



LAPIN YLIOPISTO
UNIVERSITY OF LAPLAND

Shafaq Malik

**Making Agile Work in Micro Startups: Challenges, What Fits, and How to
Adapt**

Master's Thesis
Faculty of Art and Design
Spring 2025

University of Lapland, Faculty of Art and Design

Title: Making Agile Work in Micro Startups: Challenges, What Fits, and How to Adapt

Author: Shafaq Malik

Degree program: Service Design Strategies and Innovations

The type of the work: Master's thesis

Number of pages, appendices: 74

Year: 2025

ABSTRACT

Agile project management has become prevalent in software development, providing flexibility, incremental delivery, and improved collaboration. Nonetheless, its utilization in micro software startups (entities with fewer than ten employees) remains inadequately examined. These startups function under significant limitations, including restricted resources, overlapping responsibilities, and fluctuating market conditions, which complicates the adaption of traditional Agile methodologies such as Scrum or Kanban. This study examines the effective adaptation of Agile methodology to meet the distinct requirements of micro software startups, highlighting context-specific difficulties and facilitators.

The research employs a pragmatic mindset and a hybrid deductive-inductive methodology, integrating thematic coding of five seminal academic publications with qualitative analysis of seven semi-structured interviews conducted with professionals in micro startups. The research examines three fundamental questions: What are the primary hurdles that micro startups encounter while implementing Agile methodologies? To what extent are current Agile frameworks appropriate in these contexts? What effective tactics can improve Agile adoption in micro startups?

The results underscore five interrelated themes: Team Empowerment, Agile Framework Fit & Frictions, Cultural & Change Barriers, Contextual Tailoring Needs and Collaboration–Quality Linkages. The strict implementation of Agile concepts frequently caused friction, but context-driven adaptation supported by empowered teams, streamlined procedures, and adaptive leadership was crucial for enduring Agile practice. Resistance to change and ambiguity in responsibilities further hindered adoption. Nonetheless, effective communication among the creative team and flexible mindset were essential determinants in the adaptation of Agile concepts.

This study presents a conceptual framework for "micro-adaptive Agile," framing agility as a dynamic, practice-oriented discipline rather than a rigid approach. It advocates for the consistent calibration of Agile methodologies and customized adjustments based on team dynamics and operational circumstances. The research provides practical insights for startup founders, project managers, and Agile coaches operating in micro-scale environments. It also encourages governments and investors to reevaluate the metrics and support mechanisms for agility in nascent software startups.

This study addresses a deficiency in Agile literature and offers an evidence-based paradigm for adaptation, so enhancing the resilience, efficiency, and contextual awareness of Agile implementation in micro software startups.

Keywords: agile, micro startups, software development, project management, agile adaption

Table of Contents

ABSTRACT.....	1
Table of Contents	3
List of Figures	5
List of Tables.....	5
Abbreviations and Glossary.....	6
1. INTRODUCTION.....	7
1.1. Research Background.....	7
1.2. Research Motivation	9
1.3. Research Goals and Research Questions	10
1.4. Research Scope.....	11
1.5. Thesis Structure	11
1.6. Ethical Consideration.....	11
2. LITERATURE REVIEW	13
2.1 Startups.....	14
2.2 Challenges in Startups	18
2.2.1 Financial Constraints.....	18
2.2.2 Work-Life Balance	18
2.2.3 Market Competition.....	19
2.3 Agile Management.....	19
2.4 Impact of Agile Management on Software Startups.....	25
2.5 Challenges of agile management in micro software startups.....	26
2.5.1 People	27
2.5.2 Agile misconceptions.....	28
2.5.3 Organization Misalignment.....	28
2.6 Literature Summary	28
3. RESEARCH DESIGN.....	31
3.1. Research philosophy	31
3.2. Research Approach.....	32
3.3. Research Strategy.....	32
3.4. Temporal Perspective	32
3.5. Sampling Strategy.....	33
3.6. Data Collection	33
3.6.1 Sample of Literature	33
3.6.2 Participants Overview	35
4. RESULTS AND FINDINGS.....	38

4.1 Overview of Analytical Approach.....	38
4.2 Data Analysis Results	38
4.2.1 Literature Coding Analysis.....	38
4.2.1.1 Cross-Paper Insights.....	41
4.2.1.2 Reflexive Thoughts.....	42
4.2.2 Interview Coding Analysis.....	42
4.2.2.1 Cross-Interview Insights	44
4.2.2.2 Reflexive Thoughts.....	45
4.3 Summary of Findings	45
4.3.1 Thematic Analysis	45
4.3.1.1 Theme 1 – Team Empowerment	47
4.3.1.2 Theme 2 – Agile Framework Fit & Frictions	48
4.3.1.3 Theme 3 – Cultural & Change Barriers	48
4.3.1.4 Theme 4 – Contextual Tailoring Needs	48
4.3.1.5 Theme 5 – Collaboration–Quality Linkages	48
4.3.2 Theme Interrelationships	49
4.3.3 Holistic Literature Triangulation	49
4.3.4 Cross-Theme Insights.....	50
4.4 Conceptual Framework.....	51
4.5 Reflexivity Statement.....	53
4.6 Results Summary	53
5. DISCUSSION AND IMPLICATION	55
5.1. Discussion	55
5.2. Practical implication	55
5.3. Future studies	56
6. CONCLUSION.....	57
LIST OF REFERENCES	58
APPENDICES.....	64
Appendix A1. Codebook v1	64
Appendix A2. Coverage Matrix	68
Appendix B1. Interview Questions.....	70
Appendix B2. Participants Demographics	72
Appendix C1. Stable Codebook (v2).....	73

List of Figures

Figure 1. Comparison of company sizes by # of employees	16
Figure 2. Agile Manifesto. Adapted from productplan.com	21
Figure 3. Types of Agile Methods.....	22
Figure 4. Agile Software Development. Adapted from Modelling the Impact of Culture on Agile Methods by S. Ali et al., 2024, P 27	24
Figure 5. Research Onion. Adapted from Saunders et al., (2023)	31
Figure 6. Stepwise pragmatic hybrid methodology	36
Figure 7. Code Heatmap for Interviews.....	44
Figure 8. Conceptual Framework for Micro-Adaptive Agile	53

List of Tables

Table 1 Comparison between Waterfall and Agile	21
Table 2 Literature for Hybrid Coding	34
Table 3 Participants Demographics	35
Table 4 Initial Code set	39
Table 5 Code Frequencies in interviews	43
Table 6 Meta Themes with Codes and Brief Synthesis	46
Table 7 Thematic Coverage Summary.....	49
Table 8 Literature Triangulation.....	50

Abbreviations and Glossary

Agile A project management methodology focused on iterative development, collaboration, and adaptability to change.

CEO Chief Executive Officer, often the founder or main leader in startups.

CPM Critical Path Method, a traditional project scheduling technique used to determine the longest sequence of dependent tasks.

DSDM Dynamic Systems Development Method, an Agile framework focused on iterative and incremental delivery.

FDD Feature-Driven Development, an Agile method centered on delivering tangible, working software features.

GDP General Data Protection Regulation, a European Union regulation on data protection and privacy.

Kanban A visual Agile method for managing workflow and improving process efficiency.

PMI Project Management Institute, a global professional organization for project managers.

RQ Research Question.

SAFe Scaled Agile Framework, an enterprise-level Agile framework (mentioned indirectly through context).

SaaS Software as a Service, a software delivery model where software is accessed online via a subscription rather than bought and installed on individual computers.

Scrum A widely used Agile framework involving roles, events, and artifacts to manage iterative software development.

SDLC Software Development Life Cycle, the process for planning, creating, testing, and deploying software.

SMEs Small and Medium-sized Enterprises, businesses with personnel and revenue below certain limits.

Sprint A set period (usually 2–4 weeks) in Scrum where specific work has to be completed and made ready for review.

Startup A temporary organization designed to search for a repeatable and scalable business model under conditions of high uncertainty.

UX User Experience, though not explicitly defined in the thesis, often implied in user-centric Agile practices.

1. INTRODUCTION

Agile management is a prevalent project management methodology distinguished by its iterative, flexible, and adaptive approach to project development. It is very common in dynamic sectors, particularly in software development. Multiple studies (Eman, 2024; Fergis, 2012; Rahman et al., 2024; Serrador & Pinto, 2015) have demonstrated the significant impact of Agile methodologies in this domain, highlighting their effectiveness in reducing time-to-market and enhancing the integration of customer feedback during the development process. Notwithstanding the established advantages of Agile, a significant deficiency persists in the study about its implementation in severely resource-constrained settings, where fiscal restrictions and limited personnel typically teams of less than ten members present unique problems. Such environments are characteristic of nascent micro software startups, which function under considerable limitations while needing to maintain agility and innovation to thrive and expand. This study aims to rigorously analyse the distinct problems encountered by micro software startups in implementing Agile techniques during their software development processes. It also seeks to find techniques and changes that can enhance the effectiveness of Agile deployment in these circumstances.

This chapter presents a summary of the study, commencing with the research background, followed by the research motivation, objectives, and research questions. It subsequently delineates the importance of the study and closes with an examination of its limitations.

1.1. Research Background

A globally accepted definition of a startup does not exist. Nevertheless, prevalent traits are identified in the literature, indicating that startups are enterprises striving to create scalable business models, frequently within high-risk contexts (Azhar, 2016). Startups generally engage in innovative marketplaces, presenting novel and frequently disruptive products or concepts, while traditional enterprises usually operate within established markets. Audretsch et al. (2020) further define startups as entities motivated not alone by profit but also by their function as incubators of innovation, irrespective of the final financial outcome.

Startups, by prioritizing innovation, inevitably encounter heightened uncertainty and risk, yet also possess the potential for significant gains. Startups may pursue alternative funding sources like venture capital, angel investors, or crowdfunding platforms, while traditional firms generally depend on loans, income streams, or personal savings. Startups emphasize scalability

and swift growth in their business concepts. Startups are characterized by flat hierarchies, flexible position assignments, and dynamic work environments, in stark contrast to the organized and inflexible structures of traditional corporations.

Software startups possess these common characteristics, emphasizing technology-driven products and services. Numerous modern companies base their business strategies on software, either providing it as an independent product or incorporating it into comprehensive service offerings (Azhar, 2016). Software startups generally focus on developing technical products that must traverse the whole Software Development Life Cycle (SDLC), including the phases of planning, design, implementation, testing, deployment, and maintenance. Overseeing this cycle is a significant task owing to the technological intricacies, swift rate of change, and the interconnected variables intrinsic to software development.

Due to the dynamic settings and complex technical requirements encountered by software startups, efficient administration of products, services, and internal processes necessitates flexibility and adaptability. This necessity underscores the essential function of project management approaches. A multitude of strategies exist to address this requirement, including:

1. Waterfall Methodology
2. Critical Path Method (CPM)
3. Agile Management

The **Waterfall Methodology** is a linear software development approach progressing sequentially through phases: requirements, design, implementation, testing, deployment, and maintenance (Khan, 2023). Each phase must be completed fully before starting the next, making it ideal for clearly defined projects. However, its rigid structure complicates handling changes during software development.

The **Critical Path Method (CPM)** is a project scheduling technique used to identify the longest chain of dependent jobs and determine the minimal project time. By focusing on critical tasks, CPM allows project managers to allocate resources effectively and minimize delays (Institute of Project Management, 2022).

Agile management is an iterative and adaptable project management methodology that prioritizes collaboration, client feedback, and swift delivery. It divides projects into smaller, manageable segments, enabling teams to swiftly react to changes and consistently enhance the development process.

1.2. Research Motivation

Iterative project management strategies, particularly Agile, are essential for software startups aiming to deliver products effectively while facilitating rapid adaptability to change and expediting time-to-market. Numerous studies have investigated the efficacy of Agile methodologies and the reasons for their implementation in software startups. These studies continuously emphasize that Agile methodologies are preferred for their capacity to enhance resource management and cultivate a customer-centric approach in software development (Javdani Gandomani et al., 2025; Mkpojiogu et al., 2019; Serrador & Pinto, 2015). By implementing Agile methodologies, software companies can maintain efficiency, foster innovation, and enhance adaptability, thereby optimizing productivity despite constrained resources.

Nonetheless, current study has predominantly concentrated on larger firms, typically consisting of teams with a minimum of 50 members, featuring distinctly delineated roles such as CEO, product manager, scrum master, project managers, engineers, and quality assurance specialists. This emphasis imposes constraints when extrapolating these findings to micro software startups, which generally consist of fewer than 10 members and have minor role differentiation. In these smaller contexts, the CEO may concurrently fulfill the role of product manager, development teams may comprise only two or three members, administrative support is sometimes scant, and the position of scrum master is frequently absent.

The current corpus of research provides insufficient insight into the specific conditions of micro software startups, where tiny team sizes, overlapping roles, and the necessity for rapid and adaptable software development pose unique obstacles. This disparity hinders the efficient execution of Agile techniques in many contexts. As a result, smaller software startup teams frequently have challenges in effectively implementing Agile techniques, so hindering their capacity to fully capitalize on the advantages of Agile management.

1.3. Research Goals and Research Questions

Given the limited research on Agile adoption in micro software startups, this study aims to develop a conceptual framework to support the effective implementation of Agile methodologies in micro software startups, addressing their unique challenges and operational contexts.

To achieve this aim, the research will focus on the following objectives:

Objective 1: To identify and categorize the unique challenges micro software startups face when adopting Agile methodologies.

Objective 2: To assess the suitability and limitations of existing Agile frameworks (e.g., Scrum, Kanban) within micro software startup environments.

Objective 3: To develop a conceptual framework that can help teams better adapt Agile practices in micro startup environments.

Based on these objectives, this study seeks to answer the following research questions:

Research Question 1 - What are the main challenges micro software startups encounter when implementing Agile methodologies?

Research Question 2 - How suitable are existing Agile frameworks (e.g., Scrum, Kanban) for micro software startups?

Research Question 3 - What key strategies and modifications can be implemented to create an effective Agile adoption framework for micro software startups?

This study seeks to enhance the current understanding of Agile adoption by analysing and assessing solutions specifically designed for the distinct circumstances of micro-scale software startups. This study aims to fill the existing research vacuum by offering useful insights into the issues faced by startups and the practical strategies necessary for effective Agile adoption in resource-limited settings. This research aims to provide tangible benefits to micro software startups by supplying them with practical ways for more efficient Agile adoption.

1.4. Research Scope

This study possesses specific limitations that require acknowledgment. The research scope is confined to micro software startups with less than ten people, specifically those involved in the development of software products known as Software as a Service (SaaS).

