difficult? *(Select all that apply)* *

- ☐ Resistance to change from leadership
- ☐ Lack of standardized Agile processes across teams
- ☐ Organizational silos & conflicting priorities
- ☐ Inflexibility due to compliance/regulatory needs

Section 5

Practical Application of Agile Training

16. Have you been able to apply Agile training in your daily tasks? *

- ◯ Yes, I apply Agile principles effectively in my work.
- ◯ Partially, I use Agile concepts but struggle with full implementation.
- ◯ No, Agile training is not relevant to my daily tasks.
- ◯ I have not received Agile training.

17. What barriers prevent you from applying Agile training in your tasks? *(Select all that apply)* *

- ☐ Lack of team alignment on Agile practices
- ☐ Conflicts with existing company processes
- ☐ Insufficient leadership support for Agile
- ☐ Agile training did not provide practical implementation guidance
- ☐ High regulatory/compliance requirements make Agile difficult
- ☐ Time constraints and workload pressure

18. How frequently do you use Agile techniques such as stand-ups in your role? *

- ◯ Daily
- ◯ Weekly
- ◯ Bi-weekly
- ◯ Monthly
- ◯ Rarely
- ◯ Never

19. How frequently do you use Agile techniques such as backlog grooming in your role? *

- ◯ Daily
- ◯ Weekly
- ◯ Bi-Weekly
- ◯ Monthly
- ◯ Rarely
- ◯ As needed
- ◯ Never

20. How frequently do you use Agile techniques such as sprint planning in your role? *

- ◯ Daily
- ◯ Weekly
- ◯ Bi-Weekly
- ◯ Monthly
- ◯ Rarely
- ◯ Never

21. How frequently do you use Agile techniques such as retrospectives in your role? *

- ◯ Daily
- ◯ Weekly
- ◯ Bi-Weekly
- ◯ Monthly
- ◯ Rarely
- ◯ Never

22. Which Agile concepts are easiest for you to apply in your tasks? *(Select all that apply)* *

- [ ] Iterative development & continuous delivery
- [ ] Self-organizing teams & collaboration
- [ ] Sprint planning & backlog prioritization
- [ ] Retrospectives & continuous improvement
- [ ] Agile metrics & predictability tracking

23. Which Agile concepts are hardest for you to apply in your tasks? *(Select all that apply)* *

- [ ] Estimating and forecasting work (story points, velocity)
- [ ] Handling unplanned work in an Agile way
- [ ] Balancing Agile flexibility with regulatory compliance
- [ ] Cross-team collaboration in large-scale Agile

24. Do you feel Agile training has improved your ability to manage project unpredictability? *(Scale: 1 = Not at all, 5 = Extremely effective)* *

☆ ☆ ☆ ☆ ☆

25. What additional Agile training or support would help you apply Agile principles better?

Enter your answer

Section 6

Team Collaboration & Agile Culture

26. How well does your team collaborate in an Agile environment? *(Scale: 1 = Poor Collaboration, 5 = Excellent Collaboration)* *

☆ ☆ ☆ ☆ ☆

27. How often does your team actively participate in the following Agile ceremonies? *(Select one per row: Always | Frequently | Sometimes | Rarely | Never)* *

	Always	Frequently	Sometimes	Rarely	Never
Daily Stand-ups	○	○	○	○	○
Sprint Planning	○	○	○	○	○
Backlog Grooming (Refinement)	○	○	○	○	○
Sprint Reviews	○	○	○	○	○
Sprint Retrospectives	○	○	○	○	○
PI Planning	○	○	○	○	○
Demos	○	○	○	○	○

28. How effective are these Agile ceremonies in improving team collaboration and project predictability? *(Scale: 1 = Not Effective, 5 = Highly Effective)* *

☆ ☆ ☆ ☆ ☆

29. Do you willingly participate in Agile ceremonies, or do you feel obligated? *

- ○ I willingly participate because they add value
- ○ I participate because it's required, but I don't see much value
- ○ I participate reluctantly because I have no choice
- ○ I try to avoid them whenever possible

30. What are the biggest challenges in Agile ceremony participation? *(Select all that apply)* *

- [ ] Meetings are too long or unproductive
- [ ] Team members do not actively contribute
- [ ] Lack of clarity on ceremony purpose or outcomes
- [ ] Leadership or stakeholders do not engage in key ceremonies
- [ ] Time zone or remote work challenges
- [ ] High workload makes it difficult to attend

31. What are the biggest challenges in Agile teamwork? *(Select all that apply)* *

- [ ] Lack of clear ownership & accountability
- [ ] Difficulty in cross-functional collaboration
- [ ] Resistance to Agile practices
- [ ] High team turnover affecting Agile adoption

32. Does your team conduct effective retrospectives that lead to real improvements? *

- ◯ Yes, retrospectives lead to actionable improvements.
- ◯ Sometimes, but improvements are not always implemented.
- ◯ No, retrospectives do not create meaningful change.
- ◯ Our team does not conduct retrospectives.

33. What would improve Agile ceremony participation and effectiveness in your team?

Enter your answer

Section 7

Video Participation in Virtual Agile Ceremonies

34. How often does your team switch on videos during virtual Agile ceremonies? *

- ◯ Always
- ◯ Frequently
- ◯ Sometimes
- ◯ Rarely
- ◯ Never

35. How does video participation impact communication and engagement in remote Agile ceremonies? *(Scale: 1 = No Impact, 5 = Significant Positive Impact)* *

☆ ☆ ☆ ☆ ☆

36. If videos are off, what are the main reasons? *(Select all that apply)* *

- ☐ Poor internet connectivity
- ☐ No company requirement to turn them on
- ☐ Team culture doesn't encourage video usage
- ☐ Privacy concerns or discomfort
- ☐ Multitasking during the meeting
- ☐ Lack of perceived value in video participation

37. Would you be more engaged in Agile ceremonies if video participation was encouraged? *(Yes / No / Maybe, depending on the meeting type)* *

- ◯ Yes
- ◯ No
- ◯ Maybe
- ◯ Depending on the meeting Type

Section 8

Challenges in Agile Adoption

38. What are the most common challenges faced after transitioning to Agile? (Select all that apply) *

- [ ] Resistance to change
- [ ] Lack of Agile expertise
- [ ] Difficulty in aligning Agile with regulatory requirements
- [ ] Unclear roles and responsibilities
- [ ] Lack of leadership support
- [ ] Difficulty in managing dependencies
- [ ] Team Active Participation in all Agile meetings
- [ ] Other

39. On a scale of 1 (Strongly Disagree) to 5 (Strongly Agree), rate the following statements: *

	Strongly Disagree	Disagree	Neutral	Agree	Strongly agree
Agile transition has improved collaboration within my team.	○	○	○	○	○
Agile methodologies have reduced project delays.	○	○	○	○	○
Agile adoption has made project planning more complex.	○	○	○	○	○
Agile has enhanced stakeholder communication and alignment.	○	○	○	○	○
Agile practices have increased workload without proportional benefits.	○	○	○	○	○

Section 9

Agile and Project Predictability

40. How has transitioning to Agile affected project predictability in your experience? *

- ◯ Greatly improved
- ◯ Somewhat improved
- ◯ No significant change
- ◯ Somewhat worsened
- ◯ Greatly worsened

41. On a scale of 1 (Strongly Disagree) to 5 (Strongly Agree), rate the following statements: *

	Strongly Disagree	Disagree	Neutral	Agree	Strongly agree
Agile has made it easier to meet project deadlines.	◯	◯	◯	◯	◯
Agile has improved the accuracy of project estimations.	◯	◯	◯	◯	◯
Agile increases project visibility for stakeholders.	◯	◯	◯	◯	◯
Agile has introduced unpredictability in certain project areas.	◯	◯	◯	◯	◯

