International Journal of Computing and Digital Systems
ISSN (2210-142X)
Int. J. Com. Dig. Sys. 16, No.1 (Jul-24)

http://dx.doi.org/10.12785/ijcds/160130

Navigating the Software Symphony: A Review of Factors and Strategies for Software Development in Startups

J Anitha Gracy[1], S Parthasarathy[2] and S Sivagurunathan[3*]

[1,2]Applied Mathematics and Computational Science,Thiagarajar College of Engineering, Madurai, India
[3]Department of Computer Science and Applications, The Gandhigram Rural Institute (DTBU), Dindigul, India

Received 12 Feb. 2024, Revised 8 Apr. 2024, Accepted 20 Apr. 2024, Published 1 Jul. 2024

Abstract: In the ever-evolving software industry, startups are catalysts of innovation, propelling transformative changes. Effective software development is pivotal for their success. The aim of the review is to reveal the successive factors, challenges, and strategies underlying software development in startups. We conducted a systematic literature review by developing a classification schema, ranking the selected studies, and analyzing the factors, challenges, and strategies to enhance software development for startups. Only a few studies are dedicated to software development factors; from the primary studies on software startups, successive factors, strategies, and challenges are extracted, categorized, and analyzed. The purpose of the review is to reveal the process of software project management and underline the key factors like Agile, Lean, customer centricity, and design thinking concepts in the software industry, information technology maturity models, and corporate culture. In order for startups to succeed in the shifting world of software development, it is advised that they find ways to overcome difficulties, find success factors, investigate opportunities, and be adaptable to new ideas. This study provides evidence that the ability of a software startup to embrace innovation, adaptability, and technology is vital for long-term growth and sustainability. This comprehensive approach empowers startups to reach excellence in the energetic software industry landscape.

Keywords: Software startups, Software sustainability,Software project management,Agile lean principles, Startup factors.

1. Introduction

The growth of software startups seems remarkable due to the rise in new markets, available technologies, and project capital. In today's rapidly evolving business landscape, startup organizations in the software industry stand at the forefront of innovation, driving transformative changes and shaping industries[1]. The unique dynamics of startups, characterized by their agility, adaptability, and rapid product development cycles, make effective software project management a critical determinant of success[2].As these startups continue to redefine traditional business models, optimizing software project performance becomes a strategic imperative for achieving competitive advantage and sustainable growth[3].This review article explores the factors of software project performance within developed software companies and software startup organizations, aiming to unveil the underlying factors, challenges, and innovative strategies that underpin their achievements. The objective of this research is to identify the successive factors that enhance the process of software development in well-established software companies while also examining the challenges that impact the development of software in startup organizations using an optimized approach. It concludes that attributes such as novelty, flexibility, organizational culture, and technological order are crucial for the advancement of software projects in startups. The motivation and contribution of this study lie in the dynamic environment where technology is rapidly flourishing while market demands are constantly shifting. In order to achieve success, startups must have knowledge regarding strategies, optimized factors, challenges, supportive scenarios, and unsupportive scenarios for efficient and innovative software development. The main objective of this review is to unveil ideas, such as factors for success, strategies, and challenges, for the benefit of software startup investors, researchers, and academics. Many studies are available to look at how well-known businesses use software development practices and methods. Startups often make software in an environment with limited resources and a move away from rigid, prescriptive processes. This distinction is what sets startups apart from established businesses in the realm of software engineering. Surprisingly, there is a clear study gap when it comes to how software is developed in startup ecosystems. especially when it comes to software development. This gap gets bigger when think about the unique problems that startups face when they try to make software. It's important to note that very few studies have tried to define, explain, or map the different software project methods and techniques

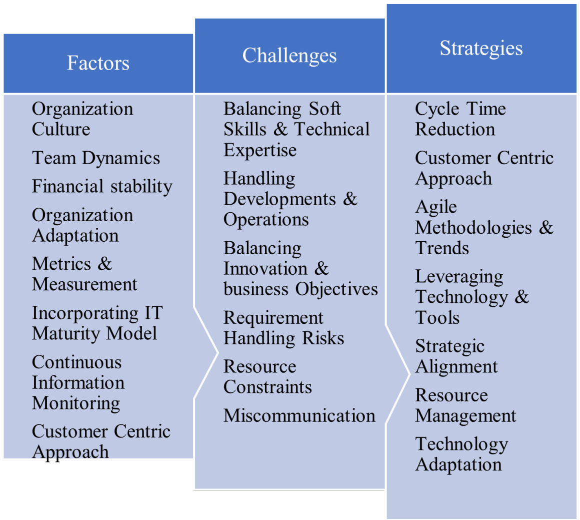
Factors
Organization Culture
Team Dynamics
Financial stability
Organization Adaptation
Metrics & Measurement
Incorporating IT Maturity Model
Continuous Information Monitoring
Customer Centric Approach
Challenges
Balancing Soft Skills & Technical Expertise
Handling Developments & Operations
Balancing Innovation & business Objectives
Requirement Handling Risks
Resource Constraints
Miscommunication
Strategies
Cycle Time Reduction
Customer Centric Approach
Agile Methodologies & Trends
Leveraging Technology & Tools
Strategic Alignment
Resource Management
Technology Adaptation

Figure 1. Identified Factors, Challenges, Strategies of Software Companies from the Literature Review

used by software startups.

So, it's important to do a thorough study to fill in this gap and give useful information about startups and how to tackle the risks in the software development life cycle. The figure 1 represents detailed study of this paper, which includes the factors that refining a supportive organizational culture, fostering effective team dynamics, maintaining financial stability, and embracing adaptability are crucial elements in enhancing the performance of software projects in startups. As well, continuous monitoring and improvement are ensured by incorporating metrics and IT maturity models. Overcoming challenges such as balancing soft skills and technical expertise, managing development and operations collaboration, aligning innovation with business objectives, and addressing resource constraints necessitates the implementation of strategic solutions. These solutions involve reducing cycle time, adopting a customer-centric approach, leveraging agile methodologies and technology trends, aligning projects with strategic goals, and optimizing resource management. The ability to navigate these factors, metrics, challenges, and strategies is essential for delivering efficient and customer-focused software projects in the dynamic landscape of startup environments.

This paper is put together in the following way: In Section 1, the introduction to established companies and startups is explained. In Section 2, study design, which includes planning, conducting, reviewing, and a summary of the review, is discussed. In Section 3, Results and Discussion, In Section 4, the study is wrapped up, and possible future directions are suggested.

2. Study Design

In the previous section, we discussed the common perspectives of software startups and established companies. This section includes planning, conducting, and reviewing phases that explore the systematic literature review, which serves as a method for discovering, assessing, and understanding all existing research pertinent to a specific research question [4],[5].Startups play a crucial role in driving innovation and economic growth, particularly in the technology sector[6].However, the high failure rate among startups underscores the importance of understanding the factors that contribute to success in software development. This literature review seeks to provide an overview of recent research on success factors and strategies for software development in startup environments, offering insights and recommendations for startup founders, investors, and other stakeholders. The existing systematic review examines recent trends in software development practices adopted by startups[6].The previous studies identified emerging technologies, methodologies, and tools that are shaping the landscape of software development in startup ecosystems [6],[7].Key findings highlight the increasing adoption of agile methodologies, DevOps practices, and cloud-based technologies among successful startups [7].This empirical study investigates the factors that contribute to software development success in early-stage startups. Through surveys and interviews with startup founders and developers, the research identifies critical success factors such as team expertise, product-market fit, and customer feedback mechanisms. The earlier studies also explore the challenges faced by startups in scaling their software development processes and offer practical recommendations for overcoming these obstacles[8].Drawing on the principles of lean startup methodology, this research examines strategies for managing software development projects in resource-constrained startup environments[9].The preceding study explores how startups can prioritize features, iterate quickly, and validate product assumptions through rapid experimentation[10],[11]. By adopting lean principles, startups can minimize waste, maximize learning, and accelerate the delivery of value to customers[12],[13].