Nonetheless, it fails to distinguish between sectors within the software startups domain, such as fintech, health tech, or educational technology, thus constraining the sector-specific relevance of its conclusions. The study employs a qualitative research methodology, which, although offering profound insights, may limit the generalizability of the findings to a wider population. Furthermore, as the research does not focus on a particular geographical area, the relevance of its findings across other markets and cultural contexts may be constrained.

1.5. Thesis Structure

This thesis is organized into five chapters, each building toward the development of a conceptual framework for Agile adoption in micro software startups. The first chapter presents the research by delineating the backdrop, problem statement, aim, objectives, and questions, as well as the study's importance, scope, and constraints. The second chapter examines literature on Agile methodology and the obstacles encountered by micro software startups, emphasizing critical issues in Agile implementation amid resource limitations.

The third chapter delineates the qualitative methodology, encompassing interview and literature-based data collecting, analytical tools, and ethical considerations. The fourth chapter elucidates and analyses significant findings in relation to extant research, highlighting patterns and obstacles in Agile adoption. The last chapter encapsulates insights, addresses the research inquiries, provides pragmatic recommendations, acknowledges limitations, and proposes avenues for future investigation.

1.6. Ethical Consideration

This research was conducted in accordance with recognized ethical standards to protect the rights, privacy, and welfare of all participants. A range of measures was implemented to address ethical concerns during the study. Before participating, all interviewees were comprehensively briefed about the study's aims, methodologies, and their rights, and verbal consent was secured, guaranteeing voluntary and informed involvement. Participants were informed that they might exit the study at any moment without incurring any adverse repercussions.

To maintain secrecy and anonymity, all personally identifiable and company-specific information was eliminated or generalized, with pseudonyms substituted for actual names. The study maintained voluntary involvement, permitting respondents to omit questions or withdraw from the process at any point. All obtained data was securely saved on a password-protected

device and encrypted cloud storage, in compliance with GDPR and applicable data protection legislation. To avoid harm, interviews were conducted in a respectful and impartial manner, with participants told, that their replies would solely be utilized for academic purposes. Additionally, measures were implemented to mitigate researcher bias through the utilization of open-ended questions and the execution of transparent data analysis, thereby ensuring that conclusions were substantiated by evidence and academic integrity was upheld consistently.

2. LITERATURE REVIEW

Startups have emerged as a crucial component of innovation and economic development in numerous nations. Ressin (2022) highlights a strong direct relationship between startup activity and progress toward the UN Sustainable Development Goals (SDGs), demonstrating how startups contribute to economic, social, environmental, and institutional goals. This emphasizes the crucial role startups play in advancing sustainable development in various regions. Startups are crucial for the economic advancement of every nation. Nonetheless, evidence indicates that the failure rate of startups is approximately 90% (Baskoro et al., 2022a). Startups differ from SMEs and large corporations regarding the issues they encounter. Startups encounter distinct challenges, including constrained resources, high levels

of uncertainty, reliance on external financing, and often unstable internal structures (Ries, 2011; Wasserman, 2008). These challenges render startups far more susceptible than SMEs and large enterprises.

Consequently, over 90% of startups worldwide fail (Baskoro et al., 2022a; CB Insights, 2021), frequently attributable to insufficient market demand and cash flow issues. The "liability of newness" and the requirements of rapid scaling elevate their risk (Freeman et al., 1983). Resolving these difficulties is essential to improve survival rates and foster economic growth while advancing the UN Sustainable Development Goals. Therefore, it is critical to understand the startup ecosystem, challenges and opportunities associated to it.

This study mainly focuses on software startups. Software startups are a unique and exceptionally vibrant sector within the larger startup ecosystem. In contrast to manufacturing companies, which require substantial capital and depend significantly on physical infrastructure, software startups function within a landscape marked by swift technology advancement, minimal entry barriers, and frequent market fluctuations (Giardino et al., 2015). These characteristics exacerbate the conventional obstacles encountered by startups, especially concerning product-market alignment, scalability, and time-to-market constraints. Furthermore, the worldwide expansion of the digital economy has enhanced the strategic significance of software startups as catalysts for innovation, employment generation, and competitiveness (Cusumano, 2010). Considering these elements, examining software startups provides significant insights into the overarching dynamics of startup success and failure, while concentrating on a sector that is increasingly crucial to national and global economic development.

This section provides an overview of existing research on startups and software startups, various project management methodologies, and the adoption, implementation, and success rates of Agile in startup environments. It entails a thorough review of available literature, including scholarly articles, books, journals, and reliable sources. Five prominent themes are identified in the literature:

1. Definition of startups and software startups
2. The challenges faced by startups
3. Understanding of Agile management
4. Impact of agile management on digital startups

5. The challenges faced by startups while implementing agile methodologies

The goal of this review is to provide a thorough summary, a broad perspective, and identify the knowledge gaps.

2.1 Startups

Browsing through the literature, it could be said that there is no one definition of startups. Different researchers have defined it differently based on the context (size and stage), the industry and their research focus. This section covers the extensively used startup definitions to establish a strong and common baseline.

Ries (2011) describes startups as human-centered organizations created to develop innovative products or services in conditions of substantial uncertainty. Graham (2012) explains that a startup is a fast-growing company by design. He continues to add that only growth is what makes a company a startup. Other widely discussed factors, such as being newly established, technology-focused, or backed by venture funding, do not necessarily define a company as a startup. Growth is what differentiates a startup from regular companies. In a similar vein, Blank (2020) characterizes a startup as a transient organization aimed at discovering a replicable and scalable business model.

Skala (2019) elucidates that a startup's definition varies according to its developmental stage. The startup life cycle comprises three fundamental stages: inception, growth, and maturity. In the inception period, a startup is an entity with constrained resources that finds a market gap, acknowledges demand, or validates its solution. The initial phase emphasizes entrepreneurship, innovation, and technology. At the growth stage, it is an organization that grows rapidly. They concentrate on enhancing product development to bring in income and achieve rapid growth, as startups must be capable of scaling. At the maturity stage it is a hyper scalable organization. Its focus shifts to managing more complex issues and finding a way to be acquired or going public (Santisteban & Mauricio, 2017).

A pivotal element of the startup ecosystem is the importance of innovation, as multiple scholars assert that startups are fundamentally driven by entrepreneurial principles (Frederick et al., n.d.; Wooder & Baker, 2012). They assert that startups rely significantly on innovative technologies, with innovation being the catalyst for entrepreneurial creativity and a crucial prerequisite for development and enduring viability.

In summary, in the literature, three recurring themes emerge: uncertainty, growth and innovation. These are widely regarded as the key indicators of a startup. In a comprehensive analysis of the literature available on defining startups, Baskoro et al. (2022) emphasize that startups are distinguished by their high growth potential, as well as the inherent uncertainty and innovation embedded in their business models and operations.

Drawing from the literature and prevailing definitions, a startup can be characterized as a transient, innovation-oriented entity operating under conditions of significant uncertainty, with the primary goal of identifying and rapidly scaling a replicable business model. Startups are defined not only by their novelty or technology orientation but also by their pronounced focus on innovation, entrepreneurial drive, and rapid expansion. They generally advance through several life cycle phases from problem identification and solution validation to market expansion and ultimate scalability. During these periods, entrepreneurs rely significantly on ongoing innovation and flexibility to endure, expand, and generate enduring economic and social influence.

While the definition of a startup highlights its dynamic characteristics, startups exhibit considerable variation in size and structure. They are typically categorized as micro, small, medium, or large enterprises. Szarek and Piecuch (2018) in their study of impact of startups on innovative economies, classified the startups depending on factors like annual revenue and team size. A micro-startup is defined as a business with no more than 10 employees and an annual net income of up to two million euros. A small enterprise consists of up to 50 employees and generates less than ten million euros per year, while a medium-sized enterprise has a workforce of no more than 200 employees and an annual net income of under fifty million euros. Startups often fall within the SME sector. Figure 1 shows a comparison of different company sizes.

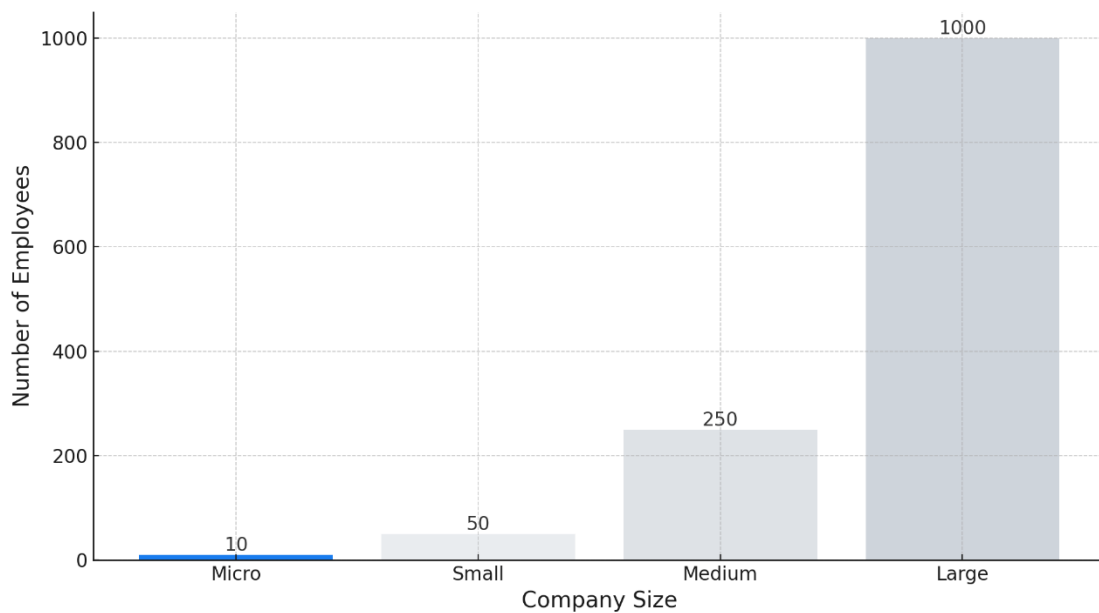


Figure 1. Comparison of company sizes by # of employees

However, it is essential to distinguish between startups and small and medium-sized enterprises (SMEs). Although in the literature the terms are often used interchangeably, Krishnan et al. (2020) through a systematic review of 15 peer-reviewed studies, highlight that startups and SMEs share certain similarities but also exhibit distinct differences. While both startups and SMEs are newly established small businesses as discussed by Szarek and Piecuch (2018), their objectives differ significantly. Startups prioritize rapid growth and scalability, whereas SMEs focus on stability, profitability, and long-term sustainability. Additionally, startups are highly innovation-driven, leveraging new technologies to disrupt industries, while SMEs typically operate with traditional technologies, making gradual improvements over time.

While understanding the general definition of startups and their distinction from SMEs offers valuable foundational context, a comprehensive understanding of the startup ecosystem also requires considering how the nature and characteristics of startups vary across industries. Among different industries, software startups represent a particularly distinctive subset, characterized by unique operational dynamics, innovation pathways, and growth trajectories. Given the specific focus of this research, it is crucial to clearly define what constitutes a software startup. Carmel (1994) was the first to introduce the concept of a software startup. He argued that software itself is emerging as a standalone product rather than just being a supplementary tool or component. He explained how software is gaining recognition not just as a part of hardware systems or larger solutions but as a standalone entity that could be

developed, marketed, and sold as a finished product. Since then, researchers have been defining the term software startups.

Software startups possess the fundamental characteristics of conventional businesses engaged in non-technological products and services. The landscape of software startups is increasingly arduous due to the swift advancement of technologies and the emergence of new devices. It exacerbates the already intricate nature of startups (Unterkalmsteiner et al., 2016). They must employ advanced tools and methodologies to create breakthrough software products and services (Sutton, 2000). According to the authors, software startups face numerous challenges stemming from limited resources, early-stage development, a range of external influences, rapidly evolving technologies, and unpredictable markets. Coleman and O'Connor (2008) describe software startups as distinctive ventures that create software using diverse methods, often without adhering to a fixed or standardized process.

Therefore, drawing from this comprehensive analysis of startups and software startups, a micro software startup can be defined as innovation-driven organizations with a maximum of 10 employees, primarily focused on developing standalone software products or services amid significant uncertainty. These startups prioritize rapid growth, scalability, and continuous innovation, operating under resource constraints and volatile market conditions without strictly adhering to standardized software development practices.

2.2 Challenges in Startups

As discussed in the last section, startups primarily target market inefficiencies and endeavor to accelerate revenue growth with minimal resources. Venczel et al. (2024) in their paper refer to a study that shows the analysis of 80 startups, among which only 10% startups will make it through their first year. Different startups face different challenges at their different life stages. Startups frequently possess constrained resources regarding personnel and capital, operating under stringent timelines. Moreover, they are typically exploratory, devoid of explicit needs, clients, and even business models. Consequently, efficiency and systematic approaches are vital; efficiency pertains to reducing development work while enhancing value, and systematic refers to the ongoing validation of whether the developed output yields customer value (Bosch et al., 2013).

The obstacles encountered by micro startups include poorly defined and implemented processes, rapidly changing requirements for relevance, definition, and content, as well as the necessity to include new or differing criteria. Moreover, the roles remain ambiguously defined, resulting in either all employees engaging in all processes or a single individual with the requisite skills and so independently overseeing certain tasks. Micro-startups frequently have challenges in determining the pertinent needs and in fulfilling these requirements in a targeted manner (Kern et al., 2024). In their comprehensive study on early-stage software startups, Giardino et al. (2015) categorize the challenges these ventures face into four key areas: financial, market-related, product development, and team dynamics.

2.2.1 Financial Constraints

Early-stage startups mostly face financial issues as they are still developing their product and don't have paying customers. Several articles (Giardino et al., 2015; Melegati & Kon, 2020) refer to financial constraints being one of the major limitations for early-stage startups. They want to hack income streams as soon as possible without burning their runway. This creates a vicious cycle, where startups must develop their product to attract customers, yet securing sustainable finances is essential for product development.

2.2.2 Work-Life Balance

Kern et al. (2024) identified that the lack of clearly defined roles and duties in startups frequently results in overlapping activities and redundant efforts, significantly increasing the likelihood that team members are functioning at double their intended capacity. Unsustainable workloads markedly elevate the risk of burnout, jeopardizing individual well-being and total team effectiveness. Burnout is a significant issue in high-pressure settings such as startups, where extended hours and incessant demands can rapidly diminish motivation, creativity, and productivity. Maintaining a healthy work-life balance is vital for preserving staff engagement and ensuring venture success. According to Lenka (2020), the indirect consequences of burnout likely had a significant role in the underperformance and failure of startups.