Section 10

Importance of Predictable Performance

42. How critical is project predictability to the success of aerospace projects? *

- ◯ Extremely critical
- ◯ Very important
- ◯ Moderately important
- ◯ Somewhat important
- ◯ Not important

43. What are the primary consequences of unpredictable project performance in aerospace projects? (Select all that apply) *

- ☐ Cost overruns
- ☐ Schedule slippage
- ☐ Regulatory compliance risks
- ☐ Safety concerns
- ☐ Customer dissatisfaction
- ☐ Project cancellations

Section 11

Constraints and Dependencies in Agile

44. What constraints and dependencies most hinder predictable project outcomes in Agile? (Select all that apply) *

- [ ] Poor backlog prioritization
- [ ] Poor story estimation
- [ ] Poor backlog refinement
- [ ] Resource availability issues
- [ ] Inter-team dependencies
- [ ] Insufficient Agile tooling
- [ ] Regulatory and compliance constraints
- [ ] Other

45. On a scale of 1 (Strongly Disagree) to 5 (Strongly Agree), rate the following statements: *

	Strongly Disagree	Disagree	Neutral	Agree	Strongly agree
Dependencies between teams negatively impact Agile project predictability.	○	○	○	○	○
Agile frameworks need additional processes to improve predictability.	○	○	○	○	○
Project constraints often force deviations from Agile principles.	○	○	○	○	○

46. How does Agile align with regulatory compliance in your industry? *(Select the biggest challenge)* *

- [ ] Agile conflicts with compliance-heavy documentation needs
- [ ] Agile increases transparency but requires extra effort to meet regulations
- [ ] Compliance requirements slow down Agile adoption
- [ ] No major conflicts, Agile and compliance align well
- [ ] Other

Section 12

Metrics for Agile Success & Predictability

47. What key performance indicators (KPIs) does your organization use to measure Agile success? *(Select all that apply)* *

- [ ] Sprint Velocity
- [ ] Lead Time & Cycle Time
- [ ] On-time Delivery Rate
- [ ] Customer Satisfaction (NPS)
- [ ] Code Quality & Defect Rates
- [ ] Employee Engagement & Team Morale
- [ ] Business Value Delivered per Sprint
- [ ] Other

48. How does your team measure Agile success? *(Select all that apply)* *

- [ ] On-time delivery of sprint commitments
- [ ] Quality improvements (defect reduction, fewer reworks)
- [ ] Customer/stakeholder satisfaction
- [ ] Team velocity and throughput
- [ ] Reduction in project delays or overruns

49. How predictable are your project outcomes since Agile adoption? *(Scale: 1 = Highly Unpredictable, 5 = Highly Predictable)* *

☆ ☆ ☆ ☆ ☆

50. What factors contribute most to project unpredictability? *(Rank in order of impact)* *

- Changing requirements
- Resource availability & skill gaps
- Dependencies on other teams
- Scope creep
- Poor estimation techniques

51. How frequently does your team meet sprint commitments? *

- ◯ Always
- ◯ Most of the time
- ◯ Sometimes
- ◯ Rarely
- ◯ Never

52. How frequently does your team measure project predictability? *

- ◯ After every sprint
- ◯ After every release
- ◯ Quarterly
- ◯ Annually
- ◯ We do not measure predictability

Section 13

Agile Decision-Making & Governance

53. Who primarily drives Agile adoption and decision-making in your organization? *

- [ ] Senior Leadership
- [ ] Product Owners & Scrum Masters
- [ ] Engineering/Development Teams
- [ ] Project/Program Management Office (PMO)
- [ ] Cross-functional Agile Teams
- [ ] Release Train Engineer (RTE)
- [ ] Technical Project Manager (TPM)

54. How involved is leadership in ensuring Agile predictability? *(Scale: 1 = Not Involved, 5 = Highly Involved)* *

☆ ☆ ☆ ☆ ☆

Section 14

Psychological & Cultural Impact of Agile Adoption

55. How has Agile adoption impacted team morale and job satisfaction? *(Scale: 1 = Negative Impact, 5 = Highly Positive Impact)* *

☆ ☆ ☆ ☆ ☆

56. What are the most common sources of resistance to Agile adoption? *(Select all that apply)* *

- [ ] Fear of job role changes
- [ ] Lack of understanding of Agile benefits
- [ ] Loss of control by middle management
- [ ] Perceived increase in workload
- [ ] Inconsistent Agile implementation across teams

Section 15

Future of Agile in Your Organization

57. What improvements would you like to see in Agile adoption within your organization?

Enter your answer

58. Do you believe Agile will still be the dominant project management methodology in the next 3-5 years? *

- ◯ Yes
- ◯ No
- ◯ Maybe
- ◯ Unsure

APPENDIX C

INTERVIEW GUIDE

C.1 Introduction

Purpose of the Study

This study aimed to explore the impact of Agile project management on project predictability and efficiency in aerospace projects. Specifically, it sought to identify challenges faced by teams and management after transitioning to Agile, solutions to mitigate these challenges, and how Agile methodologies influenced project predictability. The study also aimed to develop a structured approach for "Predictive Agile Delivery" that balanced flexibility with predictability in high-risk industries.

Interview Objectives

The interviews were conducted to:

1. Identify challenges encountered during Agile adoption.
2. Understand strategies and solutions for overcoming Agile adoption pain points.
3. Assess how Agile methodologies impacted project predictability and performance.
4. Evaluate constraints and dependencies affecting predictable Agile delivery.
5. Gather expert insights on integrating predictive analytics into Agile project management.

C.2 Interview Process and Guidelines

C.2.1 Interview Format

- Type of Interview: Semi-structured with follow-up questions based on responses.
- Duration: 45–60 minutes.
- Mode: In-person and virtual (via Teams/Zoom).
- Recording: Participants were informed that the session could be recorded for accuracy in transcription and analysis.
- Confidentiality: Responses were anonymized, and no personal identifiers were used in research reports.

C.2.2 Participant Selection

Participants were selected based on their involvement in Agile project management within the aerospace industry.

Ideal candidates included:

- Agile practitioners (Scrum Masters, Agile Coaches, Product Owners).
- Project Managers and Program Managers.
- Engineering Leads and Technical Managers.
- Senior Executives responsible for Agile adoption.
- Participants had at least two years of experience working in an Agile environment.

C.2.3 Ethical Considerations

- Verbal Consent: Instead of providing a formal consent form, participants were verbally informed about the study's purpose, confidentiality, and their right to withdraw at any time.
- Voluntary Participation: Participants were reminded that they could withdraw at any time without consequences.
- Confidentiality: Data was stored securely, and responses were anonymized.

C.3 Interview Structure and Questions

C.3.1 Opening Remarks

1. Welcome and Introduction: Participants were welcomed and thanked for their time. They were informed that the interview was part of a research study aimed at understanding how Agile project management impacted project predictability and efficiency in aerospace projects.

2. Purpose of the Interview: It was explained that the interview focused on Agile adoption challenges, solutions to improve Agile practices, and ways to enhance project predictability through structured Agile methodologies.

3. Confidentiality and Verbal Consent: Participants were informed that their responses would remain confidential, and that no personally identifiable information would be included in the research findings. They were asked if they agreed to proceed with the interview and whether they were comfortable with the session being recorded for accuracy.