Even though the previous research explains the various methods of software setup, there's a need to evaluate the permanent footprint of identified factors and strategies on startup success, considering their sustainability and long-term growth. This paper addressing these gaps will not only enrich academic discussions but also provide valuable insights for startup stakeholders navigating the intricacies of today's entrepreneurial landscape. The review was carried out according to a protocol that was created in advance. The following figure 2 shows the overall stages of this review.

A. *Planning*

A systematic literature review serves as a method for discovering, assessing, and comprehending all existing research pertinent to a specific research question, topic area, or phenomenon of interest [4],[5].IEEE Xplore, Scopus, and Web of Science were the three databases from which the information was gathered. Out of a total of 1195 papers, this investigation found 54 primary studies that were pertinent to the research questions at hand. The flow of this literature

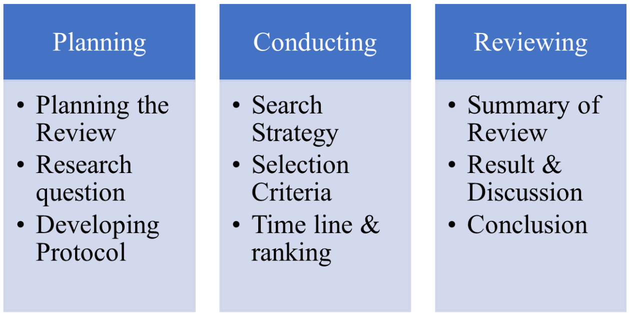

Figure 2. Review Phases

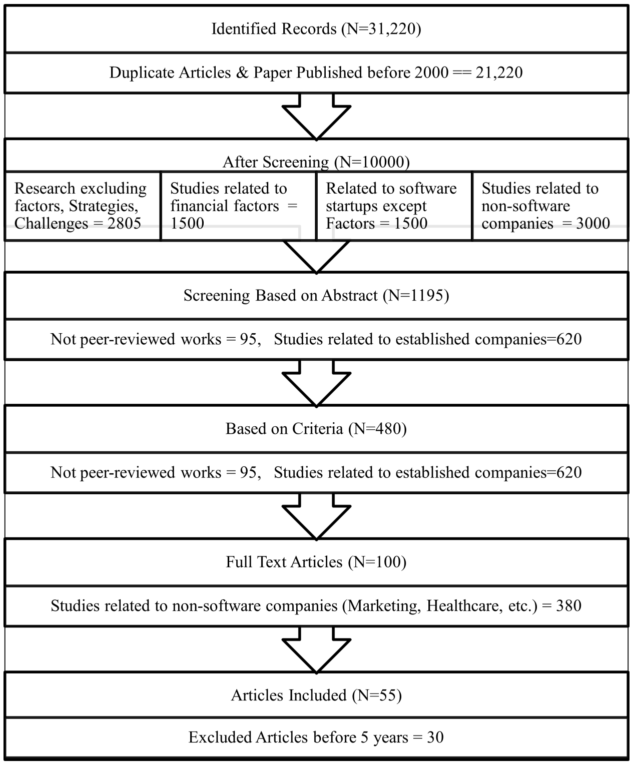
Identified Records (N=31,220)
Duplicate Articles & Paper Published before 2000 == 21,220
After Screening (N=10000)
Research excluding factors, Strategies, Challenges = 2805
Studies related to financial factors = 1500
Related to software startups except Factors = 1500
Studies related to non-software companies = 3000
Screening Based on Abstract (N=1195)
Not peer-reviewed works = 95, Studies related to established companies=620
Based on Criteria (N=480)
Not peer-reviewed works = 95, Studies related to established companies=620
Full Text Articles (N=100)
Studies related to non-software companies (Marketing, Healthcare, etc.) = 380
Articles Included (N=55)
Excluded Articles before 5 years = 30

Figure 3. Flow of Literature Review

review is presented in the following figure 3.The screening process involved terms like avoiding those unrelated to the objective of the paper, lack of full text availability, and not related to the software company. There are a number of stages to take during the review that are part of the planning process. The first step of planning is shaping the demands for the review, followed by the formulation of the research questions. Following the research questions, a development protocol is created by outlining fundamental systematic review procedures[14],[7].

1) *Research Question*

The formulation of the research questions is designed to address and bridge the existing research gap for boosting software development at software startups. We can gain insight into the sequential factors and strategies employed by established IT companies in order to foster the growth of new startups. Even though IT companies and startups share common ground, there are some inherent differences. IT companies and startup companies demonstrate noticeable distinctions in terms of their size, structure, culture, and business approach[14].Well-established IT firms are distinguished by their significant scale, encompassing clear-cut procedures and a substantial workforce with diverse expertise. Possessing an extensive client base and financial stability, IT companies are capable of engaging in long-term planning, research, and development endeavors, as well as investments in cutting-edge technologies. On the other hand, startup companies are renowned for their agility, innovation, and entrepreneurial spirit[7].Typically operating on a smaller scale, startups adopt a flat organizational structure that facilitates prompt decision-making. By prioritizing flexibility, adaptability, and speed in their development processes, startups aim to rapidly introduce inventive solutions to the market. The culture within startup companies fosters close collaboration, instilling a sense of ownership among team members and a strong customer-centric focus. While IT companies thrive on stability and established practices, startups thrive on innovation and the ability to promptly respond to market changes, each embodying a distinct set of characteristics that define their identities within the business landscape. Based on this distinction, the research question was framed, and factors were analyzed. Many startup companies emulate the practices and strategies of IT companies. Accordingly, research question 1 examines the factors that have occurred in IT companies, while question 2 investigates the relevant factors that are applicable and necessary for software startups. Lastly, question 3 focuses on the necessary strategies required for the development of software startups. This study seeks to provide answers to the below-stated questions to explore insights in the startup area. The research question will be answered in the Result and Discussion Section.The following table I formulates the research questions and its goal:

2) *Developing Review Protocol:*

It is a thorough strategy for locating relevant research materials to address the stated research questions. To reduce potential researcher biases, a predetermined review methodology is required. The technique is based on the review procedure outlined in the SLR performance guidelines, which are adapted from[8].The review protocol defines the procedures to be undertaken for the execution of a specific systematic review[8],[15].

· The SLR's historical context.

· The review's research questions should be developed.

· Setting up the search idea, search standings, and database to be used.

· Information on the inclusion and exclusion criteria for candidates and the selection processes that will be used.