2.2.3 Market Competition

In the highly competitive software industry, time-to-market is increasingly crucial for startup success. Rapid imitation by competitors is a threat also, after the business idea is introduced during development. In a case study of two early-stage startup failures, Giardino et

al. (2014) noted that the startups pursued product/market fit strategies too early, despite users not yet being fully convinced of the product. Another major challenge identified is understanding the ever-changing market needs. Razdan and Kambalimath (2019) explain how these changing dynamics impose a market risk to product development, and make the team question: even with a product, does it serve a market?

2.3 Agile Management

Over the past century, project management methodologies have undergone significant evolution, driven by shifting business landscapes, technological progress, and the growing complexity of projects. Efficient project management is essential for making certain that initiatives are executed on time, within financial constraints, and in accordance with quality benchmarks. As organizations strive to improve efficiency, agility, and the satisfaction of stakeholders, selecting the appropriate project management approach has become a pivotal decision. Among the most widely adopted approaches today are the Waterfall and Agile methodologies. Winston W. Royce originally introduced the Waterfall project management approach in 1970 (as cited in Royce, 2021). Waterfall management approach is linear and structured. It heavily focuses on pre-planning of all the phases of the project, and each phase depends on the next one i.e. phase A has to be completed before moving to phase B, hence the linear and sequential approach. Its systematic nature makes it suitable for projects where requirements are well-defined and unlikely to change (for example construction projects). This approach can provide clarity and predictability in a suitable environment with a better resource management.

However, with rapid technological advancements and evolving customer demands, the Waterfall methodology struggles to keep pace with this dynamic environment in software development. In case of software development, waterfall poses to be a rigid process, not giving enough room for iterative improvements. Sirisha et al. (2024) identify numerous issues related to the waterfall methodology in their research. Their analysis indicates that the approach provides less flexibility, complicating the incorporation of modifications once work has commenced. The methodology frequently lacks client engagement during the process, resulting in a disconnect between the finished product and user requirements. The research highlights that waterfall programs entail significant risk, especially when problems are usually detected late in the development process.

Furthermore, the protracted development cycles characteristic of this strategy may postpone product delivery and diminish responsiveness to market demands. Sirisha et al. (2024) contend that the waterfall methodology is typically inappropriate for complicated projects necessitating iterative development and ongoing input. Collyer et al. (2010), through interviews with 31 project managers from 10 different industries, found that traditional project management methods often fell short in fast-changing environments. They identified three main types of changes that commonly disrupted projects: 1) shifts in project objectives, 2) changes in materials, resources, tools, or methodologies, and 3) evolving interdependencies with other relevant projects, services, or products. These challenges create further delays and inefficiencies making it hard to meet the growing customer needs and technical advancements in software industry.

To cater these needs, a more iterative and flexible project management approach was needed which led to the development of agile methodology in the early 2000s. Agile methodology was formalized by a group of software engineers in the form of ‘Agile Manifesto’. Figure 2 describes the four fundamental ideals of the Agile Manifesto: prioritizing humans and interactions over processes and tools; valuing working software over extensive documentation; emphasizing customer collaboration over contract negotiation; and favouring responsiveness to change over adherence to a plan.

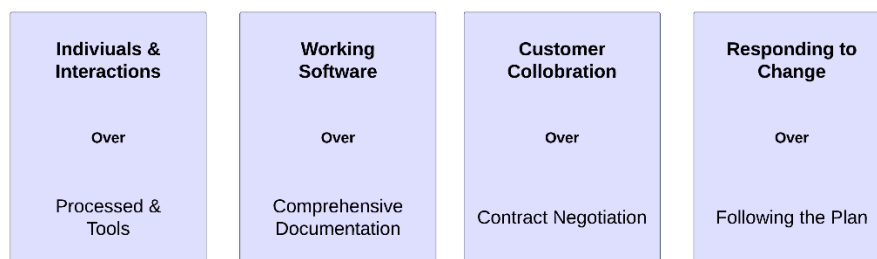


Figure 2. Agile Manifesto. Adapted from productplan.com

The Agile technique employs an iterative approach characterized by brief development cycles, swift feedback loops, and the ongoing incorporation of code modifications into the result. Thus, Agile emphasizes the delivery of outcomes while minimizing superfluous documentation, planning, and processes that do not enhance customer value.

The primary distinction between Agile and Waterfall pertains to their approaches to change and adaptability. As previously mentioned, the waterfall methodology is inherently rigid and lacks flexibility. In contrast, Agile is intrinsically flexible, rendering it more appropriate for

projects necessitating frequent modifications and characterized by significant uncertainty. Table 1 shows the fundamental distinctions between waterfall and agile methodologies.

Table 1 Comparison between Waterfall and Agile

	Waterfall	Agile
Approach	Linear and sequential	Iterative and incremental
Process Structure	Fixed phases (requirements, design, implementation, testing, deployment)	Continuous cycles (sprints or iterations)
Flexibility	Limited flexibility after project initiation	Highly flexible, allowing for frequent changes
Customer Involvement	Minimal customer involvement after the initial phase	Ongoing customer collaboration throughout the process
Documentation	Heavy documentation required	Minimal documentation, focusing on working software
Testing	Testing occurs after development is complete	Testing is integrated throughout the process
Delivery	Single delivery at the end of the project	Frequent releases of smaller, working parts
Risk Management	Higher risk due to delayed feedback and testing	Lower risk due to continuous feedback and adjustments

Agile methodology includes specific approaches (detailed in Figure 3) that help teams implement the principles and values outlined in the Agile Manifesto. Mkpojiogu et al. (2019) observe that although numerous strategies exist, specific procedures are notably prevalent among startups. Their research emphasizes Scrum, Lean methods, Extreme Programming (XP), Dynamic Systems Development Method (DSDM), and Feature Driven Development (FDD) as the most prevalent methodologies.

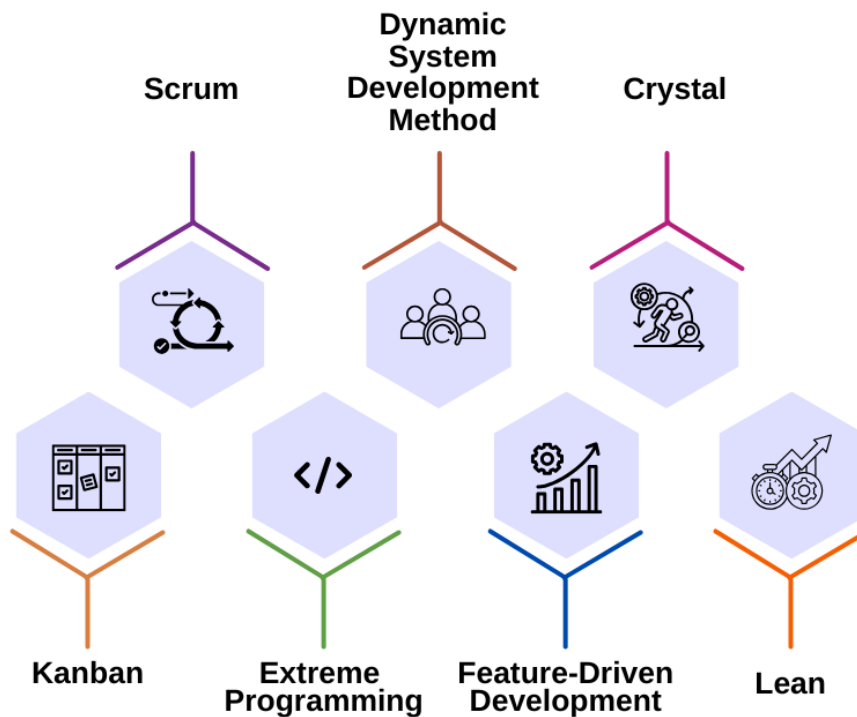


Figure 3. Types of Agile Methods

These methodologies are preferred for their focus on adaptability, swift iteration, and customer-centric development, rendering them particularly appropriate for the volatile and unpredictable contexts in which startups generally function. Verwijs and Russo (2023) in a market research study, reported that that 81% of organizations utilized Scrum in their Agile practices. For context, Scrum is briefly explained to establish a solid foundation for understanding Agile implementation in startups, given its dominance among the top five Agile methodologies.

Scrum is an Agile framework that organizations and teams use to adopt Agile principles in practice. It enables teams to remain flexible, adapt to change, and iterate efficiently throughout the development process. As per Altuwajiri and Ferrario (2022), Scrum is a framework designed to help teams and organizations create value by developing adaptive solutions to tackle complex challenges. To support and achieve this, the team must adhere to 5 scrum values, which are: commitment, focus, openness, respect, and courage (Panzavolta, 2019). Scrum enables teams to break down larger objectives into smaller, manageable goals, which are completed within fixed timeframes known as sprints. Scrum is executed with the help of multiple ceremonies to ensure project transparency and track the progress, these ceremonies are

1. Sprint Planning
2. Daily Standup
3. Sprint Retrospective

4. Backlog Refinement

The process commences with sprint planning, during which the team delineates the sprint backlog a compilation of activities designated for completion within a specified timeframe, generally ranging from two to four weeks. During the sprint, daily stand-up meetings facilitate progress tracking, challenge resolution, and effort alignment. Upon the conclusion of the sprint, a sprint review is performed to assess deliverables and obtain stakeholder input. A sprint retrospective is conducted to evaluate the workflow, pinpoint areas for enhancement, and strategize improvements for the subsequent iteration.

The essential component of Scrum is a compact, cross-functional, self-organizing, and empowered team that depends on an exceptionally communicative and collaborative culture (Kadenic et al., 2022). The Scrum team is composed of a Scrum Master, a Product Owner, and developers, with no sub-teams or hierarchical divisions. The artifacts produced in a sprint are product backlogs, sprint backlogs, and increments (Altuwaijri & Ferrario, 2022).

Scrum is often combined with Kanban (Scrumban), where the scrum master uses Kanban boards with swim lanes depending on the needs of the projects. For simpler projects, there are 3 swim lanes, backlog, in progress and completed however, there can be more depending on the needs for example Blocked, QA, etc.

Kanban is a widely utilized paradigm for visualizing Agile software development. Work items are clearly depicted on a kanban board, enabling team members to observe the status of any task at any moment. In 2004, Kanban was included into the Agile framework when David Anderson implemented it while supporting a software development team at Microsoft. Due to the visualized project processes, Kanban plays a vital role in increasing communication and cooperation among team members (Banijamali et al., 2017).

It all together forms a layered agile development framework that builds from foundational values to specific methods, ensuring alignment between team processes and organizational goals. As Figure 4 depicts the framework, at the foundation are the Core Values from the Agile Manifesto such as valuing individuals and interactions, working software, customer collaboration, and responding to change. These values support a set of Agile Principles, which emphasize delivering valuable software early and continuously, embracing changing requirements, and fostering self-organized teams. Based on these principles, teams adopt Agile Practices like daily meetings, retrospectives, and planning sessions, while defining specific

Roles such as Scrum Master and Product Owner. These roles and practices are applied within various Agile Methods, such as Scrum and Extreme Programming (XP), to structure and guide the development process effectively. The upward arrows indicate how each level builds on the previous one to support adaptive, efficient software delivery.

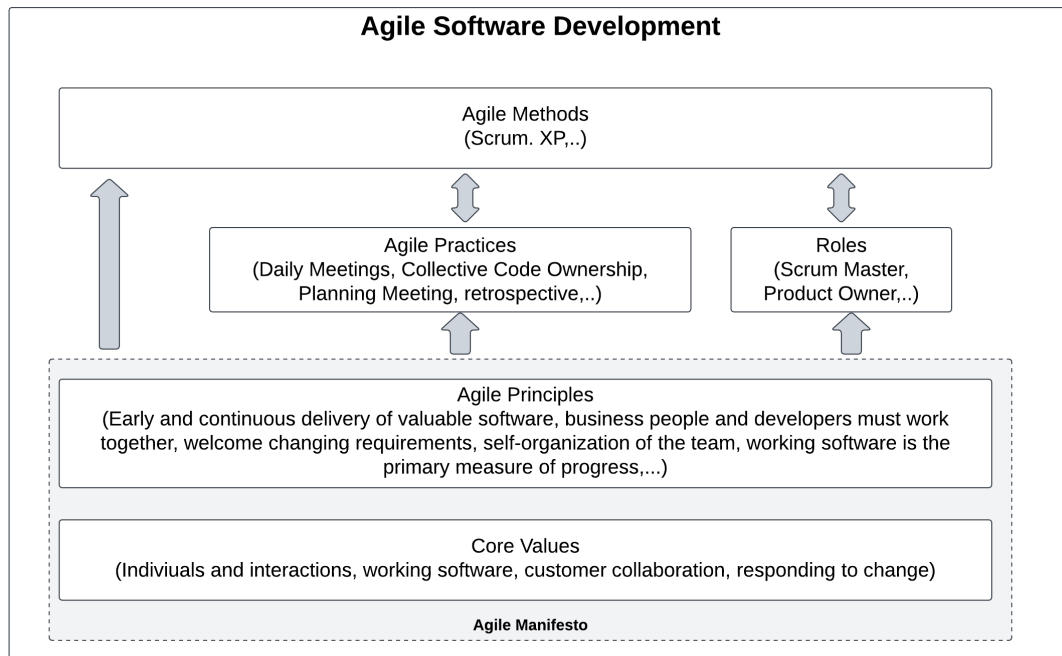


Figure 4. Agile Software Development. Adapted from *Modelling the Impact of Culture on Agile Methods* by S. Ali et al., 2024, P 27

It is essential to recognize that agility can be a goal at any phase of a startup's journey. For instance, startups strive to maintain agility not only during the development phase but also when conceiving an idea and ensuring its alignment with product-market fit. Several studies in the literature, such as Bosch et al. (2013) propose a comprehensive framework based on Lean management, enabling software startups to explore multiple product ideas simultaneously. This approach enhances efficiency and reduces time wastage. However, once an idea is conceived and development begins, Agile methodologies such as Scrum, Kanban, and others play a crucial role in ensuring a swift, efficient, and effective development process.

2.4 Impact of Agile Management on Software Startups

As discussed in detail in 2.1 and 2.2 that software startups are working in an industry that is changing dynamically, with the advent of new technologies, Project management methodologies like agile management helps startups to adapt quickly and stay flexible. Agile can have a great impact on the overall efficiency and product development of software startups.

In a thorough study of 1,002 projects, Serrador and Pinto (2015) investigated whether Agile works and concluded that there is a clear positive correlation between the adoption of Agile or iterative approaches and project success. Projects that reported greater use of Agile practices also tended to report higher success rates.