C.3.2 Interview Questions and Follow-ups

Section 1: Background Information

1. Could you describe your role and experience with Agile project management?
2. How long had you worked with Agile, and which frameworks (Scrum, SAFe, Kanban, etc.) had you used?
    a. *Follow-up: What motivated your team or organization to transition to Agile?*
    b. *Follow-up: How had your role evolved after adopting Agile?*

Section 2: Challenges in Agile Adoption

3. What were the most significant challenges your team faced after transitioning to Agile?
4. How did these challenges impact project delivery, budgets, or timelines?
5. Did you experience resistance from leadership, teams, or stakeholders?
    a. *Follow-up: Could you provide a specific example of a challenge and how it was addressed?*
    b. *Follow-up: What unexpected difficulties emerged after Agile adoption?*

Section 3: Agile Solutions and Best Practices

6. What strategies had you found most effective in addressing Agile adoption challenges?
7. How did you ensure that Agile teams aligned with broader organizational goals?
8. What role did leadership play in ensuring Agile success?

a. *Follow-up: Had Agile training or coaching efforts been helpful?*
b. *Follow-up: What metrics did your organization use to measure Agile success?*

Section 4: Agile and Project Predictability

9. Did you believe Agile had improved or reduced project predictability in your experience? Why?
10. What metrics or KPIs did you use to track project predictability in Agile?
11. Had you encountered instances where Agile resulted in unpredictable outcomes?
    a. *Follow-up: How did you balance Agile flexibility with the need for predictability?*
    b. *Follow-up: Could you share a case where Agile directly improved or harmed predictability?*

Section 5: Constraints and Dependencies in Agile Projects

12. What were the biggest constraints and dependencies that challenged predictable project outcomes?
13. How did you manage cross-team dependencies in Agile projects?
    a. *Follow-up: What tools or techniques helped manage dependencies?*
    b. *Follow-up: How did regulatory or supplier constraints impact Agile predictability?*

Section 6: Predictive Agile Delivery and Future Enhancements

14. What were your thoughts on integrating predictive analytics or structured frameworks into Agile?

15. How did you think Predictive Agile Delivery could improve project outcomes?
    a. *Follow-up: Had you used data-driven techniques for Agile forecasting?*
    b. *Follow-up: What additional support or tools would have helped improve predictability?*

Closing Questions

16. If you could have changed one thing about Agile in your organization, what would it have been?
17. What advice would you give to organizations looking to enhance predictability in Agile projects?
    a. *Follow-up: Would you like to add any final thoughts or insights?*

C.4 Post-Interview Process

C.4.1 Data Collection and Transcription

- Recorded interviews were transcribed for analysis.
- Any personally identifiable information was removed to maintain confidentiality.

C.4.2 Data Analysis

- Thematic analysis was conducted to identify patterns and key themes.
- Responses were categorized based on common Agile adoption challenges, solutions, and predictability factors.

C.4.3 Participant Follow-Up

- Participants could be contacted for clarification or additional insights if required.
- A summary of key findings could be shared with participants upon request.

C.5 Summary of Ethical Considerations

Aspect	Details
Verbal Consent	Participants were informed about the study's purpose, confidentiality, and their voluntary participation.
Voluntary Participation	Participants could withdraw at any time without consequences.
Confidentiality	All responses were anonymized, and data was stored securely.
Data Storage	Interview recordings and transcripts were securely stored and only accessible to the research team.

C.6 Conclusion

This interview protocol ensured a structured and ethical approach to collecting qualitative insights on Agile project management in aerospace. The findings from these interviews contributed to developing a comprehensive Predictive Agile Delivery framework to enhance project predictability and efficiency.

APPENDIX D

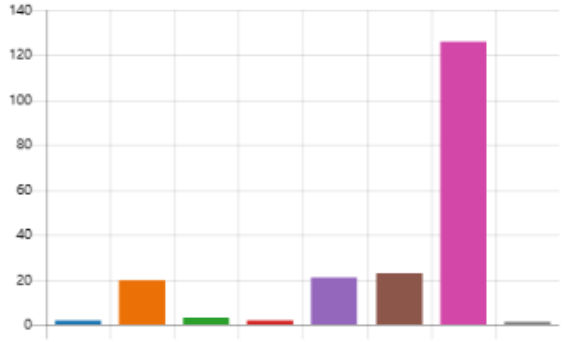
SURVEY DATA

Impact of Agile Approach on Project Predictability and Performance in Aerospace Projects

198 Responses 39:36 Average time to complete Closed Status

1. What is your Role in the Organization?

Senior Leader	2
Manager	20
Agile Coach	3
Release Train Engineer (RTE)	2
Product Owner (PO)	21
Scrum Master (SM)	23
Team Member (Developer/Self...	126
Other	1



2. How many years of experience you have in the organization?

0-3	60
4-8	52
9-12	49
13+	37

3. What is the Agile maturity level of your Organization?

Beginner (New to Agile)	7
Intermediate (Some Agile ado...	97
Advanced (Fully Agile organiz...	94

4. What is the size of your typical project team?

1–5 members	1
6–10 members	23
11–15 members	156
More than 15 members	18

5. How complex are the projects you work on?

Low complexity	0
Medium complexity	24
High complexity	60
Very high complexity	114

6. What level of involvement do you have in Agile decision-making? *(Scale: 1 = No Involvement, 5 = Highly Involved)*

198
Responses

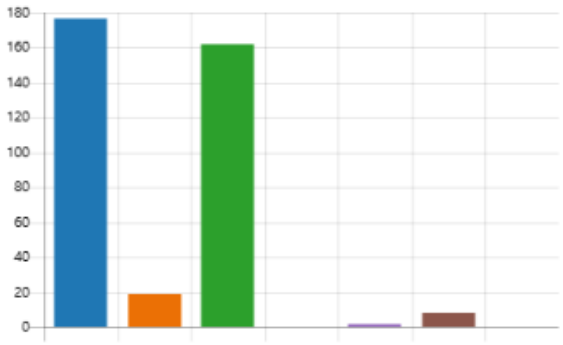
★★★½☆
3.51 Average Rating

7. How long has your team been practicing Agile?

Less than 1 year	3
1–3 years	86
3–5 years	99
More than 5 years	10

8. Which Agile frameworks/methodologies does your organization follow? *(Select all that apply)*

Scrum	177
Kanban	19
SAFe (Scaled Agile Framework)	162
LeSS (Large Scale Scrum)	0
Disciplined Agile (DA)	2
Hybrid (Agile + Waterfall)	8
Other	0

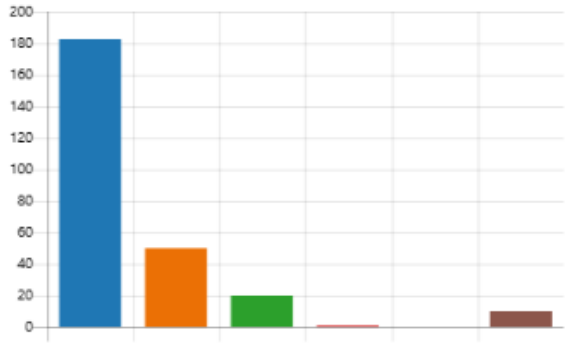


9. If your organization uses a hybrid model, what percentage of project management follows Agile vs. Traditional methods?