TABLE I. Research Questions and Goals

RQ’s	Research Question	Goal
RQ 1	What are all the factors that enhance the software development in IT Companies?	Pinpoint the factors that inspire the software development process in IT Companies.
RQ 2	What are all the factors used to enhancing the software development in startup companies?	Identifying the factors that enhance the software development in startup companies.
RQ 3	What factors within Agile methodologies, Lean principles, and Design Thinking contribute to the success of startup software projects?	Finding the insights of Agile methods for software startup companies.

· Establishing a knowledge mining method to retrieve the necessary data from each primary study to answer the research questions.

· Summary.

B. Conducting

The main goal of the search strategy is to identify a comprehensive range of studies pertaining to the research inquiries. To accomplish this goal, we develop and employ a search strategy across diverse electronic databases. An initial pilot search is conducted to gauge the volume of articles and subsequently refine our search strings. Table IIlists the sample search keywords and database pilot search. The presented table is designed according to Kitchenham’s and Webster’s guidelines[16],[17].

1) Selection Criteria

The selection process involves the following criteria, such as inclusion and exclusion, that are tailored to align with the research question.The identification of irrelevant papers that match the exclusion criteria is done by evaluating the metadata of each paper.Table III shows the criteria of included and excluded papers [18],[13],[19].

2) Time line of Published Articles

The Figure 4 displays the year of the selected paper, breakdown of credentials on software development methodologies, and accomplishments in software startups from 2000 to 2023. Which includes a number of journals, conferences, and books. The selection of appropriate journals will play an essential role in ensuring the quality and credibility of the information produced to answer the research questions as part of this literature review. Figure 3 represents the selected articles used in this systematic literature review.

TABLE II. Result from pilot search

Search Keywords	Database	Result (Population)	Result (After Preprocessing)
(Reason for Software project success + Factors of Software Project + Critical Success Factors + Software Startups* OR startup Companies + software start-up + Boosting Software startups +Critical failure Factor + Software Performance in startups + IT startup + lean startup) + (Design Thinking in software startups OR development model + software engineering) + (Agile technologies in software startups* + method* + process*+Software Performance strategy + Organization Approach)+ (Factors of Software Projects + in Software startups))	IEEE Xplore	427	196
	Scopus	356	161
	Web of Science	412	123
	Total	**1195**	**480**

3) Articles Ranking:

The following figure 5 represent the articles that were taken for review and journal rankings. The strategic implementation of article ranking in a review paper functions as a tool for organization, prioritization, and communication of the profound significance embedded within the existing body of literature.

C. Reviewing

Startups in the software industry are known for their dynamic and fast-paced nature, presenting unique challenges and opportunities for achieving successful project outcomes. This literature review delves into the various innovative approaches and key factors that influence software project performance in startup organizations. By examining these influential factors, we gain insights into how startups can

TABLE III. Inclusion and Exclusion Criteria

Inclusion Criteria	Exclusion Criteria
The study should be published from 2015 to 2023.	Duplicate articles.
One or more of this study's research questions are directly answered by the study	Articles published in low-ranking journals.
The study should have a clear focus on the title.	Papers written before 2000.
The study should be accessible in full-text.	Studies are not clearly focused on factors in traditional companies.
The study should fill the research gap over the software startup success factors.	Studies are not clearly focused on factors in software startup companies.
A form of study taken from an experience report, applied engineering techniques, development models, or lessons learned	Related to software startups but not from a software project perspective.
	Studies related to established companies
	Studies related to non-software companies (Marketing, Healthcare, etc.)
	Not peer-reviewed works

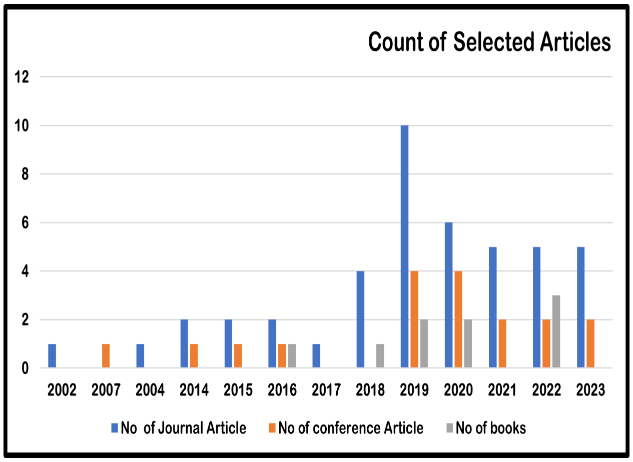

Figure 4. Representation of selected articles

enhance their project management strategies and overall success.

1) Innovation in Software Project Management

Innovation is at the heart of effective software project management in startups. IT maturity models offer a structured pathway for startups to navigate their development phases [20]. Effective software project management for startups is predicated on innovation, which enables these businesses to flourish in an environment that is always changing [21],[22]. Startups can achieve the delicate balance between creativity and discipline by harmonizing innovation with these models, thereby positioning themselves for sustained growth and success in the ever-changing world of technological entrepreneurship [23],[24]. The integration of Agile methodologies, Lean principles, and Design Thinking introduces adaptability, customer-centricity, and early validation, thereby boosting project performance [25]. These approaches empower startups to respond swiftly to market changes and ensure project success [26],[27],[6].

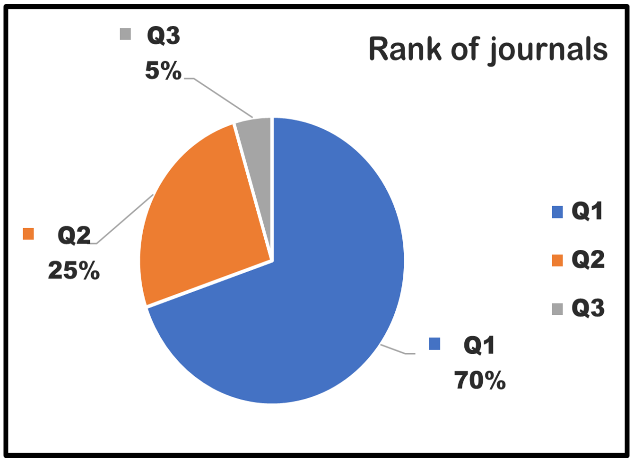

Figure 5. Rank of selected journals

2) Organizational Culture and Team Dynamics

The culture and dynamics within startup organizations have a profound impact on software project performance. Successful startups foster an entrepreneurial spirit [28] while addressing challenges such as managing technical debt and transitioning to growth phases [10],[29],[9]. The atmosphere and dynamics of the firms that are just getting started when it comes to software development are quite important [30],[31]. Startups have the ability to build an atmosphere that is conducive to innovation and the flourishing of software projects if they cultivate an entrepreneurial spirit, address any technical debt, and efficiently navigate the growth phases of their business [32],[33],[34]. Agile practices, including prototyping and Lean methodologies, support startups in navigating these challenges by promoting iterative development and early customer involvement [35],[36].

3) External Features and Adaptation

The extraordinary tasks brought about by the COVID-19 pandemic have tested the resilience of startup software projects [37],[38],[39]. Startup organizations have displayed adaptability by embracing remote work and innovative practices [40]. Agile methodologies and the ability to pivot quickly have proven essential in addressing disruptions and maintaining project momentum [41]. By embracing remote workers and agile processes, startup companies have demonstrated their adaptability to the changing business landscape [42],[12]. These strategies make it possible for startups to efficiently respond to disturbances and keep project momentum going in an environment that is always updating [43],[44]. They embrace agile approaches for iterative development, and they maintain the flexibility to make strategic course corrections [45],[46].