Agile methodologies foster trust and empathy within teams. Developers actively participate in the planning phase, gaining a deep understanding of user needs. This trust encourages open communication, enabling developers to clarify customer requirements and build software that aligns with them. Agile provides a structured approach to updating customer needs, ensuring the software adapts to evolving requirements. This incremental development approach minimizes costs associated with changes. Additionally, Agile facilitates early detection of faulty code in the software by incorporating testing in each iteration, reducing development costs by identifying and addressing errors promptly (Fergis, 2012). Developers favor Agile because it supports self-directed teams, which are essential for project success. Empowering teams helps management strengthen trust and respect. The positive work atmosphere fosters problem-solving and creativity, enabling teams to work more effectively. Additionally, Agile enhances cognitive processing, leading to greater innovation and idea generation throughout development (Eman, 2024).

Rahman et al. (2024) conducted a 12-month case study on multiple software development teams within a mid-sized technology company, analyzing productivity metrics before and after adopting Agile practices. They observed an increase in team satisfaction, development speed, and code quality. They also noticed a decrease in team stress, response time to changes, and task completion time. The primary findings indicate that the iterative and collaborative framework of agile significantly enhances team efficiency and effectiveness. The five key motivations for adopting Agile, highlighted by Mkpojiogu et al. (2019), include: the need for expedited product delivery, improved capacity to manage shifting priorities, augmented software maintainability, streamlined development processes, and enhanced delivery predictability.

2.5 Challenges of agile management in micro software startups

Although Agile approaches are frequently considered a crucial factor in project success, their efficacy is not universally applicable to all organizational structures. Small organizations have encountered an unsatisfactory implementation experience and faced significant hurdles

they were unprepared to manage. The most affected organizations are micro-startups (Putrianasari et al., 2024).

Several studies have explored the use of Agile practices in the context of startups across various industries. As Soares Bastos et al. (2019) , through 14 in-depth semi-structured interviews with CEOs and CTOs of early-stage startups, found that the most frequently used Agile practices were associated with DevOps, Agile Fundamentals, Design, and Extreme Programming (XP). However, they specifically do not address how Agile can be effectively implemented and adopted in early-stage startups. Another study conducted by Zielske et al. (2022) examines how Agile practices can be applied within the logistics industry. Nonetheless, the literature largely fails to address the challenges and practical solutions involved in implementing Agile in micro-startups.

Micro-startups are also frequently influenced by the prevailing discourse that adopting agile practices is essential for success. However, this assumption does not always hold true in practice, as these startups often encounter unforeseen challenges when implementing Agile-Scrum software development. This is primarily because agile methodologies and frameworks were originally designed for larger organizations with well-established management structures. Nevertheless, agile management and frameworks cannot be entirely disregarded, as micro-startups must compete in dynamic markets where speed and adaptability are critical. Agile methodologies, when appropriately tailored to the specific needs and constraints of micro-startups, can enhance their ability to deliver products efficiently and respond swiftly to market demands.

The challenges faced by micro startups when implementing agile methodologies in software development span through 3 domains within the organization, people, agile misconceptions, organizational misalignment (Putrianasari et al., 2024).

2.5.1 People

The literature highlights three major challenges concerning the people dimension in micro-startups when it comes to adapting Agile, namely internal communication, small team size, and resistance to change. While it may appear that communication within a small team would be straightforward, this is not always the case. In the early stages of a business, founders often have strong confidence in their vision, but may not actively communicate their initial enthusiasm with the rest of the team (Lenka, 2020). Consequently, developers and other team

members focus on their individual tasks without a clear understanding of the organization's overarching goals. This misalignment can create an environment that lacks open communication and collaboration, ultimately limiting the effectiveness of agile methodologies within startups (Abdalhamid & Mishra, 2017).

A report by PMI (2023) based on insights from 3,500 professionals, identifies communication as the most critical skill in the IT industry. Fontana and Marczak (2020) also highlight that insufficient collaboration and communication within teams significantly impact the successful implementation of Agile practices. Due to the small team size in early-stage startups, employees are often assigned to multiple projects (Raharjo and Purwandari, 2020), which increases the chances of burnout and negatively affects the overall working environment.

Additional significant concerns related to Agile adoption include resistance to change, as discussed by Fontana and Marczak (2020) The absence of participative decision-making, as noted by Raharjo and Purwandari (2020), together with deficiencies in skills, knowledge, and employee engagement, poses challenges to the effective implementation of Agile methodologies.

2.5.2 Agile misconceptions

A principal obstacle encountered by micro-startups in adopting Agile methodology is a lack of understanding of its principles and concepts Inayat et al., (2014). Numerous firms rapidly embrace this method, having observed its prior implementation and the benefits received by others. Agile is not a universal paradigm; it necessitates contextual comprehension and intentional adaptation to align with an organization's maturity and operational limitations. Micro-startups, frequently constrained by time and budgetary limitations, typically use Agile frameworks such as Scrum without adequately tailoring their procedures to align with their scale and culture. This may lead to excessively restrictive ceremonies, misaligned sprint planning, and superficial retrospectives that do not yield significant change Fontana & Sabrina Marczak, (2020).

2.5.3 Organization Misalignment

At the organizational level, insufficient training constituted a significant issue; organizations were inadequately prepared for the training Abdalhamid and Mishra (2017). Additional issues to contemplate included the absence of official direction for the process, ambiguous role definitions for quality control and evaluation, and elevated work pressure for all personnel. Many micro-startups operate with flat hierarchies and limited managerial oversight, which can be both an advantage and a liability. While flat structures support rapid decision-making and flexibility, they can also result in a lack of accountability and role confusion when adopting structured Agile frameworks.

Another organizational hurdle is the misalignment between leadership vision and Agile execution. Founders may promote agility as a strategic value without allocating the necessary time and resources to support its implementation, leading to superficial buy-in and sporadic adoption. Agile transformation requires cultural shifts such as openness to feedback, trust in self-organized teams, and a tolerance for iterative failure elements that can be difficult to embed in the fast-paced, survival-focused environment of micro-startups (Venczel et al., 2024).

2.6 Literature Summary

This literature review has conducted a comprehensive examination of the intricate and evolving characteristics of startups, emphasizing the difficulties encountered by software startups and the function of Agile management approaches in mitigating these obstacles. Startups, distinguished by their emphasis on innovation, swift expansion, and considerable unpredictability, are essential to contemporary economies and significantly contribute to technical progress and the attainment of overarching societal objectives, including the UN Sustainable Development Goals.

The review initially elucidated the notion of startups and underscored the distinctive role of software startups within this domain. In contrast to conventional SMEs, startups emphasize scalability and disruptive innovation, functioning in extremely dynamic situations where adaptation is essential. Software companies face rapid technical advancements, constant market fluctuations, and considerable resource limitations, rendering their journey to success particularly perilous. The primary hurdles faced by startups were categorized as financial limitations, team dynamics, market competitiveness, and the essential requirement for attaining

product-market fit. These issues are further intensified in micro-startups, where organizational procedures are underdeveloped, and resource constraints are more evident.

In this context, the literature advocates for the implementation of Agile management strategies to address the deficiencies of conventional project management methods such as Waterfall. Agile provides adaptability, iterative progression, and customer-focused methodologies that are more aligned with the dynamic and unpredictable nature of startups. The tenets of Agile adaptability to change, close customer engagement, regular delivery of functioning products, and team autonomy correspond effectively with the requirements of startups seeking swift learning and adaptation.

The beneficial effects of Agile on startup performance are well-established, with empirical research indicating enhancements in project success rates, development velocity, customer happiness, and team morale. Methodologies like Scrum and Kanban have demonstrated notable efficacy in enhancing transparency, facilitating collaboration, and guaranteeing the continuous delivery of value. Furthermore, Agile approaches augment cognitive processing among teams, foster invention, and cultivate a culture of trust and autonomy essential for creative problem solving in companies.

Nonetheless, the research underscores considerable obstacles in the implementation of Agile methodologies within micro-startups. These hurdles encompass interpersonal issues (such as communication deficiencies and resistance to change), procedural obstacles (including superficial comprehension and inadequate application of Agile concepts), and organizational impediments (notably the absence of formal training and role ambiguity). Many Agile methods were initially developed for larger, more structured businesses, resulting in a misalignment when immediately implemented in micro-startups without suitable modification. Despite Agile's popularity, the literature indicates a significant research gap: numerous studies investigate Agile implementation and startup difficulties independently, while few explore their intersection specifically, how Agile may be effectively adapted and optimized for micro-scale startups. A more nuanced understanding is required regarding how micro-startups might surmount the challenges of Agile adoption to fully realize its advantages.

This study draws on ideas from the literature to further examine the specific demands, difficulties, and potential solutions for Agile management in micro-startups. This initiative is

to provide practical guidance for early-stage software startups aiming to effectively utilize Agile principles for sustained growth and success.

3. RESEARCH DESIGN

This chapter explains the research design employed to investigate the obstacles encountered by micro software startups in the implementation of Agile methodologies and to formulate a customized conceptual Agile adoption framework (see Figure 5, the Research Onion). It begins by clarifying seven essential methodological choices: research philosophy, approach, strategy, temporal perspective, sampling strategy, data collection methods, and data analysis techniques. Every option is assessed in relation to the study's objectives and research inquiries. The chapter finishes with an examination of inherent restrictions and a summary of the chosen design.



Figure 5. Research Onion. Adapted from Saunders et al., (2023)

3.1. Research philosophy

The research employed a pragmatic philosophy, emphasizing the resolution of practical issues rather than rigid conformity to positivist or interpretivist frameworks. Pragmatism corresponded with the study's dual objectives of examining lived experiences (RQ1 and RQ2) and suggesting practical improvements based on the conceptual framework (RQ3). This attitude afforded the adaptability to choose methodologies that most effectively addressed each research question, facilitating a blend of deductive and inductive strategies. Though pragmatism might produce methodological eclecticism, I addressed this by systematically assessing methodological selections based on two criteria: practical relevance and congruence with research aims. This guaranteed methodological consistency and rigor throughout the investigation.

3.2. Research Approach

A hybrid coding approach combining deductive and inductive methods was used in the study (Fereday & Muir-Cochrane, 2006). First, five carefully chosen academic papers were analysed using deductive coding based on known theoretical frameworks. Insights from this analysis were then used to develop a preliminary codebook. This codebook subsequently served as a guide for the inductive open coding of the interview data, allowing new themes and ideas to emerge directly from the participants' experiences. This integrated methodology

guaranteed that the study was theoretically grounded while being attuned to context-specific circumstances, so enhancing both theoretical rigor and empirical validity.

3.3. Research Strategy

Multiple-case qualitative research strategy was adapted, investigating 7 micro software startups through semi structure interviews, as individual cases. This strategy enabled an in-depth exploration of each startup's Agile adoption experiences, while cross-case analysis facilitated the identification of common patterns and significant variations. The multiple-case design enhanced the analytical generalizability of the findings through replication logic, whereby consistent results across cases strengthened the credibility of the emerging conceptual framework. The choice of a qualitative, multiple-case strategy was particularly appropriate for addressing the study's objectives: exploring obstacles, understanding adaptation processes, and developing a conceptual framework to aid Agile adaptation in micro startup environments.

3.4. Temporal Perspective

The study employed a cross-sectional temporal framework, gathering participants' retrospective and current reflections within a specified data collection time. This snapshot methodology corresponded with the project's academic timeline and facilitated the development of comprehensive, thorough narratives regarding Agile adoption experiences.

While cross-sectional designs naturally restrict the capacity to directly monitor temporal changes inside participants' organizations, the five academic publications examined in the deductive phase offered significant insights into the historical evolution and application of Agile methodologies in comparable situations. These secondary sources served as temporal references, augmenting the empirical data by providing insights into the evolution of Agile techniques in micro and small startup contexts. Consequently, by integrating retrospective interviews with insights from existing literature, the study achieved a more refined temporal comprehension while adhering to the limitations of a cross-sectional methodology.

3.5. Sampling Strategy

A purposive sampling method was utilized, focusing on those directly engaged in Agile adoption inside micro software startups, such as founders, project managers, team members, and investors. This method guaranteed that participants had direct, pertinent experience essential for addressing the research inquiries. To augment the diversity and transferability of

the findings, the recruitment was broadened beyond researcher's direct professional network by disseminating participation invitations via startup forums and LinkedIn groups. This technique reduced selection bias and encompassed a wider array of viewpoints across many technology industries.

3.6. Data Collection

Data were gathered through semi-structured interviews and literature analysis. The semi-structured interviews enabled a thorough analysis of the problems, adjustments, and perceived successes related to Agile adoption in micro software startups. The interview questions (Appendix B1) were formulated utilizing ideas from the theme literature research and executed a pilot interview to enhance their clarity and pertinence.

Alongside interviews, a literature analysis of five meticulously chosen scholarly articles were conducted. These studies offered additional insights into the development of Agile practices and the wider contextual aspects affecting Agile adaptation in micro startup settings. This secondary data source enhanced the empirical findings and facilitated triangulation.

3.6.1 Sample of Literature

Five basic scholarly publications were selected for hybrid deductive-inductive coding to anchor the study in recognized theory and ensure methodological rigor. These papers were specifically chosen for their relevant relevance to the aims and research questions of the study, therefore offering a wealth of information on the challenges of Agile adoption, motivating factors, and contextual dynamics in software companies.

For instance, Abdalhamid and Mishra (2017), in their study *Factors in Agile Methods Adoption* (P1), identified critical success and failure criteria directly related to Objective 1 and Research Question 1, which address the difficulties faced by micro software companies.

The research conducted by Mkpojiogu et al. (2019), referred to as P2, focused on software startups, examining the motives for Agile adoption—an essential aspect for understanding the relevance of Agile frameworks in micro settings (Objective 2 and RQ2).

The systematic review by Putrianasari et al. (2024) (P3) elucidated the challenges associated with Agile-Scrum implementation in small startups, offering a detailed examination of the contextual difficulties typically encountered by micro startups.

The quantitative analysis by Prasongko and Adianto (2019) referred to as P4, provided empirical validation for the efficacy of Agile methods by examining both their strengths and limitations, thereby reinforcing assessments of Agile appropriateness (Objective 2 and RQ2).

Finally, Serrador and Pinto (2015)(P5) offered insights into Agile leadership models, which are essential for understanding the strategic implementation adjustments required for effective Agile adoption frameworks, directly corresponding to Objective 3 and RQ3.

These articles collectively offer strong theoretical and empirical foundations to methodically tackle the study objectives and suggest a framework for Agile adoption. The selected literature is displayed in Table 2 below.