Mostly Agile (80% Agile, 20% ...	187
Balanced (50% Agile, 50% Tra...	11
Mostly Traditional (20% Agile, ...	0

10. Have you received formal Agile training or certifications? *(Select all that apply)*

Agile 101	183
Certified Scrum Master (CSM)	50
Certified Product Owner (CPO)	20
SAFe Agilist (SA) / SAFe Progr...	1
PMI-ACP (Agile Certified Pract...	0
None	10



11. How confident are you in your Agile knowledge and application? *(Scale: 1 = Not Confident, 5 = Highly Confident)*

198
Responses

3.59 Average Rating

12. Does your organization provide formal Agile training and mentorship?

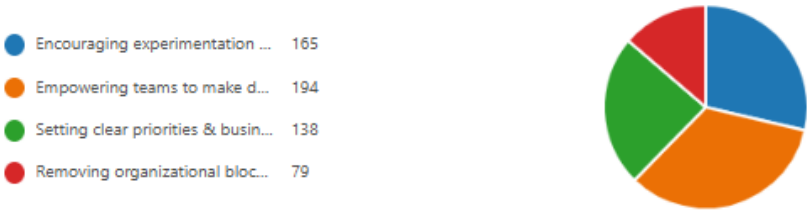
Yes, extensive training is provi...	122
Some training, but mostly self...	74
No formal training, learning is ...	1
No Agile training at all	1

13. Does leadership actively support Agile transformation? *(Scale: 1 = Not Supportive, 5 = Highly Supportive)*

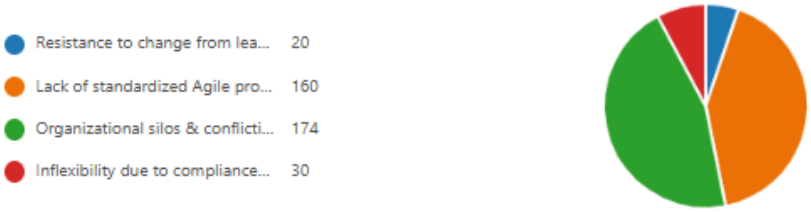
198
Responses

3.91 Average Rating

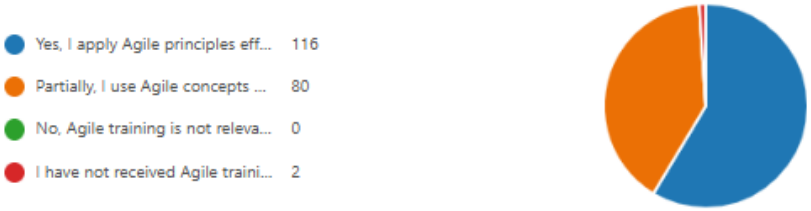
14. What leadership behaviors have the most impact on Agile adoption? *(Select all that apply)*



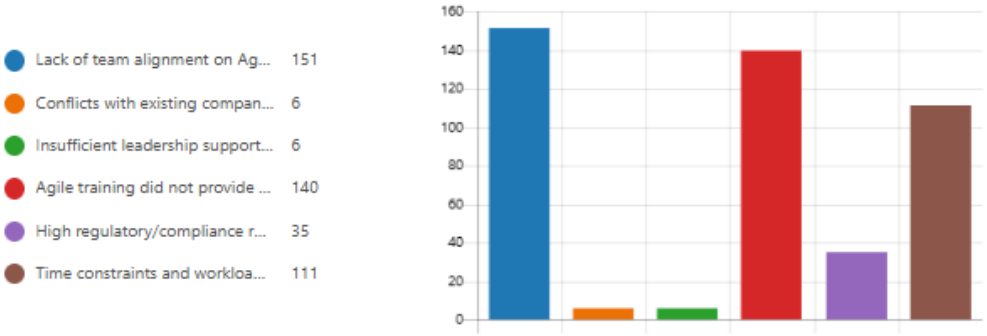
15. What organizational barriers make Agile adoption difficult? *(Select all that apply)*



16. Have you been able to apply Agile training in your daily tasks?

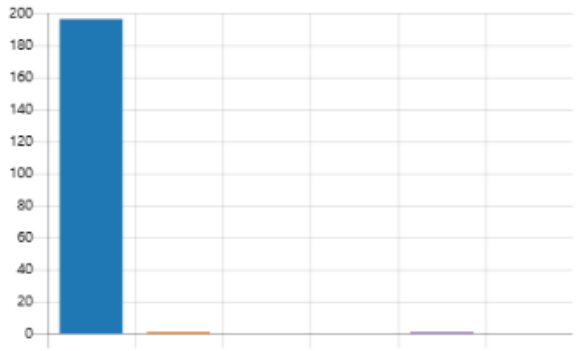


17. What barriers prevent you from applying Agile training in your tasks? *(Select all that apply)*



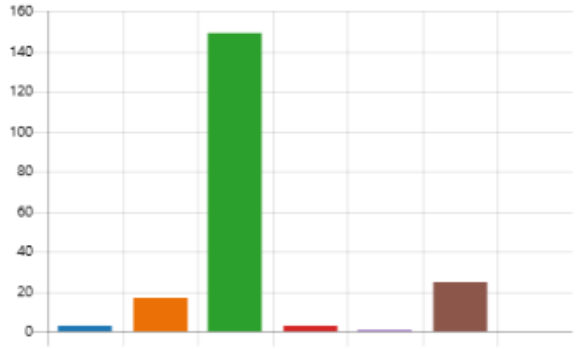
18. How frequently do you use Agile techniques such as stand-ups in your role?

Daily	196
Weekly	1
Bi-weekly	0
Monthly	0
Rarely	1
Never	0



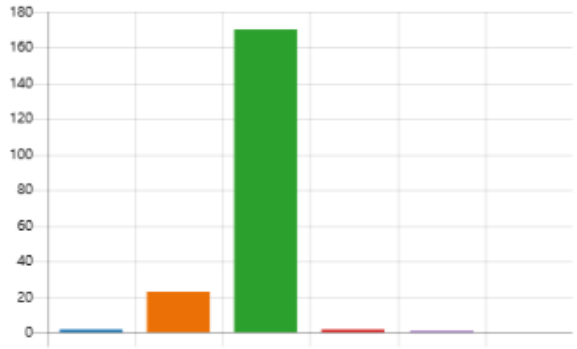
19. How frequently do you use Agile techniques such as backlog grooming in your role?

Daily	3
Weekly	17
Bi-Weekly	149
Monthly	3
Rarely	1
As needed	25
Never	0



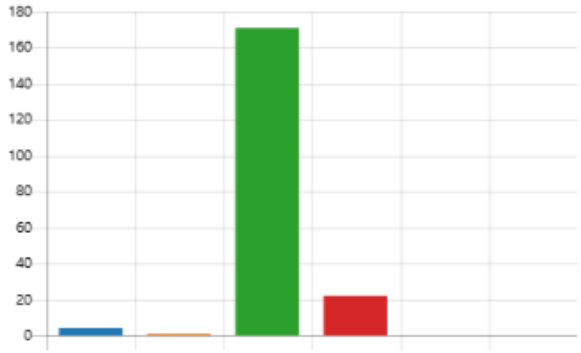
20. How frequently do you use Agile techniques such as sprint planning in your role?