4) *Metrics and Measurement*

Tailored metrics are crucial for startups to gauge project performance aligned with their goals. Customized metrics frameworks offer startups the ability to make informed decisions and monitor progress effectively [47],[48]. Innovations in measurement techniques, such as machine learning and data analytics, have emerged as valuable tools for enhancing project performance and ensuring continuous improvement [47],[48],[11].

The success of software projects in startup organizations hinges on the interplay of various factors. Innovative software project management methodologies, coupled with an entrepreneurial culture, enable startups to overcome challenges and thrive in dynamic environments [49],[50]. As startups continue to shape the software landscape, a keen focus on innovative approaches and responsive strategies remains paramount for boosting software project performance and achieving sustained growth [47],[48],[11].

5) *Incorporating IT Maturity Models*

IT maturity models provide startups with a structured framework for navigating their developmental phases, ensuring that projects are aligned with organizational goals and industry best practices [11],[51]. Integrating these models fosters a proactive approach to project management and enables startups to assess their technological capabilities, thus contributing to enhanced project performance [51],[52],[53]. IT maturity models offer entrepreneurs a methodical and all-encompassing strategy for managing their technological endeavors [54],[55],[56]. The incorporation of these models into a startup's project management processes enables the company to conduct an audit of its existing technology capabilities, ensure that its projects are in line with its overall objectives, and continually enhance its overall effectiveness [57],[58],[59]. This preventative strategy makes a substantial contribution, not only to improved project performance but also to the overall success of the company in a business environment marked by intense competition.

6) *Continuous Information Monitoring*

In the realm of software startups, continuous information monitoring empowers project teams to stay updated with the latest market trends, user preferences, and technological advancements [60],[31],[61]. By leveraging real-time information, startups can make informed decisions, adapt quickly to changing conditions, and refine project strategies to ensure optimal performance. For software companies just getting started, constant information monitoring is a strategic must-have [62],[61]. It gives businesses the ability to continue being nimble, customer-oriented, and competitive despite the quickly shifting environment [61].

7) *Cycle-Time Reduction Metrics Framework*

The adoption of a metrics framework focused on cycle-time reduction is crucial for startups aiming to streamline their software value creation processes [63],[64],[65]. The path that startups take to optimize their software development processes is greatly aided by the adoption of a metrics framework with a primary emphasis on the reduction of cycle time [66],[67]. Discovering inefficiencies, effectively allocating resources, and accelerating project delivery are all things that may be accomplished for startups that collect and analyze relevant data [67],[68]. Not only can this approach, which is data-driven, improve project performance, but it also places startups in a position to succeed in markets that are dynamic and competitive.

8) *Agile Methodologies and Trends*

Agile methodologies offer startups a dynamic approach to project management, emphasizing iterative development, customer collaboration, and adaptability [60],[62],[59]. Embracing Agile practices allows startups to respond promptly to evolving requirements and market dynamics, resulting in improved project performance and a higher likelihood of meeting customer expectations [55],[57],[59]. Agile procedures give companies the ability to react quickly, maintain a focus on the client, increase the quality and efficiency of the project, keep transparency intact, and cultivate a culture of continuous improvement [50],[32]. In today's business scene, this, in turn, results in increased project performance and a considerably higher possibility of not just meeting but exceeding client expectations.

9) *Strategic Project Management Maturity*

Adopting a project management maturity model tailored to startups provides a roadmap for optimizing project practices and processes [68]. A strategic decision that recognizes the characteristics of startup environments is to adopt a project management maturity model that is adapted specifically for startups [69]. It gives new businesses a road map that is flexible, resource-efficient, and changeable and that can be used to optimize project procedures and processes. Startups can improve their project management skills and raise the possibility that they will achieve their goals by adhering to this customized road map in an environment that is both dynamic and competitive in the business world [70],[71].

10)*Balancing Soft Skills and Technical Expertise*

The success of software projects in startups relies on a harmonious blend of soft skills, such as effective communication and collaboration, with technical expertise [72],[65],[73],[67],[63]. The combination of Lean Startup and Design Thinking principles is a game-changing strategy that helps firms create a customer-centric attitude at a fundamental level [69],[71]. It enables startups to confirm their ideas, collect input from users, and iterate quickly, which ultimately results in software solutions that are finely tuned to the requirements and preferences of the client [71],[74]. This customer focused strategy is an important factor in software project success, and it enables companies to produce products that resonate with customers and succeed in a competitive marketplace [71],[75].The following table IV represents the summary of factors,strategies,methodologies, and supporting tools.

3. Result and Discussion

A. *RQ1: What are all the factors that enhance software development in IT companies?*

In a traditional company, there are several ways to increase software project performance, which are simply presented in the figure 6. Make sure everyone understands the project first. The organization must analyze the project's requirements to avoid misconceptions and scope adjustments[27]. Use requirements workshops or user interviews to acquire all the needed data. Finally, construct a project plan including tasks, dates, resources, and dependencies [28],[29] and we need a technical and domain-knowledge project team. Team members need good communication and teamwork abilities. Select a capable project manager to lead the team, manage risks, track progress, and keep the project on schedule and within budget. Implement quality assurance procedures to guarantee software satisfies quality standards. It involves code reviews, testing, and quality metrics. Openly communicate with team members and stakeholders. Regularly inform stakeholders of project progress, concerns, and adjustments. Changes should be assessed for their influence on the project timeline and budget [28][29][30]. Early project risk identification and mitigation. Track and manage risks throughout the project.

The traditional companies analyze the project's needs and, based on that, they choose a model such as Agile, Waterfall, or a hybrid strategy. The traditional IT companies maintain document project plans, design documents, coding standards, and user manuals [31][32][33]. The companies effectively allocate and manage employees, hardware, and software to assure availability.

Take feedback from end-users and stakeholders throughout the development process to ensure the product satisfies their needs. Provide the development tools, infrastructure, and technology stack to achieve project goals. To find and fix bugs early in development, utilize unit, integration, and user acceptability testing. Monitor system performance and fix bottlenecks immediately.

By considering and implementing these factors, software startup companies can enhance their software project performance and increase their chances of delivering successful products to the market. Startups can leverage tailored metrics frameworks to gauge project performance and identify areas for improvement [25, 26, 17]. By harnessing data analytics and machine learning techniques, startups gain the ability to make data-driven decisions, optimize processes, and ensure continuous enhancement of project performance.

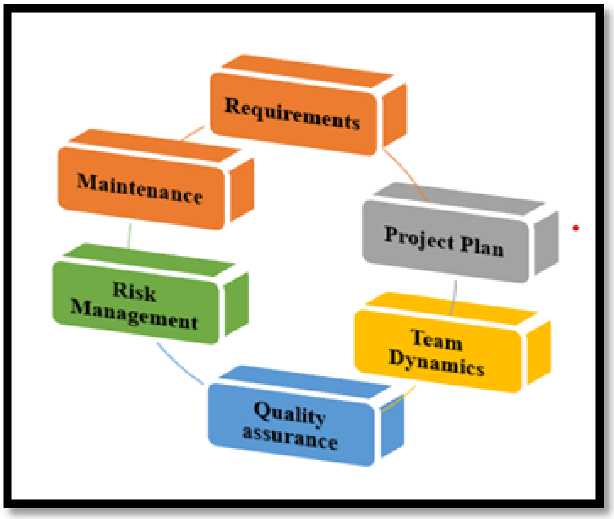

Figure 6. Factors in IT Companies

B. *RQ2: What are all the factors used to enhance the software development in software Startups?*

Startups must balance creativity, resource limits, risk management, and agility when managing software projects. Innovation is key to success, but companies must overcome obstacles to deploy and scale innovative project management methods [35][36][37].