Table 2 Literature for Hybrid Coding

ID	Title	Authors	Year
P1	Factors in Agile Methods Adoption	Samia Abdalhamid, Alok Mishra	2017
P2	Software Startups: Motivations for Agile Adoption	Emmanuel Mkpojiogu et al.	2019
P3	Problems in The Adoption of Agile-Scrum Software Development Process in Small Organization	Rahmawati Putrianasari et al.	2024
P4	The Role of the Agile Leadership Model as a Competitive Advantage for the Future Leader in the Era of Globalization and Industrial Revolution 4.0	Angga Prasongko, Tri Adianto	2019
P5	Does Agile work? — A quantitative analysis of agile project success	Pedro Serrador, Jeffrey K. Pinto	2015

3.6.2 Participants Overview

Concurrently with the literature-based research, primary empirical data were gathered via semi-structured interviews with seven persons occupying various roles within micro software startups. Each transcript was methodically coded utilizing the established codebook (v1), formulated from the analysed literature. This participant sample provided diverse

perspectives from founders and product managers to developers and investors yielding a comprehensive understanding of Agile processes in micro-scale organizations. Appendix B2 provides a comprehensive demographic profile, encompassing each participant's role, years of experience, and the size of the startup while Table 3 gives a snapshot of participants demographics.

Table 3 Participants Demographics

TranscriptID	Role
T01	Founder
T02	Founder
T03	Investor
T04	Product Manager
T05	Product Manager
T06	Team Member
T07	Team Member

3.7. Data Analysis

The thematic analysis of the data was conducted following Braun and Clarke's six-phase framework (Braun & Clarke, 2006). Initially a deductive coding framework was employed, derived from the literature-based codebook while remaining receptive to the inductive identification of new codes from the interviews. Subsequently, cross-case synthesis was conducted to analyse commonalities among companies, revealing common obstacles and distinct adjustments. Figure 6 visually depicts the stepwise pragmatic hybrid methodology that guided this study.

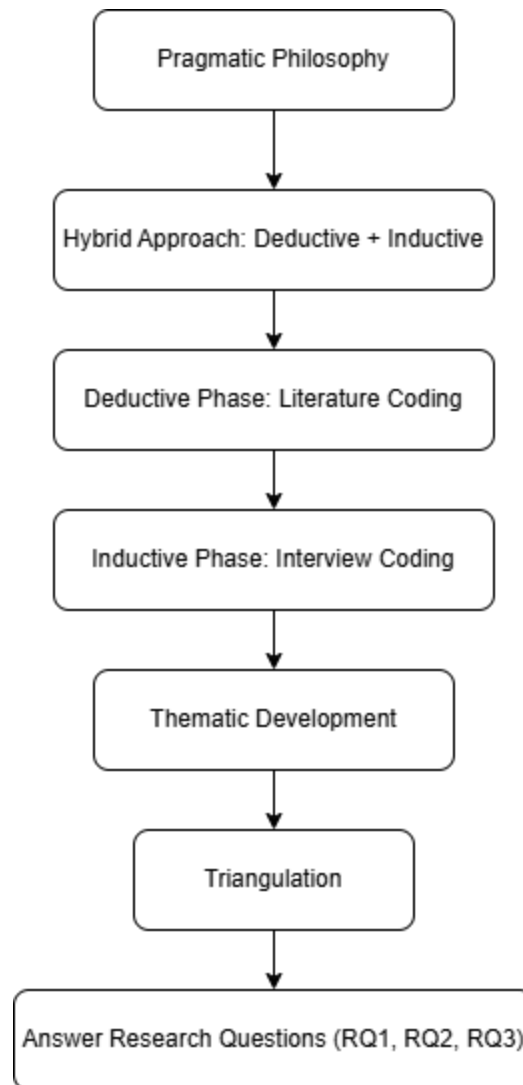


Figure 6. Stepwise pragmatic hybrid methodology

3.8. Constraints

Despite its benefits, the selected design has limitations. The cross-sectional, qualitative case methodology precludes statistical generalization; rather, the results offer analytical generalizability to alternative contexts. Secondly, although deliberate and diverse recruitment reduces bias, self-selection and access limitations may skew sample composition. Third, retrospective interviews are prone to recall bias; however, document analysis can mitigate, albeit not completely eradicate, this concern.

3.9. Synopsis and Transition

This chapter has thoroughly elucidated and substantiated each methodological decision in accordance with the study's objectives to identify difficulties (RQ1 & RQ2) and to develop

a feasible conceptual Agile framework (RQ3). The design incorporates many qualitative methodologies inside a pragmatic framework, yielding a synthesis of depth, flexibility, and rigor. The forthcoming chapter will elucidate the utilization of data collection and analysis techniques and show the resultant insights.

4. RESULTS AND FINDINGS

4.1 Overview of Analytical Approach

This chapter presents the conclusions of the thematic analysis employing a hybrid (deductive-inductive) coding style, examining how micro software startups perceive and implement Agile methodologies. The analytical procedure began with the creation of a codebook, directly based on five seminal scholarly works. The codebook was methodically applied to the primary data, consisting of seven interview transcripts. The use of these codes revealed initial patterns and insights. Subsequent to coding, the coded data were aggregated into overarching meta-themes grounded in conceptual affinity and related patterns identified across the segments. Every emergent theme was then closely matched with the main interview material and the body of existing research. With academic debate, this triangulated approach helped to enable a thorough examination of the alignment or divergence of views.

Relevant coverage statistics, frequency tables, and representative participant quotes accompany the results. Included in the appendices are comprehensive coding matrices, participant demographics, and careful literature triangulation summaries to increase auditability, rigor, and openness.

4.2 Data Analysis Results

4.2.1 Literature Coding Analysis

Following the selection of key literature sources (discussed in 3.6.1) a systematic coding process was applied to extract pertinent codes and thematic insights. Cross-referencing was conducted to verify the consistency of emerging patterns with recognized Agile success factors, motivational influences, and process challenges. The resulting code set (shown in Table 4) served as the analytical foundation for triangulating theoretical models with empirical findings. Detailed code definitions are provided in Appendix A1, and their distribution across the literature corpus is illustrated in Appendix A2.

Table 4 Initial Code set

Code Label	Papers
------------	--------

Organisational Culture & Readiness	P1 P3
Management Support & Sponsorship	P1 P3
Agile Leadership Style	P1 P4
Team Capability (Skills & Experience)	P1 P5
Team Collaboration & Communication	P1 P3 P4
Team Morale & Motivation	P2
Customer Involvement & Collaboration	P1 P3
Change Resistance & Adaptability	P1 P3 P4 P5
Process Tailoring & Clarity	P1 P2 P3
Project Planning & Vision	P1 P5
Tracking & Transparency	P1 P2

Tools & Technical Infrastructure	P1 P3
Project Suitability (Characteristics & Complexity)	P1 P2 P5
Cost & Risk Management	P2
Rapid Delivery & Time-to-Market	P1 P2
Handling Changing Requirements	P1 P2 P5
Product Quality & Maintainability	P1 P2
Process Simplicity & Efficiency	P1 P2 P5
Business Alignment & Customer Satisfaction	P1 P2 P5
Distributed Team Management	P2

A total of 20 distinct codes were identified through the analysis of the selected literature sources. Numerous codes appear consistently across many articles; for example, Organizational Culture and Management Support are repeatedly identified as significant factors influencing adoption in micro startups. Certain concepts overlap or interconnect, such as change. Resistance frequently stems from an inflexible culture or inadequate leadership, signifying the interdependence of the codes. The motivation-focused study (P2) highlights distinct advantages such as Team Morale and the management of Distributed Teams, which

were not explicitly addressed in the challenge-oriented publications. In contrast, the SLR (P3) emphasized cultural concerns and top management commitment as essential, corresponding with the failure reasons identified in P1, while advocating for research into leadership, a gap partially addressed by including an agile leadership approach (P4).

The overlaps and gaps indicate that although some underlying concerns (such as culture, support, and process adaptation) are widely recognized, aspects like leadership style and team morale in micro-startups require further examination.

4.2.1.1 Cross-Paper Insights

From the five studies, a clear, multi-layered picture of how Agile is used in micro software startups is formed. Organizational culture and support from top management come up over and over as foundational enablers or barriers. Even the strongest processes have trouble taking root without a culture that welcomes change and leaders who actively support Agile. In addition, a flexible leadership style which includes making decisions quickly, being willing to take risks, and giving teams a lot of freedom shows how to put an organization's goals into action in the real world. In small startups where every choice has a big effect, leaders who act in an Agile way not only make the change easier, but they also make it more ingrained.

Technical know-how and good communication are essential at the team level. Papers that look at failure factors stress that Agile projects often fail because people don't have the right skills or don't work together well. On the other hand, a quantitative study shows that skilled teams and clear communication between members improve sprint success and stakeholder satisfaction. One study is the only one that looks at the morale and motivation of very small, often distributed teams. These micro-startups depend on teamwork and enthusiasm, and the fact that they can work together from different places shows both a challenge and an opportunity for customized Agile ceremonies and simple collaboration tools.

4.2.1.2 Reflexive Thoughts

Upon reflection of the literature coding process, it became apparent that the adoption of Agile methodologies in micro software startups involves not only the implementation of

technical practices but also the navigation of team dynamics and cultural transformations. Culture, leadership, and team motivation are profoundly interrelated aspects. To more effectively encapsulate overlapping concepts, comprehensive synthesized codes such as Business Alignment and Customer Satisfaction were developed, incorporating related topics like as value realization and customer participation.

4.2.2 Interview Coding Analysis

The coding of interview transcripts was performed utilizing the established codebook (v1) (Appendix A1), originally developed from the literature analysis. Although most participant responses were consistent with established codes, many infrequent notions, including "Agile Ritualism," "Training," and "Creative Communication," surfaced during the research. After thorough assessment, these novel concepts were found to conceptually intersect with established categories and were consequently incorporated into overarching codes such as "Process Tailoring & Clarity," "Team Capability (Skills & Experience)," and "Team Collaboration & Communication." This methodology-maintained coding consistency while encapsulating the intricate practices and modifications articulated by participants.

This hybrid methodology integrating deductive frameworks with receptiveness to inductive insights allowed the codebook to preserve both conceptual integrity and contextual adaptability. Since no entirely new codes were necessary and all emerging themes could be consistently integrated, the initially literature-derived codebook (v1) was validated as stable and thereafter designated as Codebook v2(Appendix C1) for the duration of the research.

The cumulative frequency of each code from the seven interviews is presented in the Table 5 below. The accompanying heatmap (Figure 7) illustrates the distribution of code application among participants, with each cell representing the quantity of segments allocated to a specific code.

Table 5 Code Frequencies in interviews

Code Label	Frequency (Total Mentions)
Process Tailoring & Clarity	6
Project Planning & Vision	5

Team Collaboration & Communication	5
Team Capability (Skills & Experience)	3
Team Morale & Motivation	3
Tracking & Transparency	4
Tools & Technical Infrastructure	3
Process Simplicity & Efficiency	5
Agile Leadership Style	5
Change Resistance & Adaptability	3
Handling Changing Requirements	2
Management Support & Sponsorship	2
Rapid Delivery & Time-to-Market	1
Cost & Risk Management	1
Distributed Team Management	2
Project Suitability (Characteristics & Complexity)	3
Customer Involvement & Collaboration	1
Business Alignment & Customer Satisfaction	1

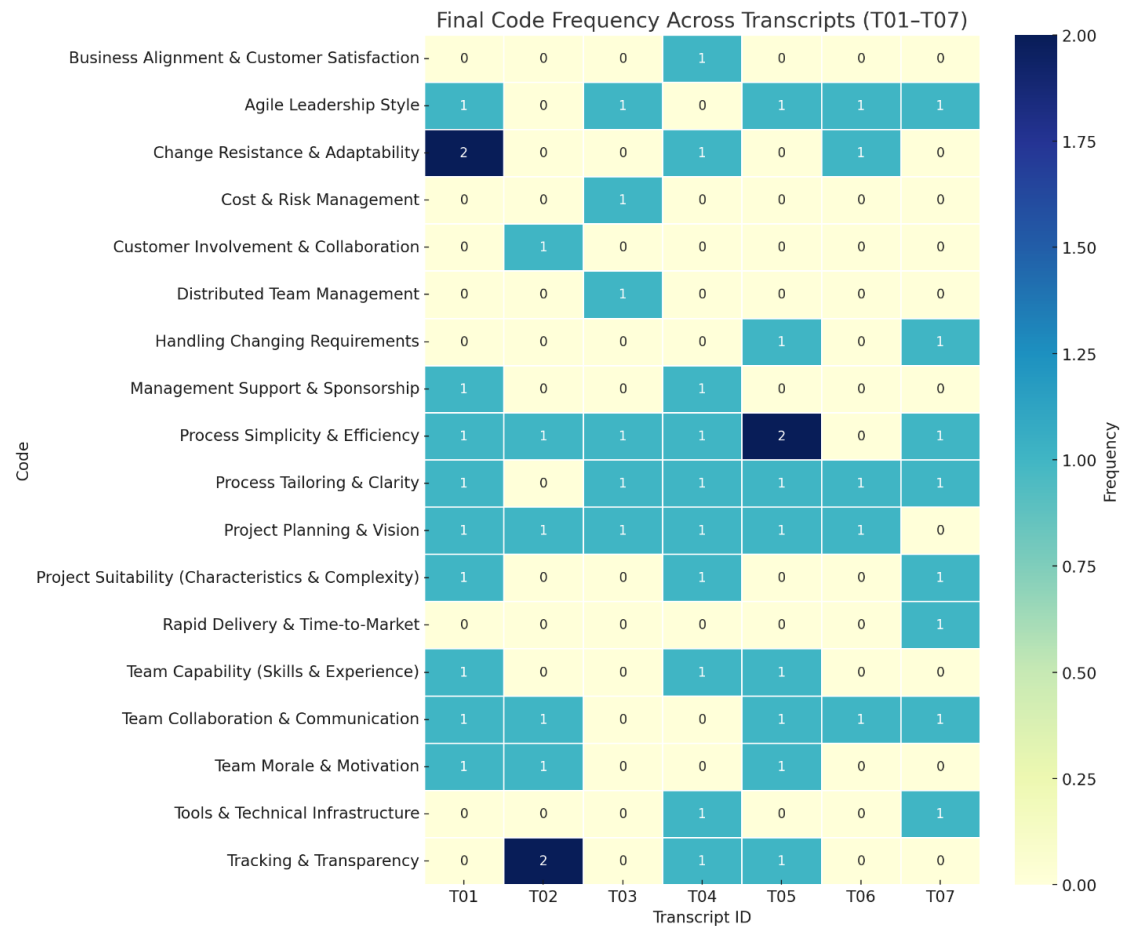


Figure 7. Code Heatmap for Interviews

4.2.2.1 Cross-Interview Insights

Throughout the interviews, process tailoring surfaced as the most frequently coded dimension, underscoring the manner in which all teams modify Agile principles according to their own environment, team size, or domain. Clarity of vision and strategic planning were essential, especially in nascent or founder-led teams (T01, T03, T04), where a firm sense of direction frequently acted as a stabilizing force amid uncertainty. Leadership styles exhibited considerable variation, encompassing compassionate servant-leadership (T01) and highly flexible, engineering-oriented leadership (T07), yet the fundamental Agile principles of responsiveness and empowerment persisted.