Daily	2
Weekly	23
Bi-Weekly	170
Monthly	2
Rarely	1
Never	0



21. How frequently do you use Agile techniques such as retrospectives in your role?

Daily	4
Weekly	1
Bi-Weekly	171
Monthly	22
Rarely	0
Never	0



22. Which Agile concepts are easiest for you to apply in your tasks? *(Select all that apply)*

Iterative development & conti...	172
Self-organizing teams & colla...	106
Sprint planning & backlog pri...	182
Retrospectives & continuous i...	187
Agile metrics & predictability t...	26

23. Which Agile concepts are hardest for you to apply in your tasks? *(Select all that apply)*

Estimating and forecasting wo...	156
Handling unplanned work in a...	171
Balancing Agile flexibility with ...	22
Cross-team collaboration in la...	140

24. Do you feel Agile training has improved your ability to manage project unpredictability? *(Scale: 1 = Not at all, 5 = Extremely effective)*

198
Responses

★★★☆☆
3.34 Average Rating

25. What additional Agile training or support would help you apply Agile principles better?

114
Responses

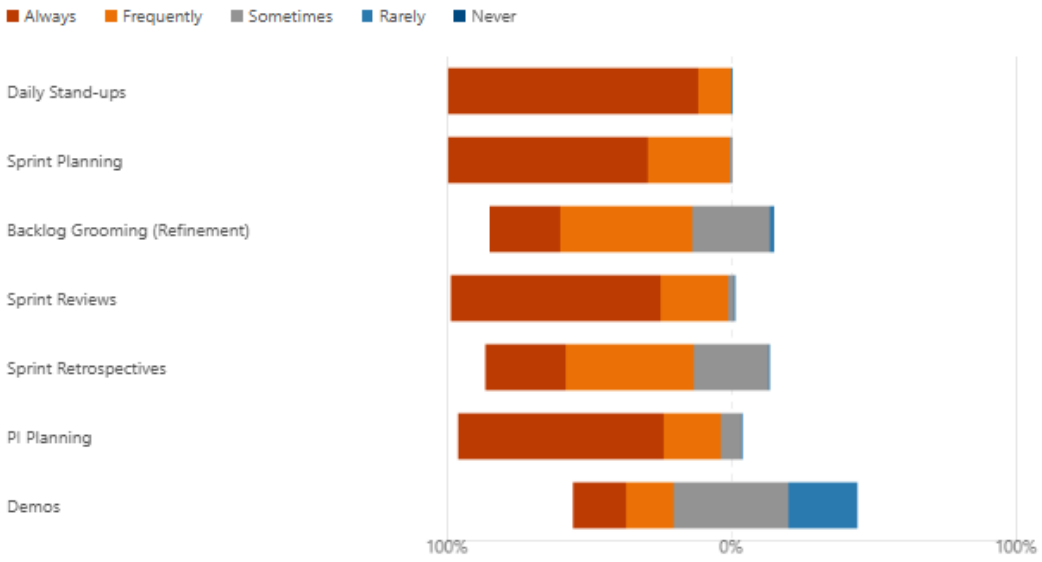
Latest Responses
"Industry specific Agile metrics "

26. How well does your team collaborate in an Agile environment? *(Scale: 1 = Poor Collaboration, 5 = Excellent Collaboration)*

198
Responses

★★★½☆
3.47 Average Rating

27. How often does your team actively participate in the following Agile ceremonies? *(Select one per row: Always | Frequently | Sometimes | Rarely | Never)*



28. How effective are these Agile ceremonies in improving team collaboration and project predictability? *(Scale: 1 = Not Effective, 5 = Highly Effective)*

198
Responses

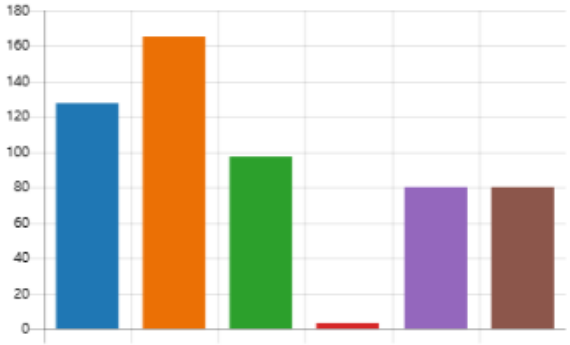
★★★★☆
3.89 Average Rating

29. Do you willingly participate in Agile ceremonies, or do you feel obligated?

I willingly participate because ...	189
I participate because it's requi...	8
I participate reluctantly becau...	1
I try to avoid them whenever ...	0

30. What are the biggest challenges in Agile ceremony participation? *(Select all that apply)*

Meetings are too long or unpr...	128
Team members do not activel...	165
Lack of clarity on ceremony p...	97
Leadership or stakeholders do...	3
Time zone or remote work cha...	80
High workload makes it difficu...	80



31. What are the biggest challenges in Agile teamwork? *(Select all that apply)*

Lack of clear ownership & acc...	161
Difficulty in cross-functional c...	172
Resistance to Agile practices	22
High team turnover affecting ...	24

32. Does your team conduct effective retrospectives that lead to real improvements?

Yes, retrospectives lead to acti...	91
Sometimes, but improvement...	107
No, retrospectives do not crea...	0
Our team does not conduct re...	0

33. What would improve Agile ceremony participation and effectiveness in your team?

113
Responses

Latest Responses

"Clear communications about expectations to engineers out of Agile C...

34. How often does your team switch on videos during virtual Agile ceremonies?

Always	5
Frequently	17
Sometimes	136
Rarely	40
Never	0

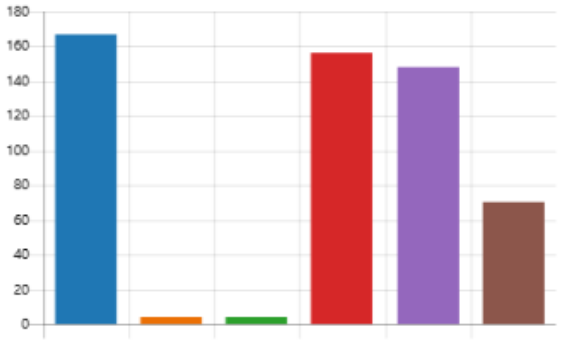
35. How does video participation impact communication and engagement in remote Agile ceremonies? *(Scale: 1 = No Impact, 5 = Significant Positive Impact)*

198
Responses

★★★★☆
3.88 Average Rating

36. If videos are off, what are the main reasons? *(Select all that apply)*

Poor internet connectivity	167
No company requirement to t...	4
Team culture doesn't encoura...	4
Privacy concerns or discomfort	156
Multitasking during the meeti...	148
Lack of perceived value in vide...	70

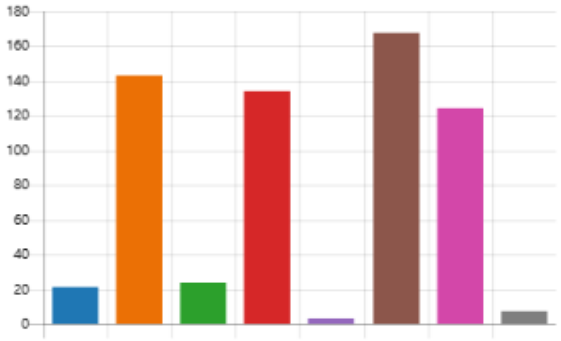


37. Would you be more engaged in Agile ceremonies if video participation was encouraged? *(Yes / No / Maybe, depending on the meeting type)*

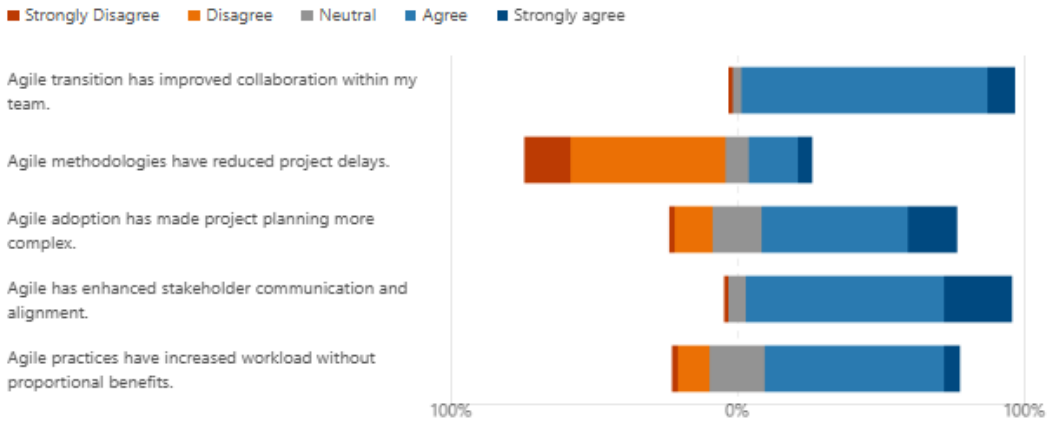
Yes	132
No	8
Maybe	7
Depending on the meeting Ty...	51