The following table presents a summary of RQ2 from this literature review. Each startup's situation may differ, but these key factors can contribute to better software project performance, table V shows the factors that are needed to enhance the software performance in startup companies.

Startup software projects thrive when guided by innovative approaches that prioritize alignment, adaptability, and customer value. The integration of IT maturity models fosters strategic alignment and growth. Agile methodologies empower startups to navigate the ever-evolving landscape with flexibility. The emphasis on resilience and customer-centricity, highlighted through the lens of Lean Startup and Design Thinking principles, reinforces the importance of user engagement. Data-driven decision-making through metrics frameworks ensures continuous improvement. As startups balance technical expertise with interpersonal skills, they chart a course toward optimized project performance and long-term success.

By considering and implementing these factors, software startup companies can enhance their software project performance and increase their chances of delivering successful products to the market.

TABLE IV. Summary of the Literature Review

Reference	Factors	Identified Strategies	Methodologies	Supporting Tools	Description
[27],[6],[29],[9]	Innovation in Software Project Management	Incorporating IT Maturity Models	Integrated Development Environment, DevOps	Visual Studio Code, IntelliJ IDEA	These factors and methods are first steps towards quality.
[30],[31],[13],[19] [18]	Automation, Organizational Culture	Organization Culture Lean Methodologies	Performance Monitoring & Profiling, DevOps	Chrome Dev Tools	Automation increase the deliveries with short cycle time.
[5],[3],[33],[34]	Strategic Project Management	Tailoring the Project Management	Continuous Integration/Continuous Deployment	Jenkins, GitLab (CI/CD)	Improves the communication which enhances the efficiency of Software development.
[33],[34],[38],[29]	External Features & Adaptation	Strategy Testing Iteration	Project Management & Collaboration, DevOps	Jira, Trello	Promotes considerable improvement in software customization.
[35],[36],[21] [25],[22]	Metrics & Measurement	Tailored Metrics Framework	AI & Machine Learning	Tensor Flow, PyTorch	Increases software customization.
[23],[15],[24],[42] [5],[21],[42]	Incorporating IT Maturity Model	IT Maturity Models	Communication & Collaboration tools, DevOps	Slack, MS Teams	IT Maturity model is foundation for the highest level of software startup success.
[44],[24],[45],[22] [18]	Continuous Information Monitoring	Continuous Information Monitoring	Project Quality & Testing, Continuous Integration/Continuous Deployment	PyTest, Junit, RSpec	Capturing the information endures the software development improvements.
[47],[48],[11],[49] [50][51]	Cycle time Reduction Metrics Framework	Resource Allocation Accelerating Project delivery	Integrated Development Environment	Visual Studio Code, IntelliJ IDEA	Increases project arrival that enrich software development.
[36],[37],[38]	Agile Methodologies & Trends	Agile Methodologies Requirement Handling Skills	Performance Monitoring & Profiling, DevOps	Chrome Dev Tools	Fasten the development of software startups.
[48],[23], [36],[38]	Balancing Soft Skill & Technology	Innovative Ideas High in Soft Skills Technology Adaptation	Continuous Integration/Continuous Deployment, DevOps	Visual Studio Code, IntelliJ IDEA	Continuous delivery enables delivering High-quality software development.
[39],[54],[40],[41]	Customer Centric Approach	Continuous Delivery Lean Requirement Handling Skill	AI & Machine Learning Test Automation	Chrome Dev Tools	Testing automation helps ensure reliability that maintains software development.

TABLE V. Factors that needed to enhance software development in startups

Reference	Factors	Identified Strategies
[2],[21],[48],[9],[30] [55],[66]	Innovation in Software Project Management	A clear software project vision and strategy are crucial. Startups should set goals, objectives, and a roadmap for how the project fits into their business plan.
[24],[16],[25],[26],[28],[10] [9],[32],[37]	Agile Development	Agile methods like Scrum and Kanban can help firms adapt to changing requirements, improve communication, and boost project flexibility.
[3],[4],[38],[39],[40],[23] [24],[16],[9]	External Factors & Adaptation	Time, budget, and staff must be managed carefully. Set priorities and deploy resources to boost productivity.
[30],[32],[38],[47],[58],[68]	Organizational Culture and Team Dynamics	Team, stakeholder, and consumer communication must be open and transparent. Clear communication prevents misconceptions and aligns colleagues with project goals.
[40],[11],[42],[50],[55]	Continuous Information Monitoring, Testing Quality	A thorough testing and quality assurance approach early in the development cycle reduces costly errors.
[30],[31],[32],[33],[68],[69]	Strategic Project Maturity	Assess risks and devise mitigation strategies. Prevention of project delays and budget overruns requires aggressive risk management.
[20],[42],[12],[43],[44],[70]	Use of Tools and Technology	Leveraging the right tools and technology for project management, version control, and collaboration can streamline workflows and improve project efficiency.
[19],[35],[44],[53],[57],[59]	Monitoring and Metrics	Implement monitoring systems to track project progress, performance, and key metrics. This data can help in making informed decisions and adjustments as needed.
[22],[23],[24],[70]	Flexibility and Adaptability	Be prepared to adapt to changing market conditions, customer preferences, and emerging technologies. Startups should be agile and ready to pivot when necessary.
[25],[6],[10],[71]	Balancing Soft Skill Acquisition	Encourage a culture of continuous learning and improvement within the organization. Stay up-to-date with industry best practices and encourage professional development among team members.
[7],[41],[49],[72],[73]	User-Centric Design	Focus on user-centric design principles to create software that addresses the needs and preferences of the target audience.
[21],[23],[30],[40],[11],[74],[13]	Legal and Compliance Considerations	Ensure that the software complies with relevant permissible and governing requirements, like data privacy laws and industry standards.

TABLE VI. Strategies and Kernels

Strategies	Kernels
Agile methodologies	Early validation and adaptability to changing requirements. [4],[51],[52]
Lean principles	The build-measure-learn feedback loop to quickly validate hypotheses and assumptions. [71],[72]
Design Thinking	Empathy for Users, start by gaining a deep understanding of users' needs, pain points, and behaviors through user research and interviews. [53],[73]
Customer-Centricity	All three approaches emphasize a deep understanding of customer needs and continuous customer involvement. [8],[55],[56]
Iterative Approach	Critical Success Factors include regular retrospectives, demo sessions, and the ability to pivot based on feedback. [57],[58],[59],[31]
Cross-functional Teams	Assemble teams with diverse skills and perspectives to ensure that development, design, and business considerations are integrated from the outset. [53],[54],[61],[63],[70]
Continuous Improvement Culture	Encourage a culture of continuous improvement, in which teams regularly look at their processes and look for ways to improve them and produce more value in less time. [4],[49],[41],[11],[65]
Flexibility and Adaptability	Quickly reprioritize and adjust the project direction. [37],[38],[39],[69],[67],[42]
Measuring and validating	Regularly review metrics and adjust. [41],[51],[58],[70],[71]