Communication methods were used with notable ingenuity, utilizing emoji-based standups (T07) and pizza metaphors to illustrate technical complexity (T07), alongside sketches (T01) and humour-infused retrospectives (T05), showcasing how teams rendered Agile rituals enjoyable and approachable. The technical infrastructure served as a discreet yet essential

facilitator in various circumstances, particularly for teams encountering limitations in tools or resources.

Distributed Agile has become an increasing problem, especially in managing asynchronous cooperation and the intricacies of alignment across time zones. Simultaneously, morale and motivation were intricately linked to the occurrence of retrospectives and the transparency of individual contributions (T02, T05), underscoring the emotional aspect of Agile work. Notably, funding and cost management issues, albeit significant in the context of startups, were unexpectedly underrepresented as clear themes; only T03 expressly addressed these difficulties, while financial pressures softly resonated across other narratives.

4.2.2.2 Reflexive Thoughts

The initial codebook demonstrated exceptional robustness, with no significant gaps that required the establishment of new categories. Many participants articulated their Agile adaptations using innovative metaphors or artifacts; nonetheless, these innovations were rationally integrated into existing codes such as Process Tailoring or Team Collaboration and Communication, without necessitating an expansion of the coding system. In many instances, conceptual overlaps emerged especially across dimensions such as Tailoring and Change Resistance, but these were meticulously addressed by concentrating on the participant's purpose, differentiating between intentional customization and reactionary friction. The variety of participant positions, including developers, product managers, and founders, enhanced the dataset, demonstrating the fluid distribution of Agile tasks among team members in micro-startup settings, rather than being confined to strict formal roles.

4.3 Summary of Findings

4.3.1 Thematic Analysis

The stable codebook (v2), created through coding of the interview material, was then evaluated to extract overarching theme patterns. This analytical procedure resulted in the discovery of five interconnected meta-themes. Table 6 delineates the themes, the associated codes that contributed to each theme, and a succinct summary of each theme.

Table 6 Meta Themes with Codes and Brief Synthesis

Theme ID	Meta-Theme	Member Codes	Synthesis
TH-1	Team Empowerment	Agile Leadership Style, Team Capability (Skills & Experience), Team Morale & Motivation, Customer Involvement & Collaboration	Identifies internal facilitators, including leadership, team competencies, motivation, and customer proximity, that are crucial for Agile success. Explicitly conveys ideas for cultivating agility within lean teams (RQ3).
TH-2	Agile Framework Fit & Frictions	Management Support & Sponsorship, Project Planning & Vision, Tracking & Transparency, Tools & Technical Infrastructure, Cost & Risk Management, Rapid Delivery & Time-to-Market, Handling Changing Requirements, Process Simplicity & Efficiency, Business Alignment & Customer Satisfaction	Examines structural and practical impediments that affect the appropriateness and efficacy of Agile implementation in smaller startups. Addresses RQ1 and RQ2 by emphasizing challenges in planning, tools, execution, and value alignment.
TH-3	Cultural & Change Barriers	Organisational Culture & Readiness, Change Resistance & Adaptability	Examines mentality and organizational inertia as impediments to Agile transformation. Extremely crucial for comprehending internal resistance (RQ1) and evaluating readiness (RQ3).

TH-4	Contextual Tailoring Needs	Process Tailoring & Clarity, Project Suitability (Characteristics & Complexity), Distributed Team Management	Indicates the necessity to tailor Agile methodologies to accommodate particular micro startup circumstances, including team size, dispersion, and project scope. Directly facilitates RQ2 and RQ3 regarding the customization of frameworks.
TH-5	Collaboration– Quality Linkages	Team Collaboration & Communication, Product Quality & Maintainability	Highlights the direct impact of internal collaboration and coordination on product sustainability and quality results. Facilitates the identification of optimal practices that have to be preserved or improved inside any customized framework (RQ3).

4.3.1.1 Theme 1 – Team Empowerment

Definition

This subject encompasses internal enablers such flexible leadership, team motivation, technical proficiency, and close user collaboration that promote effective Agile deployment.

Illustrative Quotation

"A scrum master would come on board and tell the product manager or the software developer to work within their domain of responsibilities..." (T01)

4.3.1.2 Theme 2 – Agile Framework Fit & Frictions

Definition

This issue examines the practical difficulties and obstacles faced while using structured Agile frameworks such as Scrum or Kanban in smaller startups.

Illustrative Quotation

“The framework is already lightweight... You need discipline before you start bending the rules.” (T02)

4.3.1.3 Theme 3 – Cultural & Change Barriers

Definition

This theme encapsulates organizational inertia, resistance to change, and entrenched practices that obstruct Agile transformation.

Illustrative Quotation

“There’s no formal pause to ask, ‘What did we learn?’” (T06)

4.3.1.4 Theme 4 – Contextual Tailoring Needs

Definition

This theme underscores the imperative to modify or integrate Agile methodologies according to startup-specific variables, like team size, distribution, and project magnitude.

Illustrative Quotation

" “We can't implement a perfect sprint each time... but in general, we do implement a lot of agile practices...” (T04)

4.3.1.5 Theme 5 – Collaboration–Quality Linkages

Definition

This theme examines the correlation between communication techniques and the capacity to generate high-quality, sustainable products.

Illustrative Quotation

"You communicate with graphs and sketches... features..." (T01)

Table 7 shows the thematic coverage across the seven interview transcripts.

Table 7 Thematic Coverage Summary

Theme	% by Participant (T01, T02, T03, T04, etc.)
-------	---

Team Empowerment	T01 (27.3%), T02 (18.2%), T04 (9.1%), T05 (27.3%), T06 (9.1%), T07 (9.1%)
Agile Framework Fit & Frictions	T01: 50.0%, T02: 16.7%, T03: 16.7%, T04: 16.7%
Cultural & Change Barriers	T04 (50.0%), T06 (50.0%)
Contextual Tailoring Needs	T01 (12.5%), T03 (25.0%), T04 (25.0%), T05 (12.5%), T06 (12.5%), T07 (12.5%)
Collaboration–Quality Linkages	T01 (12.5%), T02 (12.5%), T03 (12.5%), T04 (25.0%), T05 (12.5%), T06 (12.5%), T07 (12.5%)

4.3.2 Theme Interrelationships

The five discovered themes are interconnected, engaging in significant and dynamic interactions. Empowered teams constitute the foundation, facilitating the proactive customization of Agile frameworks to align with particular startup circumstances. Nonetheless, the unyielding application of frameworks, when unregulated, incites resistance, underscoring the necessity of adaptability. Efforts to tailor responses, frequently driven by survival, are facilitated by robust collaborative procedures that guarantee coherence and quality in the face of changing operational challenges. Culture change develops incrementally via everyday interactions rather than through orchestrated interventions.

4.3.3 Holistic Literature Triangulation

Table 8 shows a comprehensive analysis demonstrates a robust correlation between the meta themes and the chosen literature.

Table 8 Literature Triangulation

Theme	Literature Support Summary
Team Empowerment	Emphasised by all five papers (P1–P5) as critical for Agile success.

Agile Framework Fit & Frictions	Frequently discussed in relation to micro startup challenges. (P1, P2, P3, P5)
Cultural & Change Barriers	Consistently identified as a major obstacle. (P1, P3, P4, P5)
Contextual Tailoring Needs	Supported as an indicator of Agile maturity. (P1, P2, P3)
Collaboration–Quality Linkages	Strongly associated with Agile effectiveness and product quality. (P1, P4, P5)

4.3.4 Cross-Theme Insights

Thematic patterns disclosed intricate interconnections among elements affecting Agile adaption in micro startups:

- Empowered teams facilitate customization: Startups in which team members experienced autonomy and leadership trust were more adept at proactively customizing Agile methodologies. Empowerment facilitated experimental changes, enabling teams to preserve agility while addressing contextual realities.
- Rigid frameworks incite resistance: The application of Agile frameworks in a strict, formulaic manner, disregarding organizational dynamics, resulted in increased resistance and friction among participants. Rigid compliance frequently conflicted with the dynamic and evolving characteristics of startup operations, resulting in either passive or aggressive disengagement.
- Tailoring is a survival mechanism: Instead of being a deliberate enhancement, the adaptation of Agile methodologies frequently arose from a reactive imperative. Startups adapted to endure resource constraints, swift pivots, and dynamic team structures. Tailoring facilitated operational continuity without burdening teams with excessive formalities.
- Collaboration is the adhesive. In settings with limited or inconsistent formal Agile frameworks, robust interpersonal communication and mutual understanding mitigated methodological deficiencies. Effective collaboration emerged as the essential factor uniting teams in the absence of stringent processes.
- Cultural transformation is intrinsic, not contrived: The shift towards Agile within organizations was perceived as a natural, ingrained process rather than a consequence of top-down change attempts. Behavioural norms evolved progressively through everyday interactions and collective learning, demonstrating that culture develops along with Agile methods rather than before to them.

4.4 Conceptual Framework

Based on the study's thematic analysis, a conceptual framework is recommended to help micro software startups adopt Agile in a way that lasts. Throughout this framework, there is a focus on communication, teamwork, and quality integration, which shows how continuous enablers and periodic strategic actions work together in a dynamic way. There are three main parts that make up the framework, the foundation pillars, episodic actions and the unifying layer.

Foundation Pillars: Team Empowerment and Cultural & Change Barriers (TH1 & TH3)

These pillars represent the internal conditions that underpin Agile success: the presence of empowered, motivated, and technically capable teams, and the parallel need to recognise and navigate cultural resistance, entrenched habits, and adaptive challenges within the organisation. These two factors are constant enablers that must be present throughout the full journey of the company. They support daily activities, the way decisions are made, and how people interact with each other.

- **Team empowerment** helps leaders be flexible, learn new skills, get motivated, and get customers involved.
- **Culture Navigation** makes sure that hidden resistance is slowly brought to the surface and dealt with through everyday encounters and relationships instead of orders from the top down.

Both elements are always active and aren't just turned on and off occasionally. They are what hold Agile maturity in micro startups together.

Episodic Actions: Agile Calibration and Contextual Tailoring (TH2 & TH4)

Along with these principles, Agile Calibration and Contextual Tailoring happen on a regular basis based on how needs change.

- The team gets together for **Agile Calibration** to reevaluate and understand Agile models again. They figure out which practices are helpful, which are burdensome, and how well current methods fit the team's needs.
- Right after tuning comes **Contextual Tailoring**, which changes some Agile practices to fit the team's size, structure, project type, and growth stage better.

Calibration and tailoring cycles happen at different times depending on internal review points, such as when the project changes, the team grows, or there are process bottlenecks. So, these two parts work together in a way that lets Agile methods adapt to the changing needs of the company.

The Unifying Layer: Collaboration, and Quality Integration (TH5)

Encompassing all components is a persistent, overarching *requirement* and *goal*: Collaboration, and Quality Integration.

- As a **requirement**, effective collaboration and quality integration are fundamental for:
 - Maintaining team’s empowerment and cultural agility.
 - Successfully calibrating and tailoring Agile practices.
- Teams are required to do all actions for empowerment, cultural agility, agile calibration, and customization, with the **goal** of maintaining communication quality, collaborative cohesiveness, and process excellence.

In this way collaboration and quality integration are both the minimum requirement and the goal, shaping, enabling, and improving every other part. Every choice, adjustment, and attempt at customization should be made with the goal of improving communication, teamwork, and the smooth integration of quality into the product and work processes. Figure 8 visually depicts the proposed conceptual framework for micro-adaptive agile.

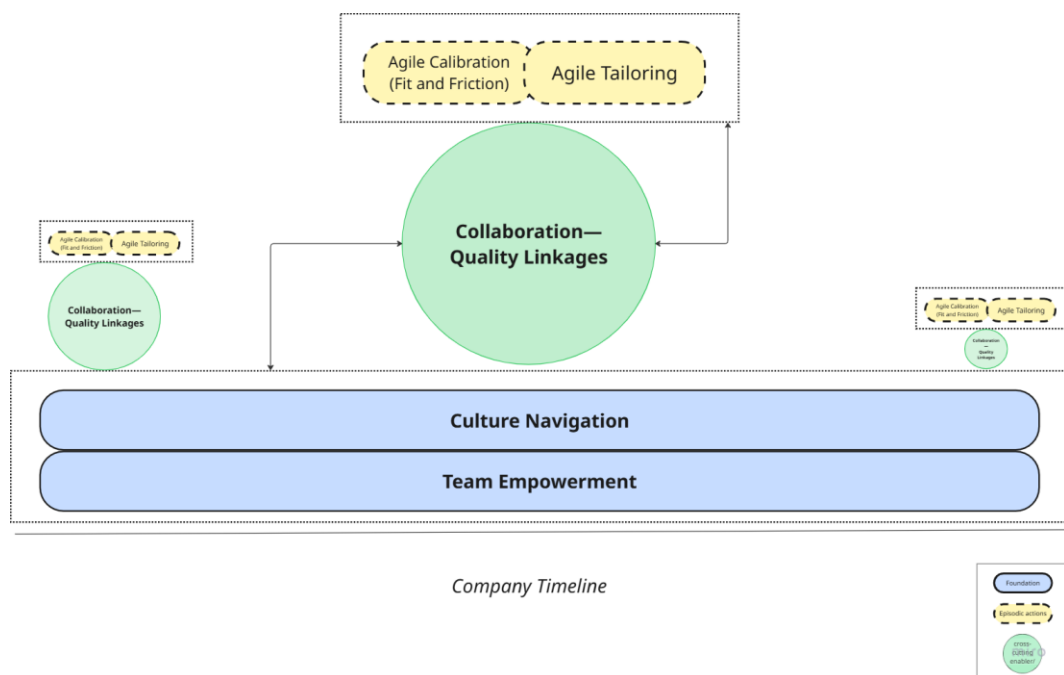


Figure 8. Conceptual Framework for Micro-Adaptive Agile

4.5 Reflexivity Statement

The researcher's expertise in software development and Agile methodologies established a robust contextual basis for data interpretation. Nonetheless, this insider position presented complications, specifically the danger of overstating participants' adherence to formal Agile principles while undervaluing localized, improvised modifications.

In the initial stages of coding, variations from normal Scrum or Kanban practices were often perceived as shortcomings rather than as strategic, resource-based adjustments. Acknowledging this bias, the researcher employed systematic memoing and iterative peer consultations to recontextualize judgments based on participants' actual experiences rather than idealized frameworks. Initially, comments characterized the lack of formal retrospectives as a failure; subsequent critical thought elucidated that teams intentionally excluded them to maintain operational efficiency in micro-scale contexts. The final analysis was to remain anchored in the real practices and meanings expressed by participants, rather than imposing normative Agile standards through this reflexive process.