38. What are the most common challenges faced after transitioning to Agile? (Select all that apply)

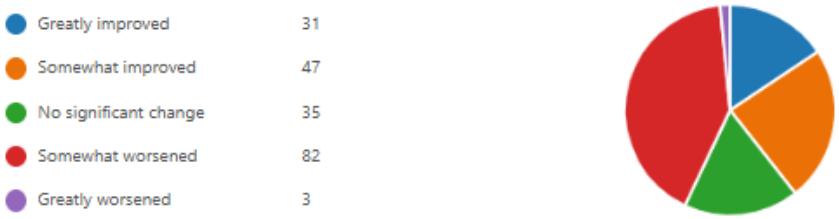
Resistance to change	21
Lack of Agile expertise	143
Difficulty in aligning Agile wit...	24
Unclear roles and responsibilit...	134
Lack of leadership support	3
Difficulty in managing depend...	168
Team Active Participation in all...	124
Other	7



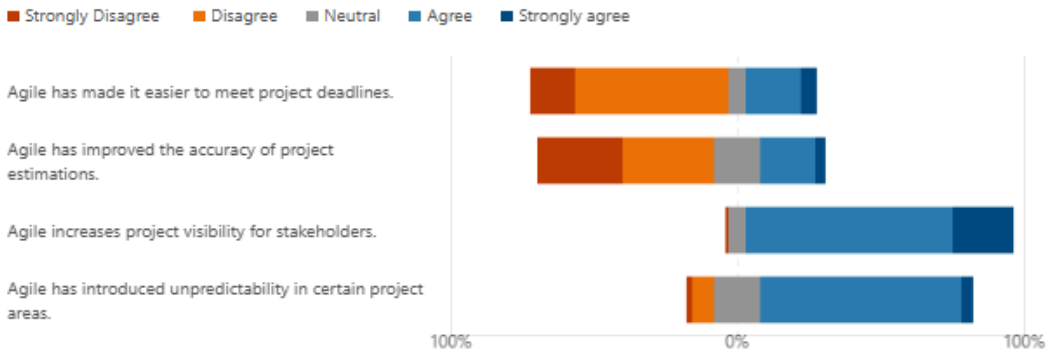
39. On a scale of 1 (Strongly Disagree) to 5 (Strongly Agree), rate the following statements:



40. How has transitioning to Agile affected project predictability in your experience?



41. On a scale of 1 (Strongly Disagree) to 5 (Strongly Agree), rate the following statements:

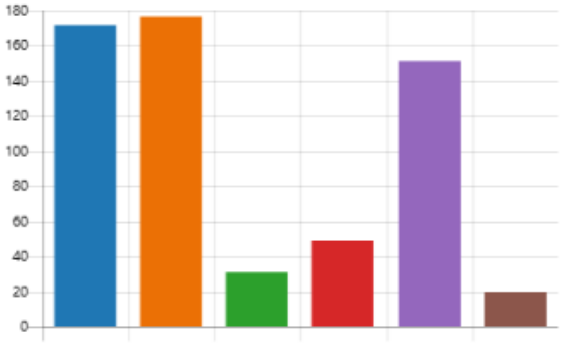


42. How critical is project predictability to the success of aerospace projects?

Extremely critical	116
Very important	74
Moderately important	3
Somewhat important	4
Not important	1

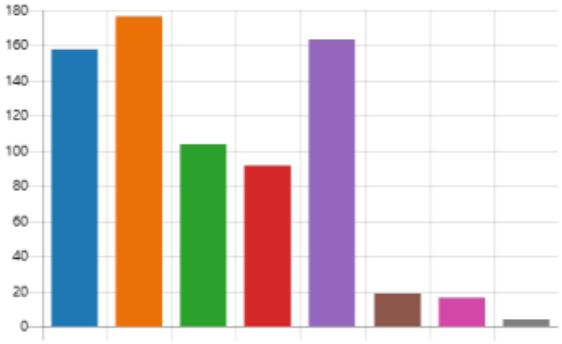
43. What are the primary consequences of unpredictable project performance in aerospace projects? (Select all that apply)

Cost overruns	172
Schedule slippage	177
Regulatory compliance risks	31
Safety concerns	49
Customer dissatisfaction	151
Project cancellations	20

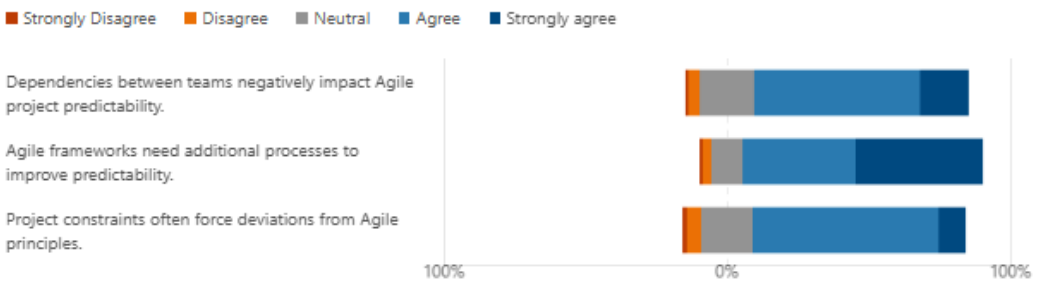


44. What constraints and dependencies most hinder predictable project outcomes in Agile? (Select all that apply)

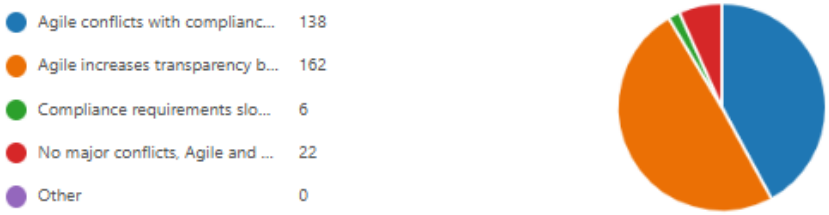
Poor backlog prioritization	158
Poor story estimation	177
Poor backlog refinement	104
Resource availability issues	92
Inter-team dependencies	164
Insufficient Agile tooling	19
Regulatory and compliance co...	16
Other	4



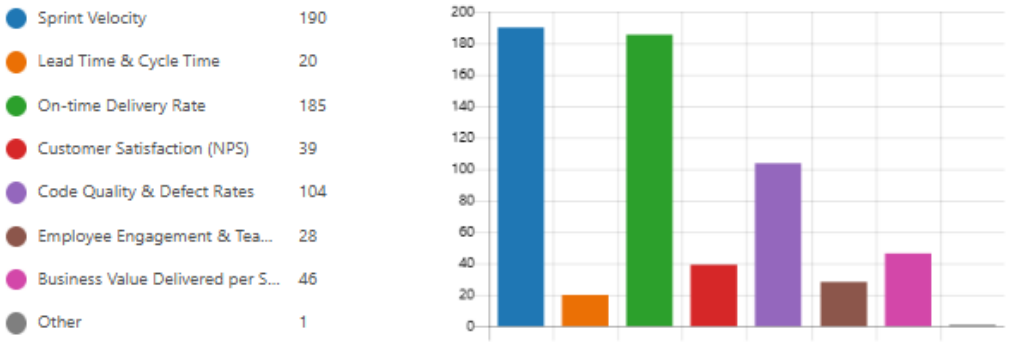
45. On a scale of 1 (Strongly Disagree) to 5 (Strongly Agree), rate the following statements:



46. How does Agile align with regulatory compliance in your industry? *(Select the biggest challenge)*



47. What key performance indicators (KPIs) does your organization use to measure Agile success? *(Select all that apply)*



48. How does your team measure Agile success? *(Select all that apply)*

On-time delivery of sprint co...	196
Quality improvements (defect ...	190
Customer/stakeholder satisfac...	56
Team velocity and throughput	83
Reduction in project delays or ...	54

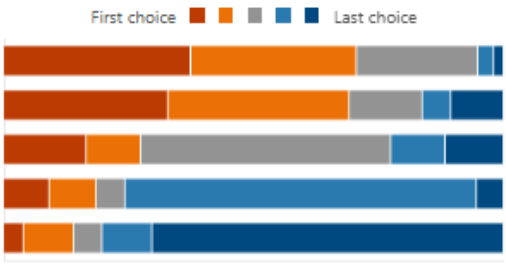
49. How predictable are your project outcomes since Agile adoption? *(Scale: 1 = Highly Unpredictable, 5 = Highly Predictable)*

198
Responses

★★★★½☆
3.61 Average Rating

50. What factors contribute most to project unpredictability? *(Rank in order of impact)*

Rank	Options
1	Changing requirements
2	Poor estimation techniques
3	Dependencies on other teams
4	Resource availability & skill ga...
5	Scope creep



51. How frequently does your team meet sprint commitments?

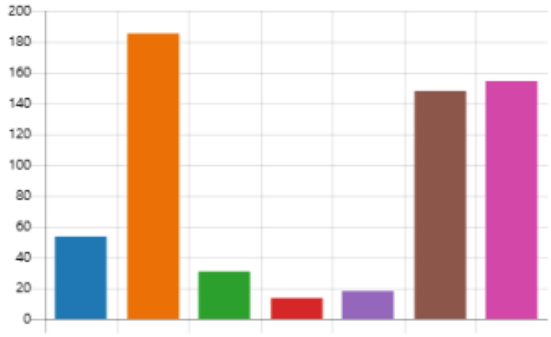
Always	9
Most of the time	179
Sometimes	9
Rarely	1
Never	0

52. How frequently does your team measure project predictability?

After every sprint	42
After every release	115
Quarterly	9
Annually	0
We do not measure predictabi...	32

53. Who primarily drives Agile adoption and decision-making in your organization?

Senior Leadership	54
Product Owners & Scrum Mas...	185
Engineering/Development Tea...	31
Project/Program Management...	14
Cross-functional Agile Teams	18
Release Train Engineer (RTE)	148
Technical Project Manager (TP...	155



54. How involved is leadership in ensuring Agile predictability? *(Scale: 1 = Not Involved, 5 = Highly Involved)*

198
Responses

★★★★☆
3.89 Average Rating

55. How has Agile adoption impacted team morale and job satisfaction? *(Scale: 1 = Negative Impact, 5 = Highly Positive Impact)*

198
Responses

★★★½☆
3.36 Average Rating

56. What are the most common sources of resistance to Agile adoption? *(Select all that apply)*

Fear of job role changes	29
Lack of understanding of Agil...	170
Loss of control by middle man...	19
Perceived increase in workload	128
Inconsistent Agile implementa...	161

57. What improvements would you like to see in Agile adoption within your organization?

107
Responses

Latest Responses

"More experimentation with different sprint durations and willingness ...

58. Do you believe Agile will still be the dominant project management methodology in the next 3-5 years?

Yes	139
No	5
Maybe	44
Unsure	10

REFERENCES

1) Agilemania. (2023). Approach to estimations in SAFe framework. Retrieved from https://agilemania.com/approach-to-estimations-in-safe-framework
2) Ambler, S. W., & Lines, M. (2012). Disciplined Agile delivery: A practitioner's guide to Agile software delivery in the enterprise. IBM Press.
3) Anderson, D. J. (2010). Kanban: Successful evolutionary change for your technology business. Blue Hole Press.
4) Atlassian. (2024). Backlog refinement meeting. Retrieved from https://www.atlassian.com/agile/project-management/backlog-refinement-meeting
5) Bass, J. M. (2016). How product owner teams scale agile methods to large distributed enterprises. Empirical Software Engineering, 20(6):1525-1557.
6) Beck, K. (1999). Extreme programming explained: Embrace change. Addison-Wesley.
7) Beck, K., et al. (2001). Manifesto for Agile Software Development. Agile Alliance. Retrieved from https://agilemanifesto.org
8) Beck, K., & Andres, C. (2004). Extreme programming explained: Embrace change (2nd ed.). Addison-Wesley.
9) Bick, S., Spohrer, K., Hoda, R., Scheerer, A., & Heinzl, A. (2018). Coordination Challenges in Large-Scale Software Development: A Case Study of Planning Misalignment in Hybrid Settings, IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, TSE-2016-04-0139.
10) Boehm, B., & Turner, R. (2004). Balancing agility and discipline: A guide for the perplexed. Addison-Wesley.
11) Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. Qualitative Research in Psychology, 3(2), 77–101.

12) Brainbok. (2024). Agile risk management. Retrieved from https://www.brainbok.com/guide/agile/agile-risk-management

13) Cockburn, A. (2007). Agile Software Development: The Cooperative Game (2nd ed.). Addison-Wesley.

14) Cohn, M. (2005). Agile estimating and planning. Pearson Education.

15) Creswell, J. W., & Plano Clark, V. L. (2018). Designing and Conducting Mixed Methods Research (3rd ed.). SAGE Publications.

16) Creswell, J. W., & Creswell, J. D. (2023). Research design: Qualitative, quantitative, and mixed methods approaches (6th ed.). Sage Publications.

17) Denning, S. (2018). The Age of Agile: How Smart Companies Are Transforming the Way Work Gets Done. AMACOM.

18) Derby, E., & Larsen, D. (2006). Agile retrospectives: Making good teams great. Pragmatic Bookshelf.

19) Digital.ai. (2023). 17th Annual State of Agile Report. Retrieved from https://digital.ai/resource-center/analyst-reports/state-of-agile-report

20) Dikert, K., Paasivaara, M., & Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations: A systematic literature review. Journal of Systems and Software, DOI:10.1007/978-3-031-56599-1_52.

21) Dillman, D. A., Smyth, J. D., & Christian, L. M. (2014). Internet, phone, mail, and mixed-mode surveys: The tailored design method (4th ed.). Wiley.

22) Dingsoyr, T., Nerur, S., Balijepally, V., & Moe, N. B. (2012). A decade of agile methodologies: Towards explaining agile software development. Journal of Systems and Software, DOI: 10.1016/j.jss.2012.02.033.

23) Easy Agile. (2024). The problem with Agile estimation. Retrieved from https://www.easyagile.com/blog/problem-with-agile-estimation

24) Easy Agile. (2024). 5 Agile Estimation Tips To Help With Backlog Prioritization. Retrieved from https://www.easyagile.com/blog/agile-estimation

25) Fowler, M. (2001). Refactoring: Improving the design of existing code. Addison-Wesley.