C. *RQ3: What strategies within Agile methodologies, Lean principles, and Design Thinking contribute to the success of startup software projects?*

Creating an environment that is adaptable, customer-centric, and early validated in startup software projects requires a combination of agile methodologies, lean principles, and design thinking [39]. Each of these approaches brings its own critical success factors in this review, In agile methodologies, iterative and incremental development break the project into small, manageable increments (sprints in Scrum) that are developed and validated iteratively. This allows for early validation and adaptability to changing requirements [39],[40]. The following table VI shows the strategies with their kernels. In the realm of software startups, the key strategies serve as integral kernels that shape the core principles guiding these ventures. Adopting agile methodologies ensures early validation and adaptability to changing requirements, enabling startups to respond swiftly to evolving customer needs. Lean principles, particularly the Build-Measure-Learn feedback loop, facilitate rapid validation of hypotheses, minimizing resource waste and promoting efficiency. Design thinking, with its emphasis on empathy for users, ensures startups deeply understand user needs through research, contributing to enhanced user experiences. Customer-centricity, a common thread among these approaches, underscores the importance of continuous customer involvement and a profound understanding of their needs. An iterative approach, supported by regular retrospectives and the ability to pivot based on feedback, ensures ongoing alignment with objectives. Cross-functional teams integrate diverse skills from the outset, fostering collaboration and holistic product development. Encouraging a continuous improvement culture ensures startups are adaptable and strive for optimization. Flexibility and adaptability allow quick reprioritization and adjustments, vital in responding to dynamic market changes. Regularly reviewing metrics and adjusting based on validated data enables startups to make informed decisions and continually enhance their offerings. Together, these strategies form a robust foundation for software startups, fostering innovation, customer satisfaction, and resilience in the face of uncertainties.

By combining these factors from Agile methodologies, Lean principles, and Design Thinking, startup software projects can create an environment that is adaptable, customer-centric, and early validated, increasing their chances of success in delivering valuable products that enhance software development, which are all considered strategies for software startup success.

4. Conclusion

This review delves into software project performance in startup organizations, uncovering pivotal factors, challenges, and innovative strategies. Factors, exemplified by Agile, Lean, and Design Thinking, rewrite adaptability and customer-centricity, driving success. Organizational culture and Agile practices harmonize, while tailored metrics and data analytics provide guiding insights. Startup resilience, evident during disruptions like the pandemic, emphasizes the need for adaptive practices. IT maturity models and continuous monitoring contribute structured overtures, enhancing efficiency. Challenges, including uncertainties and resource constraints, punctuate the symphony. Skill development and balanced leadership address this discordance.

To attain sustained growth, startups must embrace innovation, align projects with business goals, nurture resilience, and leverage metrics. User-centric design thinking and DevOps practices enrich the composition, while balancing innovation and quality assurance refines the harmony.

Through this multifaceted approach, startups can achieve excellence, resonating within the dynamic landscape of the software industry. Collectively, the literature review elucidates the multidimensional nature of software project performance in startups. It emphasizes the symbiotic relationship between innovative approaches, adaptability, and technological leverage. By addressing challenges, optimizing strategies, and embracing evolving technologies, startups can navigate the complex terrain of software project management with agility and achieve sustainable growth. Additionally, frameworks for balancing innovation with quality assurance and continuous improvement methodologies like DevOps could enhance efficiency and product reliability in startup software development processes. Through these endeavors, researchers can contribute to fostering excellence and sustainability within the ever-evolving landscape of the software industry.

5. Acknowledgement

The authors express their gratitude to the Thiagarajar College of Engineering (TCE) for supporting us to carry out this research work. Also, the financial support from TCE under Thiagarajar Research Fellowship scheme (File.no: TRF/Jan-2023/08) is gratefully acknowledged.

References

[1] T. Dyba, T. Dingsoyr, and G. K. Hanssen, "Applying systematic reviews to diverse study types: An experience report," in *First international symposium on empirical software engineering and measurement (ESEM 2007)*. IEEE, 2007, pp. 225–234.

[2] B. Kitchenham, "Procedures for performing systematic reviews," *Keele, UK, Keele University*, vol. 33, no. 2004, pp. 1–26, 2004.

[3] J. Webster and R. T. Watson, "Analyzing the past to prepare for the future: Writing a literature review," *MIS quarterly*, pp. xiii–xxiii, 2002.

[4] K. K. Kemell, A. Nguyen-Duc, M. Suoranta, and P. Abrahamsson, "Startcards—a method for early-stage software startups," *Information and Software Technology*, vol. 160, p. 107224, 2023.

[5] P. Ciancarini, M. Farina, O. Okonicha, M. Smirnova, and G. Succi, "Software as storytelling: A systematic literature review," *Computer Science Review*, vol. 47, p. 100517, 2023.

[6] C. Spreafico and D. Russo, "Eco-assessment software: A quantitative review involving papers and patents," *Computer Science Review*, vol. 40, p. 100401, 2021.

[7] M. Morandini, T. A. Coleti, E. Oliveira Jr, and P. L. P. Corrêa, "Considerations about the efficiency and sufficiency of the utilization of the scrum methodology: A survey for analyzing results for development teams," *Computer Science Review*, vol. 39, p. 100314, 2021.

[8] J. Saad, S. Martinelli, L. S. Machado, C. R. de Souza, A. Alvaro, and L. Zaina, "Ux work in software startups: A thematic analysis of the literature," *Information and Software Technology*, vol. 140, p. 106688, 2021.

[9] A. Khosravi, M. Rajabzadeh, V. Zaloga, and I. Dyadyura, "Customer knowledge management in enterprise software development companies: organizational, human and technological perspective," *Management Systems in Production Engineering*, 2022.

[10] M. Zorzetti, I. Signoretti, L. Salerno, S. Marczak, and R. Bastos, "Improving agile software development using user-centered design and lean startup," *Information and Software Technology*, vol. 141, p. 106718, 2022.

[11] C. Matthies, R. Heinrich, and R. Wohlrab, "Investigating software engineering artifacts in devops through the lens of boundary objects," in *Proceedings of the 27th International Conference on Evaluation and Assessment in Software Engineering*, June 2023, pp. 12–21.

[12] L. Zaina, J. Choma, J. Saad, L. Barroca, H. Sharp, L. Machado, and C. R. de Souza, "What do software startups need from ux work?" *Empirical Software Engineering*, vol. 28, no. 3, p. 80, 2023.

[13] D. Ciric Lalic, B. Lalic, M. Delic, D. Gracanin, and D. Stefanovic, "How project management approach impact project success? from traditional to agile," *International Journal of Managing Projects in Business*, vol. 15, no. 3, pp. 494–521, 2022.

[14] A. Mishra and Z. Otaiwi, "Devops and software quality: A systematic mapping," *Computer Science Review*, vol. 38, p. 100308, 2020.

[15] E. W. Tegegne, P. Seppänen, and M. O. Ahmad, "Software development methodologies and practices in start-ups," *IET Software*, vol. 13, no. 6, pp. 497–509, 2019.

[16] V. Berg, J. Birkeland, A. Nguyen-Duc, I. O. Pappas, and L. Jaccheri, "Software startup engineering: A systematic mapping study," *Journal of Systems and Software*, vol. 144, pp. 255–274, 2018.