4.6 Results Summary

This chapter has methodically outlined the results of a thematic analysis with hybrid coding technique about Agile adaptation in micro software startups. Utilizing a literature-based codebook on primary interview data and corroborating results with fundamental academic sources, five interrelated meta-themes were discerned: Team Empowerment, Agile Framework Fit & Frictions, Cultural & Change Barriers, Contextual Tailoring Needs, and Collaboration–Quality Linkages.

The investigation indicated that although Agile concepts possess considerable promise for improving startup agility, their successful adoption depends on adaptable practices, empowered leadership, and a conducive cultural climate. Inflexible structures lacking contextual adaptation frequently intensified opposition, while teamwork proved essential for sustaining quality within limitations. Moreover, organizational transformation was identified as an emerging, spontaneous process rather than a top-down imposition.

These findings provide a detailed comprehension of how micro startups effectively manoeuvre through the opportunities and constraints presented by Agile methodology. The subsequent

chapter examines these observations concerning overarching practical ramifications, establishing a foundation for actionable recommendations and future research avenues.

5. DISCUSSION AND IMPLICATION

5.1. Discussion

The study's results thoroughly respond to the three research inquiries. Initially, while addressing RQ1, the obstacles encountered by micro software companies in adopting Agile were elucidated through themes including Cultural and Change Barriers, as well as Agile Framework Fit and Frictions. These obstacles encompass passive resistance, stringent framework prerequisites, and an absence of formal structure, all of which impede successful Agile implementation.

The findings pertaining to RQ2 indicate that Agile Framework Fit, Frictions, and Contextual Tailoring Requirements reveal that established frameworks such as Scrum and Kanban frequently do not correspond with the actual conditions of micro startups. Participants underscored the necessity for adaptation and simplification, acknowledging the partial applicability yet overall inflexibility of traditional Agile methodologies. Ultimately, RQ3 is resolved via the proposed conceptual framework, which delineates actionable strategies such as team empowerment, cultural navigation, regular Agile calibration, and contextual tailoring

as vital for cultivating a more sustainable and adaptable Agile adoption model in micro startups. The framework's central emphasis on communication, collaboration, and quality integration underscores the necessity for Agile to be perpetually redefined in response to changing internal dynamics and external limitations.

The findings enhance current Agile theory by illustrating that micro startup environments amplify the significance of adaptability while diminishing the effectiveness of inflexible frameworks. Classical Agile principles, such as Scrum roles and ceremonies, presuppose team stability and role clarity, which are frequently lacking. This paper proposes a "micro-adaptive Agile" paradigm, viewing agility not as a static methodology but as a dynamic practice rooted in interpersonal relationships and pragmatic resource management.

5.2. Practical implication

The findings suggest that Agile methodologies should be significantly simplified for implementation in micro startups. Priority should be given to leadership empowerment, flexible tracking, and adaptive planning rather than rigid compliance with formalized procedures. Policymakers advocating for startup ecosystems could contemplate financing mentorship programs that provide founders with Agile leadership competencies, rather than solely technical instruction. Investors analysing micro startups may find it advantageous to assess agility not by methodological rigor but through indicators of adaptability.

5.3. Future studies

Future research may expand on this study by developing and empirically assessing a "micro-adaptive Agile" framework specifically designed for the requirements of micro, dynamic teams functioning in high-uncertainty environments. This approach would provide practical recommendations on scalable, efficient methods for nascent enterprises.

Furthermore, longitudinal studies monitoring Agile adaption over time from the origin of startups through various development and scaling phases could yield significant insights into the evolution of Agile practices and the factors that either promote or impede their efficacy. Comparative analysis of micro startups that effectively implement Agile methodologies versus those that encounter ongoing difficulties would be advantageous, particularly if it investigates mediating factors such as leadership styles, organizational learning capabilities, and external influences like funding structures and investor anticipations.

6. CONCLUSION

This study examined the intricate reality of Agile implementation in micro software startups, a situation frequently neglected in conventional Agile literature. Five principal themes emerged from a thematic analysis of firsthand accounts: Team Empowerment, Agile Framework Fit and Frictions, Cultural and Change Barriers, Contextual Tailoring Requirements, and Collaboration-Quality Linkages. These findings indicate that Agile in micro startups prioritizes the ongoing adaptation of established frameworks over strict adherence to them, particularly in dynamic, resource-limited contexts. Internal team dynamics, informal leadership, and relationship trust emerged as essential facilitators of Agile maturity, but cultural inertia and framework mismatch represented considerable obstacles.

The research enhances both theoretical and practical domains by presenting a conceptual framework that amalgamates fundamental enablers with situational calibration and contextual adaptation, all interconnected through communication and quality-centric collaboration. This approach elucidates the sustainable implementation of Agile in micro startups and emphasizes the necessity of perceiving Agile as a dynamic, growing practice instead of a fixed set of rules. As the startup landscape evolves, adaptable, lean, and context-sensitive Agile models will be essential for fostering resilience, innovation, and sustainable growth in micro software startups.

LIST OF REFERENCES

- Abdalhamid, S., & Mishra, A. (2017). Factors in Agile Methods Adoption. *TEM JOURNAL - Technology, Education, Management, Informatics*, 6, 416–421.
<https://doi.org/10.18421/TEM62-29>
- Altuwajri, F., & Ferrario, M. A. (2022). A Framework for the Adoption of Agile within Software SMEs in Saudi Arabia. *Proceedings of the 2021 European Symposium on Software Engineering*, 73–77. <https://doi.org/10.1145/3501774.3501785>
- Audretsch, D., Colombelli, A., Grilli, L., Minola, T., & Rasmussen, E. (2020). Innovative start-ups and policy initiatives. *Innovative Start-Ups and Policy Initiatives*, 49(10), 104027.
<https://doi.org/10.1016/j.respol.2020.104027>
- Azhar, M. U. (2016). *CHALLENGES FACED BY STARTUP COMPANIES IN SOFTWARE PROJECT MANAGEMENT* [Tampere University of Technology].
<https://trepo.tuni.fi/bitstream/handle/123456789/25352/Mazhar.pdf?sequence=4&isAllowed=y>
- Banijamali, A., Dawadi, R., Ahmad, M. O., Similä, J., Oivo, M., & Liukkunen, K. (2017). *Empirical Investigation of Scrumban in Global Software Development* (pp. 229–248).
https://doi.org/10.1007/978-3-319-66302-9_12
- Baskoro, H., Prabowo, H., Meyliana, M., & Gaol, F. (2022a). Predicting Startup Success, a Literature Review. *International Conference on Information Science and Technology Innovation (ICoSTEC)*, 1, 123–129. <https://doi.org/10.35842/icostec.v1i1.6>
- Baskoro, H., Prabowo, H., Meyliana, M., & Gaol, F. (2022b). Predicting Startup Success, a Literature Review. *International Conference on Information Science and Technology Innovation (ICoSTEC)*, 1, 123–129. <https://doi.org/10.35842/icostec.v1i1.6>
- Blank, S. (2020). *The four steps to the epiphany: Successful strategies for products that win*. John Wiley & Sons.

- Bosch, J., Olsson, H., Björk, J., & Ljungblad, J. (2013). The Early Stage Software Startup Development Model: A Framework for Operationalizing Lean Principles in Software Startups. *LESS 2013. LNBIP*, 167, 1–15. https://doi.org/10.1007/978-3-642-44930-7_1
- Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, 3, 77–101. <https://doi.org/10.1191/1478088706qp063oa>
- Carmel, E. (1994). Time-to-completion in software package startups. *1994 Proceedings of the Twenty-Seventh Hawaii International Conference on System Sciences*, 4, 498–507.
- CB Insights, C. I. (2021, August 3). The Top 12 Reasons Startups Fail. *The Top 12 Reasons Startups Fail*. <https://www.cbinsights.com/research/report/startup-failure-reasons-top/>
- Coleman, G., & O'Connor, R. V. (2008). An investigation into software development process formation in software start-ups. *Journal of Enterprise Information Management*, 21(6), 633–648. <https://doi.org/10.1108/17410390810911221>
- Collyer, S., Warren, C., Hemsley, B., & Stevens, C. (2010). Aim, Fire, Aim – Project Planning Styles in Dynamic Environments. *Project Management Journal*, 41. <https://doi.org/10.1002/pmj.20199>
- Cusumano, M. A. (2010). *Staying power: Six enduring principles for managing strategy and innovation in an uncertain world*. Oxford University Press.
- Eman, A. (2024). Impact of Agile Methodology on Software Development. *Computer and Information Science*, 8, 9–9. <https://doi.org/10.5539/cis.v8n2p9>
- Fereday, J., & Muir-Cochrane, E. (2006). Demonstrating rigor using thematic analysis: A hybrid approach of inductive and deductive coding and theme development. *International Journal of Qualitative Methods*, 5(1), 80–92. <https://doi.org/10.1177/160940690600500107>
- Fergis, K. (2012). *The Impact of an Agile Methodology on Software Development Costs*.
- Fontana, R. & Sabrina Marczak. (2020). Characteristics and Challenges of Agile Software Development Adoption in Brazilian Government. *Journal of Technology Management and Innovation*, 15, 3–10. <https://doi.org/10.4067/S0718-27242020000200003>

- Frederick, T., Lam, T., & Martin, V. (n.d.). A Lean Innovation Model To Help Organizations Leverage Innovation For Economic Value: A Proposal. *International Journal of Management & Information Systems*, 18(2).
- Freeman, J., Carroll, G. R., & Hannan, M. T. (1983). The Liability of Newness: Age Dependence in Organizational Death Rates. *American Sociological Review*, 48(5), 692–710. JSTOR. <https://doi.org/10.2307/2094928>
- Giardino, C., Bajwa, S. S., Wang, X., & Abrahamsson, P. (2015). Key Challenges in Early-Stage Software Startups. In C. Lassenius, T. Dingsøyr, & M. Paasivaara (Eds.), *Agile Processes in Software Engineering and Extreme Programming* (pp. 52–63). Springer International Publishing.
- Giardino, C., Wang, X., & Abrahamsson, P. (2014). Why Early-Stage Software Startups Fail: A Behavioral Framework. In C. Lassenius & K. Smolander (Eds.), *Software Business. Towards Continuous Value Delivery* (pp. 27–41). Springer International Publishing.
- Graham, P. (2012, September). *Startup = Growth*. <https://www.paulgraham.com/growth.html>
- Inayat, I., Salim, S. S., Marczak, S., Daneva, M., & Band, S. (2014). A systematic literature review on agile requirements engineering practices and challenges. *Computers in Human Behavior*. <https://doi.org/10.1016/j.chb.2014.10.046>
- Institute of Project Management. (2022, August 15). *Critical Path Method*. <https://projectmanagement.ie/blog/critical-path-method/>
- Javdani Gandomani, T., Zulzalil, H., & Bahsoon, R. (2025). Empowering software startups with agile methods and practices: A design science research. *Software Practice and Experience*, 55, 220–242. <https://doi.org/10.1002/spe.3371>
- Kadenic, M., Konstantinos, K., & Junker-Jensen, L. (2022). Mastering Scrum with a focus on Team Maturity and Key Components of Scrum. *Information and Software Technology*, 153, 107079. <https://doi.org/10.1016/j.infsof.2022.107079>
- Kern, C., Klute-Wenig, S., & Refflinghaus, R. (2024). Using Agile Techniques for Setting Up a Quality Management System for Startups. *RIF Institute of Research and Transfer*;

Department of Process Optimization, Corresponding Author: Christian Kern, christian.kern@rif-ev.de; Sandra Klute-Wenig, sandra.klute-wenig@rif-de; Robert Refflinghaus, robert.refflinghaus@rif-de.

- Khan, S. M. (2023). *Waterfall Model Used in Software Development Reference: Software Requirements Engineering Waterfall Model*.
<https://doi.org/10.13140/RG.2.2.29580.69764>
- Krishnan, S. N., Ganesh, L. S., & Rajendran, C. (2020). Characterizing and Distinguishing 'Innovative Start-ups' Among Micro, Small and Medium Enterprises (MSME). *Journal of New Business Ventures*, 1(1-2), 125-156. <https://doi.org/10.1177/2632962X20964418>
- Lenka, M. (2020). Startups and reasons for their failure. *Current Problems of the Corporate Sector 2020*. SHS Web of Conferences 83. <https://doi.org/10.1051/shsconf/20208301046>
- Melegati, J., & Kon, F. (2020). *Early-Stage Software Startups: Main Challenges and Possible Answers* (pp. 129-143). https://doi.org/10.1007/978-3-030-35983-6_8
- Mkpojiogu, E., Hashim, N. L., Assaqqaf, A., & Hussain, A. (2019). *Software Startups: Motivations for Agile Adoption*.
- Panzavolta, F. (2019, February 5). *Exploring the Scrum Values Workshop*.
<https://www.scrum.org/resources/blog/exploring-scrum-values-workshop>
- PMI. (2023, March). *PMI Pulse of the Profession 2023*. <https://www.pmi.org/-/media/pmi/documents/public/pdf/learning/thought-leadership/pmi-pulse-of-the-profession-2023-report.pdf>
- Prasongko, A., & Adianto, T. (2019). THE ROLE OF THE AGILE LEADERSHIP MODEL AS A COMPETITIVE ADVANTAGE FOR THE FUTURE LEADER IN THE ERA OF GLOBALIZATION AND INDUSTRIAL REVOLUTION 4.0. *Jurnal Pertahanan*, 5, 126.
<https://doi.org/10.33172/jp.v5i3.596>
- Putrianasari, R., Budiardjo, E., Mahatma, K., & Raharjo, T. (2024). Problems in The Adoption of Agile-Scrum Software Development Process in Small Organization: A Systematic Literature Review. *Sinkron*, 9, 495-504. <https://doi.org/10.33395/sinkron.v9i1.13271>