26) Fitzgerald, B., & Stol, K.-J. (2017). Continuous software engineering: A roadmap and agenda. Journal of Systems and Software, 123, 176-189.

27) Forsgren, N., Humble, J., & Kim, G. (2018). Accelerate: The Science of Lean Software and DevOps: Building and Scaling High Performing Technology Organizations. IT Revolution Press.

28) Growing Scrum Masters. (2024). What are the challenges of prioritizing a product backlog in a rapidly changing market? Retrieved from https://www.growingscrummasters.com/deploy-improve-scrum/what-are-the-challenges-of-prioritizing-a-product-backlog-in-a-rapidly-changing-market

29) Highsmith, J. (2009). Agile project management: Creating innovative products. Pearson Education.

30) Kerzner, H. (2022). Project Management: A Systems Approach to Planning, Scheduling, and Controlling (13th ed.). Wiley.

31) Kim, G., Humble, J., Debois, P., & Willis, J. (2016). The DevOps handbook: How to create world-class agility, reliability, & security in technology organizations. IT Revolution.

32) Knaster, R., & Leffingwell, D. (2020). SAFe distilled: Applying the Scaled Agile Framework for Lean enterprises (5th ed.). Addison-Wesley.

33) Kniberg, H., & Skarin, M. (2010). Kanban and Scrum: Making the most of both. C4Media.

34) KnowledgeHut. (2025, February 14). SAFe Agile Ceremonies: Your Ultimate Guide. Retrieved from https://www.knowledgehut.com/blog/agile/safe-agile-ceremonies

35) Larman, C., & Vodde, B. (2010). Practices for scaling lean & agile development: Large, multisite, and offshore product development with large-scale Scrum. Addison-Wesley.

36) Larman, C., & Vodde, B. (2010). Large-Scale Scrum: More with LeSS. Pearson Education.

37) Leffingwell, D. (2019). SAFe 4.5 reference guide: Scaled Agile Framework for Lean Enterprises. Addison-Wesley Professional.

38) Leffingwell, D. (2019). SAFe 5.0: The world's leading framework for business agility. Scaled Agile, Inc.

39) Leopold, K., & Kaltenecker, S. (2015). Kanban change leadership: Creating a culture of continuous improvement. Wiley.

40) Mahnic, V., & Hovelja, T. (2012). On using planning poker for estimating user stories. Journal of Systems and Software, 85(9), 2086-2095.

41) Martin, R. C. (2003). Agile software development: Principles, patterns, and practices. Prentice Hall.

42) Montgomery, D. C., Peck, E. A., & Vining, G. G. (2012). Introduction to Linear Regression Analysis (5th ed.). Wiley.

43) Morse, J. M. (2015a). Data saturation in qualitative research. Qualitative Health Research, 25(5), 587-588.

44) Net Solutions. (2022). Top 8 Agile Estimation Techniques (Explained with Examples). Retrieved from https://www.netsolutions.com/insights/how-to-estimate-projects-in-agile/

45) Parabol. (2024). Agile estimation techniques. Retrieved from https://www.parabol.co/blog/agile-estimation-techniques

46) Petersen, K., & Wohlin, C. (2010). The effect of moving from a plan-driven to an incremental and agile development approach: An industrial case study. Empirical Software Engineering, 15(6), 654-693.

47) Ramesh, B., Cao, L., & Baskerville, R. (2010). Agile requirements engineering practices and challenges: an empirical study. Information Systems Journal, 20(5), 449-480.

48) Rapidr. (2023). How to Manage Product Backlog (Strategies, Tips & Tricks). Retrieved from https://rapidr.io/blog/product-backlog-management/

49) Rising, L., & Janoff, N. S. (2000). The Scrum software development process for small teams. IEEE Software, 17(4), 26-32.

50) Saunders, M., Lewis, P., & Thornhill, A. (2019). Research methods for business students (8th ed.). Pearson Education.

51) Scaled Agile, Inc. (2023). SAFe framework: Agile Release Train (ART). Retrieved from https://www.scaledagileframework.com

52) Scaled Agile, Inc. (2023). SAFe 6.0 for Lean Enterprises. Scaled Agile, Inc.

53) Scaled Agile, Inc. (2023). SAFe framework overview. Scaled Agile, Inc. https://www.scaledagileframework.com

54) Scaled Agile Framework. (n.d.). Estimation and WSJF in SAFe. Retrieved from https://framework.scaledagile.com/wsjf

55) Scaled Agile Framework. (n.d.). Essential SAFe. Retrieved from https://scaledagileframework.com/essential-safe

56) Scaled Agile Framework. (n.d.). SAFe Scrum. Retrieved from https://scaledagileframework.com/safe-scrum

57) Scaled Agile Framework. (n.d.). Backlog refinement. Retrieved from https://scaledagileframework.com/backlog-refinement

58) Scaled Agile Framework. (n.d.). Scrum of Scrums and PO Sync. Retrieved from https://scaledagileframework.com/scrum-of-scrums

59) Scaled Agile Framework. (n.d.). Lean Portfolio Management (LPM). Retrieved from https://scaledagileframework.com/lean-portfolio-management

60) Scaled Agile Framework. (n.d.). Solution Train Engineering. Retrieved from https://scaledagileframework.com/solution-train

61) Schwaber, K., & Sutherland, J. (2017). The Scrum guide: The definitive guide to Scrum: The rules of the game. Scrum.org.

62) Scrum Institute. (n.d.). Scrum Effort Estimation: A Guide of Best Estimation Techniques for Agile Scrum Teams. Retrieved from https://www.scrum-institute.org/scrum-effort-estimations-planning-poker-the-scrum-framework.php

63) Scrum Institute. (n.d.). Scrum Grooming Meeting: Backlog Refinement Practices. Retrieved from https://www.scrum-institute.org/scrum-grooming-meeting-the-scrum-framework.php

64) Scrum.org. (2024). Overcoming common challenges in initial sprint planning. Retrieved from https://www.scrum.org/forum/scrum-forum/87194/overcoming-common-challenges-initial-sprint-planning

65) Scrum Alliance. (n.d.). The Top 3 Challenges Companies Face During the Transition to Scrum. Retrieved from https://resources.scrumalliance.org/Article/3-challenges-companies-face-transition-scrum

66) Senge, P. M. (1990). The Fifth Discipline: The Art and Practice of the Learning Organization. Doubleday.

67) Serrador, P., & Pinto, J. K. (2015). Does Agile work? A quantitative analysis of Agile project success. International Journal of Project Management, 33(5), 1040-1051.

68) Sivell, S., Prout, H., Hopewell-Kelly, N., Baillie, J., Byrne, A., Edwards, M., ... & Nelson, A. (2015). Sensitive, Challenging, and Difficult Topics: Experiences and Practical Considerations for Qualitative Researchers. BMC Palliative Care, DOI: 10.1177/16094069221124739.

69) StarAgile Consulting. (2025, February 12). Overview Of SAFe Ceremonies. Retrieved from https://staragile.com/blog/safe-ceremonies

70) Taber, K. S. (2018). The use of Cronbach's Alpha when developing and reporting research instruments in science education. Research in Science Education, 48(6), 1273–1296.

71) Tong, A., & Dew, K. (2016). Qualitative Research in Transplantation: Ensuring Relevance and Rigor, DOI: 10.1097/TP.0000000000001117.

72) VersionOne. (2021). State of Agile Report. Digital.ai.

73) Womack, J. P., & Jones, D. T. (2003). Lean Thinking: Banish Waste and Create Wealth in Your Corporation. Free Press.

74) Yamane, T. (1967). Statistics: An Introductory Analysis (2nd ed.). Harper and Row.