[17] C. Giardino, M. Unterkalmsteiner, N. Paternoster, T. Gorschek, and P. Abrahamsson, "What do we know about software development in startups?" *IEEE Software*, vol. 31, no. 5, pp. 28–32, 2014.

[18] R. Pereira and J. Serrano, "A review of methods used on it maturity models development: A systematic literature review and a critical analysis," *Journal of Information Technology*, vol. 35, no. 2, pp. 161–178, 2020.

[19] P. Savolainen, J. J. Ahonen, and I. Richardson, "Software development project success and failure from the supplier's perspective: A systematic literature review," *International Journal of Project Management*, vol. 30, no. 4, pp. 458–469, 2012.

[20] H. Edison, X. Wang, R. Jabangwe, and P. Abrahamsson, "Innovation initiatives in large software companies: a systematic mapping study," in *Proceedings of the 2018 International Conference on Software and System Process*, May 2018, pp. 140–141.

[21] M. A. Akbar, K. Smolander, S. Mahmood, and A. Alsanad, "Toward successful devsecops in software development organizations: A decision-making framework," *Information and Software Technology*, vol. 147, p. 106894, 2022.

[22] Y. Lindsjørn, D. I. Sjøberg, T. Dingsøyr, G. R. Bergersen, and T. Dybå, "Teamwork quality and project success in software development: A survey of agile development teams," *Journal of Systems and Software*, vol. 122, pp. 274–286, 2016.

[23] N. Paternoster, C. Giardino, M. Unterkalmsteiner, T. Gorschek, and

P. Abrahamsson, "Software development in startup companies: A systematic mapping study," *Information and Software Technology*, vol. 56, no. 10, pp. 1200–1218, 2014.

[24] J. Vom Brocke and A. Maedche, "The dsr grid: six core dimensions for effectively planning and communicating design science research projects," *Electronic Markets*, vol. 29, pp. 379–385, 2019.

[25] B. D. Sethanandha, B. Massey, and W. Jones, "Managing open source contributions for software project sustainability," in *PICMET 2010 Technology Management For Global Economic Growth*. IEEE, 2010, pp. 1–9.

[26] C. Becker, S. Betz, R. Chitchyan, L. Duboc, S. M. Easterbrook, B. Penzenstadler, and C. C. Venters, "Requirements: The key to sustainability," *IEEE Software*, vol. 33, no. 1, pp. 56–65, 2015.

[27] N. Condori-Fernandez and P. Lago, "Characterizing the contribution of quality requirements to software sustainability," *Journal of Systems and Software*, vol. 137, pp. 289–305, 2018.

[28] M. A. Akbar, J. Sang, A. A. Khan, S. Mahmood, S. F. Qadri, H. Hu, and H. Xiang, "Success factors influencing requirements change management process in global software development," *Journal of Computer Languages*, vol. 51, pp. 112–130, 2019.

[29] C. Giardino, N. Paternoster, M. Unterkalmsteiner, T. Gorschek, and P. Abrahamsson, "Software development in startup companies: the greenfield startup model," *IEEE Transactions on Software Engineering*, vol. 42, no. 6, pp. 585–604, 2015.

[30] R. Parizi, M. Prestes, S. Marczak, and T. Conte, "How has design thinking being used and integrated into software development activities? a systematic mapping," *Journal of Systems and Software*, vol. 187, p. 111217, 2022.

[31] A. Nguyen-Duc, K. K. Kemell, and P. Abrahamsson, "The entrepreneurial logic of startup software development: A study of 40 software startups," *Empirical Software Engineering*, vol. 26, pp. 1–55, 2021.

[32] O. Cico, R. Souza, L. Jaccheri, A. Nguyen Duc, and I. Machado, "Startups transitioning from early to growth phase-a pilot study of technical debt perception," in *Software Business: 11th International Conference, ICSOB 2020, Karlskrona, Sweden, November 16–18, 2020, Proceedings*, vol. 11. Springer International Publishing, 2021, pp. 102–117.

[33] V. Berg, J. Birkeland, A. Nguyen-Duc, I. O. Pappas, and L. Jaccheri, "Achieving agility and quality in product development-an empirical study of hardware startups," *Journal of Systems and Software*, vol. 167, p. 110599, 2020.

[34] P. Ralph, S. Baltes, G. Adisaputri, R. Torkar, V. Kovalenko, M. Kalinowski, and R. Alkadhi, "Pandemic programming: How covid-19 affects software developers and how their organizations can help," *Empirical Software Engineering*, vol. 25, pp. 4927–4961, 2020.

[35] M. Niazi, S. Mahmood, M. Alshayeb, A. M. Qureshi, K. Faisal, and N. Cerpa, "Toward successful project management in global software development," *International Journal of Project Management*, vol. 34, no. 8, pp. 1553–1567, 2016.

[36] Z. Wan, X. Xia, D. Lo, and G. C. Murphy, "How does machine learning change software development practices?" *IEEE Transactions on Software Engineering*, vol. 47, no. 9, pp. 1857–1871, 2019.

[37] S. Al-Saqqa, S. Sawalha, and H. AbdelNabi, "Agile software development: Methodologies and trends," *International Journal of Interactive Mobile Technologies*, vol. 14, no. 11, 2020.

[38] H. Kerzner, *Using the Project Management Maturity Model: Strategic Planning for Project Management*. John Wiley & Sons, 2019.

[39] G. Matturro, F. Raschetti, and C. Fontán, "A systematic mapping study on soft skills in software engineering," *Journal of Universal Computer Science*, vol. 25, no. 1, pp. 16–41, 2019.

[40] N. Tripathi, M. Oivo, K. Liukkunen, and J. Markkula, "Startup ecosystem effect on minimum viable product development in software startups," *Information and Software Technology*, vol. 114, pp. 77–91, 2019.

[41] P. Rodríguez, M. Mäntylä, M. Oivo, L. E. Lwakatare, P. Seppänen, and P. Kuvaja, "Advances in using agile and lean processes for software development," in *Advances in Computers*. Elsevier, 2019, vol. 113, pp. 135–224.

[42] F. P. Zasa, A. Patrucco, and E. Pellizzoni, "Managing the hybrid organization: How can agile and traditional project management coexist?" *Research-Technology Management*, vol. 64, no. 1, pp. 54–63, 2020.

[43] E. Klotins, M. Unterkalmsteiner, P. Chatzipetrou, T. Gorschek, R. Prikladnicki, N. Tripathi, and L. B. Pompermaier, "A progression model of software engineering goals, challenges, and practices in start-ups," *IEEE Transactions on Software Engineering*, vol. 47, no. 3, pp. 498–521, 2019.

[44] V. Gupta, J. M. Fernandez-Crehuet, T. Hanne, and R. Telesko, "Requirements engineering in software startups: A systematic mapping study," *Applied Sciences*, vol. 10, no. 17, p. 6125, 2020.

[45] P. Tyrväinen, M. Saarikallio, T. Aho, T. Lehtonen, and R. Paukeri, "Metrics framework for cycle-time reduction in software value creation," in *International Conference on Software Engineering Advances (IARIA)*, 2015.