- Raharjo, T., & Purwandari, B. (2020). *Agile Project Management Challenges and Mapping Solutions: A Systematic Literature Review* (p. 129).
<https://doi.org/10.1145/3378936.3378949>
- Rahman, A., Indrajit, E., Unggul, A., & Dazki, E. (2024). Agile Project Management Impacts Software Development Team Productivity. *Sinkron*, 8, 1847–1858.
<https://doi.org/10.33395/sinkron.v8i3.13853>
- Razdan, R., & Kambalimath, S. (2019). *Super Lean Software Startup Engineering Management* (p. 6). <https://doi.org/10.1109/TEMSCON.2019.8813609>
- Ressin, M. (2022). Start-ups as drivers of economic growth. *Research in Economics*, 76(4), 345–354. <https://doi.org/10.1016/j.rie.2022.08.003>
- Ries, E. (2011). *The lean startup: How today's en-trepreneurs use continuous innovation to createradically successful businesses*.
- Royce, W. (2021). *Managing the Development of Large Software Systems (1970)* (pp. 321–332). <https://doi.org/10.7551/mitpress/12274.003.0035>
- S. M. Sutton. (2000). The role of process in software start-up. *IEEE Software*, 17(4), 33–39.
<https://doi.org/10.1109/52.854066>
- Santisteban, J., & Mauricio, D. (2017). *SYSTEMATIC LITERATURE REVIEW OF CRITICALSUCCESS FACTORS OF INFORMATION TECHNOLOGY STARTUPS*. 23(2).
- Saunders, M., Lewis, P., & Thornhill, A. (2023). *Research Methods for Business Students*.
- 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.
<https://doi.org/10.1016/j.ijproman.2015.01.006>
- Sirisha, M., Sarada, V., John, D., Ramakrishna, B., Bisen, D., & Haralayya, D. (2024). Project Management Methodologies: A Comparative Analysis of Agile and Waterfall Approaches. *Library Progress (International)*, 44, 17237–17246.
- Skala, A. (2019). *Digital Startups in Transition Economies: Challenges for Management, Entrepreneurship and Education*. <https://doi.org/10.1007/978-3-030-01500-8>

- Soares Bastos, L., Silva, F., Machado, I., & de Souza, R. M. (2019). *Investigating Agile Practices in Software Startups*. <https://doi.org/10.1145/3350768.3350786>
- Szarek, J., & Piecuch, J. (2018). The importance of startups for construction of innovative economies. In K. Wach & M. Maciejewski (Eds.), *International Entrepreneurship as the Bridge between International Economics and International Business: Conference Proceedings of the 9th ENTRE Conference – 5th AIB-CEE Chapter Annual Conference* (pp. 389–398). Cracow University of Economics.
- Unterkalmsteiner, M., Abrahamsson, P., Wang, X., Nguyen Duc, A., Shah, S., Sohaib, S., Baltes, G., Conboy, K., Cullina, E., Dennehy, D., Edison, H., Fernández, C., Garbajosa, J., Gorschek, T., Klotins, E., Hokkanen, L., Kon, F., Lunesu, M. I., Marchesi, M., & Yagüe, A. (2016). Software Startups—A Research Agenda. *E-Informatica Software Engineering Journal*, 2016, 89–124. <https://doi.org/10.5277/e-Inf160105>
- Venczel, T. B., Berényi, L., & Hriczó, K. (2024). The Project and Risk Management Challenges of Start-ups. *Acta Polytechnica Hungarica*, 21(2).
- Verwijns, C., & Russo, D. (2023). A Theory of Scrum Team Effectiveness. *ACM Trans. Softw. Eng. Methodol.*, 32(3). <https://doi.org/10.1145/3571849>
- Wasserman, N. (2008). The Founder's Dilemma. *Harvard Business Review*, 86, 102–109, 138.
- Wooder, S., & Baker, S. (2012). Extracting Key Lessons in Service Innovation. *Journal of Product Innovation Management*, 29, 13–20. <https://doi.org/10.1111/j.1540-5885.2011.00875.x>
- Zielske, M., Held, T., & Kourouklis, A. (2022). A Framework on the Use of Agile Methods in Logistics Startups. *Logistics*, 6(1). <https://doi.org/10.3390/logistics6010019>

APPENDICES

Appendix A1. Codebook v1

Code Label	Papers	Definition	Typical Inclusion	Typical Exclusion	Example Quote (page)
Lit_Organisational Culture & Readiness	P1 P3	Extent to which a startup's culture and conventions facilitate or obstruct Agile adoption.	traditional culture that is inflexible and resistant to change.	Only technical issues; just a personal attitude.	"...factors in organisational dimension that can be the reason of failure such as the culture of an organisation being too traditional..." (P1 p.420)
Lit_Management Support & Sponsorship	P1 P3	Support and endorsement from founders or senior management for Agile methodologies.	resources for executive sponsorship and change.	pure enthusiasm for the team.	"Top management commitment was another critical factor ..." (P3 p.503)
Lit_Agile Leadership Style	P1 P4	Leadership defined by adaptability, swift decision-making, and team empowerment.	Rapid, daring, and empowering leaders.	command-and-control leadership.	"The agile leadership model ... fast, responsive ... risk-taker ... very adaptive ..." (P4 p.130)
Lit_Team Capability (Skills & Experience)	P1 P5	Proficiency and expertise of the team in development and Agile methodologies.	adequate technical and nimble skills.	Skills in customer service; minor personal traits.	"Lack of the necessary skill set. Lack of project-management competence." (P1 p.420)

Lit_Team Collaboration & Communication	P1 P3 P4	teamwork and internal communication quality.	regular communication and cross-functional cooperation.	Customer service (separate code).	"Lack of team work." (P1 p.420)
Lit_Team Morale & Motivation	P2	The team's confidence and enthusiasm during the implementation of Agile.	high vitality as a result of empowerment or early successes.	general order at work.	"Improved team morale ..." (P2 p.458)
Lit_Customer Involvement & Collaboration	P1 P3	interaction between the team and clients/end users.	Participation in reviews and regular feedback.	only collaboration within the company.	"Keeping good relationship with customer ... if there is any problem ... leads to failure." (P1 p.420)
Lit_Change Resistance & Adaptability	P1 P3 P4 P5	degree of adaptability or resistance to change.	Fear of new processes; flexibility.	formal rulings regarding tailoring.	"Insufficient support from management and resistance to change ..." (P3 p.503)
Lit_Process Tailoring & Clarity	P1 P2 P3	establishing a clear workflow and adapting Agile techniques to the situation.	Clearly defined roles and adaptation to Scrum/Kanban.	"Bookish" rigidity.	"Absence of the right Agile practices ..." (P1 p.420)
Lit_Project Planning & Vision	P1 P5	Adaptive planning and a clear product vision.	distinct goals and a dynamic backlog.	tracking daily tasks.	"Quality of the vision/goals is a marginally significant moderator ..." (P5 p.1046)
Lit_Tracking & Transparency	P1 P2	mechanisms for predictable and observable progress.	Task boards, open status, and burndowns.	micromanagement and pointless KPIs.	"Not having tracking mechanisms for Agile progress." (P1 p.420)

Lit_Tools & Technical Infrastructure	P1 P3	infrastructure and software technologies that facilitate Agile.	CI/CD, version control, and collaboration tools.	raw coding ability; choice of methodology.	"Tools and technology are inappropriate." (P1 p.420)
Lit_Project Suitability (Characteristics & Complexity)	P1 P2 P5	Agile and project qualities fit together in the setting of microstartups.	small groups and evolving needs.	routine work with a fixed scope.	"Agile methods are most appropriate to projects where requirements are not well characterised ..." (P1 p.418)
Lit_Cost & Risk Management	P2	Agile is used to control costs and risks.	decreased iterative risk mitigation and rework.	Schedule only; pure quality concerns.	"Motivations ... include ... reduced project risk, reduced project cost ..." (P2 p.458)
Lit_Rapid Delivery & Time-to-Market	P1 P2	pressure to use Agile to develop things fast.	Regular releases and early delivery.	Speed without delivery; stand-up duration.	"Need for accelerated product delivery ..." (P2 p.457)
Lit_Handling Changing Requirements	P1 P2 P5	the capacity to adjust to changing priorities and circumstances.	Rearranging the backlog and embracing change.	unchecked scope creep.	"Need for an enhanced ability to manage changing priorities." (P2 p.457)
Lit_Product Quality & Maintainability	P1 P2	ensuring a codebase that is both high quality and manageable.	automated refactoring and testing.	speed at the price of quality.	"Enhance software quality ... increase software maintainability ..." (P2 p.460)
Lit_Process Simplicity & Efficiency	P1 P2 P5	drive to increase productivity and streamline procedures.	elimination of waste and lightweight structures.	Enterprise bureaucracy is heavy.	"Need for a simplified development process ..." (P2 p.457)

Lit_Business Alignment & Customer Satisfaction	P1 P2 P5	enhancing customer satisfaction and coordinating development with corporate objectives.	frequent releases determined by the company's worth.	Separate internal morale; ambiguous marketing objectives.	"... Agile methodologies can possibly give a greater amount of consumer satisfaction ..." (P1 p.417)
Lit_Distributed Team Management	P2	coordinating Agile work among micro-startup teams that are not located in the same place.	remote customs; async collaboration tools.	Teams in the same room; big enterprise projects that are outsourced.	"Better managed distributed teams ..." (P2 p.458)

Appendix A2. Coverage Matrix

Code Label	P1	P2	P3	P4	P5
Lit_Organisational Culture & Readiness	✓		✓		
Lit_Management Support & Sponsorship	✓		✓		
Lit_Agile Leadership Style	✓			✓	
Lit_Team Capability (Skills & Experience)	✓				✓
Lit_Team Collaboration & Communication	✓		✓	✓	
Lit_Team Morale & Motivation		✓			
Lit_Customer Involvement & Collaboration	✓		✓		
Lit_Change Resistance & Adaptability	✓		✓	✓	✓
Lit_Process Tailoring & Clarity	✓	✓	✓		

Lit_Project Planning & Vision	✓				✓
Lit_Tracking & Transparency	✓	✓			
Lit_Tools & Technical Infrastructure	✓		✓		
Lit_Project Suitability (Characteristics & Complexity)	✓	✓			✓
Lit_Cost & Risk Management		✓			
Lit_Rapid Delivery & Time-to-Market	✓	✓			
Lit_Handling Changing Requirements	✓	✓			✓
Lit_Product Quality & Maintainability	✓	✓			
Lit_Process Simplicity & Efficiency	✓	✓			✓
Lit_Business Alignment & Customer Satisfaction	✓	✓			✓
Lit_Distributed Team Management		✓			

Appendix B1. Interview Questions

For Founders / Product Managers/Team Member

General Background

1. Can you briefly describe your role within the startup and how your team is structured?
2. How long has your startup been operating, and how many team members do you currently have?
3. Is your startup currently funded? If not, how do you manage resource limitations?
4. What is the general skill level and professional experience of your team members?

Agile Awareness and Adoption

5. Are you familiar with Agile methodologies?
6. Have you implemented any Agile practices in your startup? If yes, which ones (e.g., sprints, Kanban boards, retrospectives)?
7. If not, what alternative approach do you use to manage your projects?
8. What motivated you (or discouraged you) from adopting Agile practices?

Agile Fit & Challenges in Micro Startups

9. What specific challenges have you encountered when trying to implement Agile in your startup (e.g., time, people, tools)?
10. Do you find Agile helpful or overwhelming for micro teams? Why?
11. How do you manage communication, alignment, and coordination within your team?
12. In your opinion, are standard Agile frameworks well suited for micro teams with limited resources?

Team Dynamics and Leadership

13. How are responsibilities distributed among your team members?
14. How do you handle Agile ceremonies (like standups or retrospectives) with a small team?
15. How would you describe your leadership style, and how does it influence your team's workflow?

Perceived Value and Adaptation of Agile

17. What do you find most valuable about Agile principles or practices?
18. Are there any Agile practices that you find difficult or impractical to apply?
19. Have you customized Agile practices to suit your team's needs? If yes, how?
20. If not, what barriers prevent you from tailoring Agile methods?

21. Would you consider adopting a tailored Agile framework designed for small startups? Why or why not?
22. How do you balance the need to move quickly with the discipline required for structured Agile implementation?

For Investors

General Background

1. Can you share your experience investing in startups?
2. What types of startups do you typically invest in (e.g., stage, industry, team size)?
3. How familiar are you with Agile management principles?
4. Have you observed Agile practices in startups you've backed? If so, how successful were those implementations?

Observed Challenges in Micro Startups

5. In your view, what are the main operational challenges faced by very small or early-stage startups?
6. How do resource limitations in unfunded startups affect their ability to implement structured practices like Agile?
7. Have you noticed recurring mistakes when such startups try to adopt Agile?

Leadership and Team Assessment

8. How does the leadership style of startup founders impact their team's ability to adopt Agile principles?
9. Do you think team size and skill level influence Agile's effectiveness? Can you explain?
10. What role do you think communication and collaboration play in the success of early-stage teams?

Expectations and Strategic Insights

11. Do you believe an Agile culture is important for startup success? Why or why not?
12. When evaluating startups, do you consider their management approach (e.g., Agile) in your decision-making?
13. How important are team dynamics and process maturity in your investment criteria?
14. Would you support a customized Agile framework designed specifically for resource-constrained startups?
15. What tools, strategies, or frameworks have you seen work well in early-stage or lean environments?

Appendix B2. Participants Demographics

TranscriptID	Role	YearsExperience	StartupSize
T01	Founder	2	5 employees
T02	Founder	3	6 employees
T03	Investor	20	<i>Not reported</i>
T04	Product Manager	3	10 employees
T05	Product Manager	5	8 employees
T06	Team Member	2	7 employees
T07	Team Member	1	4 employees

Appendix C1. Stable Codebook (v2)

Code Label	Definition
Lit_Organisational Culture & Readiness	Degree to which a startup's culture and norms support or hinder Agile adoption.
Lit_Management Support & Sponsorship	Commitment and backing from founders or top management for Agile.

Lit_Agile Leadership Style	Leadership characterised by adaptability, rapid decisions and team empowerment.
Lit_Team Capability (Skills & Experience)	Skill level and competence of the team in development and Agile practices.
Lit_Team Collaboration & Communication	Quality of teamwork and internal communication.
Lit_Team Morale & Motivation	Enthusiasm and confidence of the team during Agile adoption.
Lit_Customer Involvement & Collaboration	Engagement of customers/end-users with the team.
Lit_Change Resistance & Adaptability	Degree of resistance or openness to change.
Lit_Process Tailoring & Clarity	Customising Agile methods to context and defining clear workflow.
Lit_Project Planning & Vision	Clarity of product vision and adaptive planning.
Lit_Tracking & Transparency	Mechanisms for visible progress and predictability.
Lit_Tools & Technical Infrastructure	Software tools and infrastructure that support Agile.
Lit_Project Suitability (Characteristics & Complexity)	Fit between project attributes and Agile in micro-startup context.
Lit_Cost & Risk Management	Use of Agile to manage risk and cost.
Lit_Rapid Delivery & Time-to-Market	Pressure to deliver products quickly via Agile.
Lit_Handling Changing Requirements	Ability to cope with evolving requirements/priorities.
Lit_Product Quality & Maintainability	Ensuring high quality and maintainable codebase.
Lit_Process Simplicity & Efficiency	Drive to simplify process and boost productivity.
Lit_Business Alignment & Customer Satisfaction	Aligning development with business goals and improving customer satisfaction.
Lit_Distributed Team Management	Coordinating Agile work across non-co-located micro-startup teams.