[46] M. Daneva and R. Bolscher, "What we know about software architecture styles in continuous delivery and devops?" in *Software Technologies: 14th International Conference, ICSOFT 2019, Prague, Czech Republic, July 26–28, 2019, Revised Selected Papers 14*. Springer International Publishing, 2020, pp. 26–39.

[47] J. Melegati, R. Chanin, X. Wang, A. Sales, and R. Prikladnicki, "Enablers and inhibitors of experimentation in early-stage software startups," in *International Conference on Product-Focused Software Process Improvement*. Springer International Publishing, November 2019, pp. 554–569.

[48] C. Giardino, S. S. Bajwa, X. Wang, and P. Abrahamsson, "Key challenges in early-stage software startups," in *Agile Processes in Software Engineering and Extreme Programming: 16th International Conference, XP 2015*. Springer International Publishing, 2015, pp. 52–63.

[49] K. Gama, G. Valença, P. Alessio, R. Formiga, A. Neves, and N. Lacerda, "The developers' design thinking toolbox in hackathons: a study on the recurring design methods in software development marathons," *International Journal of Human–Computer Interaction*, vol. 39, no. 12, pp. 2269–2291, 2023.

[50] K. K. Kemell, A. Nguyen-Duc, X. Wang, J. Risku, and P. Abrahamsson, "Software startup essence: How should software startups

work?" in *Fundamentals of Software Startups: Essential Engineering and Business Aspects*. Springer International Publishing, 2020, pp. 97–109.

[51] R. W. Macarthy and J. M. Bass, "An empirical taxonomy of devops in practice," in *2020 46th euromicro conference on software engineering and advanced applications (seaa)*. IEEE, August 2020, pp. 221–228.

[52] N. Rompho, "Operational performance measures for startups," *Measuring Business Excellence*, vol. 22, no. 1, pp. 31–41, 2018.

[53] M. Guerriero, M. Garriga, D. A. Tamburri, and F. Palomba, "Adoption, support, and challenges of infrastructure-as-code: Insights from industry," in *2019 IEEE International Conference on Software Maintenance and Evolution (ICSME)*. IEEE, September 2019, pp. 580–589.

[54] J. Saad, P. P. Costa, A. Alvaro, and L. A. Zaina, "Developers experience (dx) in alm software tools: An investigation on virtual kanban boards," in *Proceedings of the XXXIV Brazilian Symposium on Software Engineering*, October 2020, pp. 167–172.

[55] A. Y. Gheni, Y. Y. Jusoh, M. A. Jabbar, N. Mohd Ali, M. Shanmugam, and H. A. Yousif, "Measuring the performance of the virtual teams in global software development projects," *Journal of Information Technology Management*, vol. 11, no. 1, pp. 42–59, 2019.

[56] C. Jones, *Software Engineering Best Practices: Lessons from Successful Projects in the Top Companies*. McGraw-Hill Education, 2010.

[57] B. Fitzgerald and K. J. Stol, "Continuous software engineering: A roadmap and agenda," *Journal of Systems and Software*, vol. 123, pp. 176–189, 2017.

[58] A. Arpteg, B. Brinne, L. Crnkovic-Friis, and J. Bosch, "Software engineering challenges of deep learning," in *2018 44th euromicro conference on software engineering and advanced applications (SEAA)*. IEEE, August 2018, pp. 50–59.

[59] R. G. Cooper, "The drivers of success in new-product development," *Industrial Marketing Management*, vol. 76, pp. 36–47, 2019.

[60] U. Ojiako, Y. Petro, A. Marshall, and T. Williams, "The impact of project portfolio management practices on the relationship between organizational ambidexterity and project performance success," *Production Planning & Control*, vol. 34, no. 3, pp. 260–274, 2023.

[61] P. C. Abrantes and A. P. Furtado, "Agile development practices applied to software startups: A systematic mapping review," in *2021 16th Iberian Conference on Information Systems and Technologies (CISTI)*. IEEE, June 2021, pp. 1–6.

[62] A. Nguyen-Duc, S. M. A. Shah, and P. Ambrahamsson, "Towards an early stage software startups evolution model," in *2016 42th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, August 2016, pp. 120–127.

[63] G. Kamulegeya, R. Mugwanya, and R. Hebig, "Requirements for measurement dashboards and their benefits: A study of start-ups in an emerging ecosystem," in *2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. IEEE, August 2019, pp. 300–308.

[64] M. Z. M. H, M. N. Mahdi, M. S. M. Azmi, L. K. Cheng, A. Yusof, and A. R. Ahmad, "Software project management using machine learning technique - a review," in *2020 8th International Conference on Information Technology and Multimedia (ICIMU)*, November 2020, pp. 363–370.

[65] P. A. Laplante and M. Kassab, *Requirements engineering for software and systems*. Auerbach Publications, 2022.

[66] U. Rafiq and X. Wang, "Continuous information monitoring in software startups," in *International Conference on Agile Software Development*. Springer International Publishing, June 2020, pp. 280–287.

[67] R. A. Haraty and G. Hu, *Software process models*, 2018.

[68] S. S. Bajwa, "Pivoting in software startups," in *Fundamentals of Software Startups: Essential Engineering and Business Aspects*, 2020, pp. 27–43.

[69] J. Hamunen, *Challenges in adopting a DevOps approach to software development and operations*, 2016.

[70] D. Lock, *Project management*. Routledge, 2020.

[71] M. Hammad, A. Alqaddoumi, and H. Al-Obaidy, "Predicting software faults based on k-nearest neighbors classification," *International Journal of Computing and Digital Systems*, vol. 8, no. 5, pp. 462–467, 2019.

[72] M. Bańka, M. Salwin, D. Masłowski, S. Rychlik, and M. Kukurba, "Start-up accelerator: State of the art and future directions," 2022.

[73] S. Lamdan, *Data cartels: The companies that control and monopolize our information*. Stanford University Press, 2022.

[74] K. Anderson, D. Visser, J. W. Mannen, Y. Jiang, and A. van Deursen, "Challenges in applying continuous experimentation: a practitioners' perspective," in *Proceedings of the 44th International Conference on Software Engineering: Software Engineering in Practice*, May 2022, pp. 107–114.

[75] O. Cico, "Lean software startup practices and software engineering education," in *Proceedings of the ACM/IEEE 44th International Conference on Software Engineering: Companion Proceedings*, May 2022, pp. 281–285.

Anitha Gracy is currently a full-time Ph.D. research scholar at Anna University Chennai pursuing her research at the research Centre – Department of Applied Mathematics and Computational Science, Thiagarajar College of Engineering, Madurai, India since January 2023. She has rich teaching experience in Research. Her current research interest includes software engineering, information systems, and data mining

Dr . Sudhaman Parthasarathy is currently working as a Professor of Data Science and HOD of Applied Mathematics and Computational Science at Thiagarajar College of Engineering, Madurai, Tamil Nadu, India. Previously, he was an Associate Professor of Computer Applications in the same institution. He has more than two decades of teaching and research experience. He completed his PhD in Computer Applications from Anna University, Chennai in 2009. He has published research papers in peer reviewed international journals from Elsevier, Springer and Emerald.

Dr . Sivagurunathan is an Assistant Professor in the Department of Computer Science and Application at Gandhigram Rural Institute.He received his Ph.D. from Anna University specializing in Computer Science.His research interests include Computer Networks,Cloud Computing and Internet of Things,and Network Security.He has been organizing several National level Workshops,seminars.