Journal of
Information Systems Engineering
and Business Intelligence

Vol.11, No.2, June 2025
Available online at: http://e-journal.unair.ac.id/index.php/JISEBI

# Aligning Software Product Management with Software Engineering Concepts: A Systematic Literature Review

**Chalani Oruthotaarachchi** [1)*], **Janaka Wijayanayake** [2)]

[1), 2)] *Department of Industrial Management, University of Kelaniya, Kelaniya, Sri Lanka*

[1)] chalani@kln.ac.lk, [2)] janaka@kln.ac.lk

***Abstract***

**Background:** Software Product Management (SPM) plays a vital role in the success of many software projects by aligning customer needs with their business objectives and ensuring a seamless and effective software product lifecycle. SPM is established as a collection of tools, techniques, and practices that help an organization accomplish its objectives and enhance the predictability and profitability of software product development. However, despite its significance, SPM research has been fragmented into specific topics having limited SPM literature reviews. This research study addresses this gap and discusses the status of the SPM domain in a more holistic spectrum.
**Objective:** The study aims to review recent literature on SPM, focusing on the alignment of SPM with software engineering concepts, a product manager's role, the existing framework, ontologies, and best practices that support ensuring the success of a product manager's role.
**Methods:** A systematic literature review was conducted using SCOPUS, IEEE Xplore, ACM Digital Library, ScienceDirect, and ProQuest Central as databases. 71 articles were selected following a rigorous screening process as per the PRISMA 2000 statement.
**Results:** Integrating SPM and SE is crucial in delivering value-driven software solutions. Available theoretical models and frameworks can help with this integration; however, implementing these frameworks often has challenges. Even though product managers play a vital role in the software lifecycle, they lack sufficient organizational support to enrich their skills and knowledge. Other major challenges are the lack of knowledge to use emerging technologies such as AI for data-driven decision-making processes and the tendency to replace humans with such technologies.
**Conclusion:** Aligning strategic vision with agile flexibility is important to integrate SPM with SE practices. To improve decision-making and ensure better alignment of SPM with business objectives, organizations have to enhance product managers' capabilities by leveraging emerging technologies. Research can focus on developing adaptable and user-friendly SPM frameworks that match both medium-scale and large-scale organizational expectations.

***Keywords:*** Organizational Value, Product Manager Role, Software Engineering Integration, Software Product Management, SPM Challenges, SPM Frameworks

***Article history:*** Received 25 November 2024, first decision 18 April 2025, accepted 26 May 2025, available online 22 July 2025

## I. Introduction

Software Product Management (SPM) plays a vital role in the success of many software projects [1]. It encompasses planning, designing, developing, launching, and maintaining a software product throughout its lifecycle. As a discipline, SPM involves overseeing the entire lifecycle of a software product from conception to retirement. This includes defining product vision and strategy, creating and managing product roadmaps, prioritizing features, and defining product requirements [2]. SPM can be defined as a practice that connects the company's strategy and the problems and needs of clients using the software product. Thus, the software product must help the company accomplish its strategic goals and solve the problems and needs of clients [3].

Researchers have given different definitions of product management. As per Hyrynsalmi, et al. [4], product management is the entire business management of products, product lines, or product portfolios to maximize value for their life cycle. An alternative definition is offered by Kittlaus and Fricker [5]. According to their definition, product management is the mix of goods and services that a supplier or development organization puts together to promote its business interests to transfer certain rights to the customer. The definition of a product manager given by Gorchels [6] seems to be the closest to our understanding. According to Gorchels [6], what the product is, how it

[*] Corresponding author

ISSN 2443-2555 (online) 2598-6333 (print) © 2025 The Authors. Published by Universitas Airlangga.

doi: http://dx.doi.org/10.20473/jisebi.11.2.143-159

functions, who it serves, and how it affects the business and consumers are all under the control of software product management.

Software product managers are responsible for managing the development of software products, ensuring that they meet customer needs and align with the company's business goals [2]. They are responsible for the entire product lifecycle, from idea generation to retirement of the project [7]. Moreover, they work closely with cross-functional teams, including development, design, marketing, sales, and support, to ensure that the product is delivered on time, within budget, and with the required features and functionality [8]. Tkalich, et al. [9] describe that a product manager's primary objective is to design experiments that will aid product teams in determining which features are required for new or existing software products. Product managers are also strongly committed to their teams, assisting with delivery, individual members, and general autonomy.

While SPM is currently widely recognized as an independent discipline, it is rooted and heavily connected to Software Engineering (SE). SE provides the technical foundation for software process, tools, architecture, and quality practices that software product managers need to be aware of and navigate. SPM applies these engineering concepts to determine what to build and why, while SE is about how to build it efficiently and effectively [10]. According to [11], [12], software engineering and product management increasingly align through shared responsibilities in requirements engineering, release planning, and quality management. Also, Lee and Chen [13] argue that SPM is important in bridging customer and market requirements with implementable engineering choices, thereby placing it at the interface of business strategy and engineering realization.

In DevOps and Agile scenarios, the alignment between SPM and SE is spotlighted especially as product managers must make rapid decisions with architectural impact, user impact, and technical debt impact [14]. Hence, current software engineering methods demand that product managers be technology-oriented and in a position to engage with the engineering teams through shared frameworks and languages. This convergence has brought about the emergence of value-based software engineering, with product value becoming a key indicator when making engineering choices [15].

Current literature highlights that agile development practices are significantly supported by high-quality SPM because effective scoping and planning contribute to the quality of software products [15]. Moreover, emerging technologies such as generative AI, digital twins, and big data analytics are reshaping product management as well as engineering practices, with the focus being on integrated approaches [16]. Therefore, an understanding of SE and SPM integration is not only a necessity for effective software delivery but also a prerequisite for leveraging modern software development practice [14]. Despite its importance, there are few literature reviews that comprehensively address the convergence of SPM and SE.

To date, product management has gained great popularity in the software field and has become a strategic business function that directly influences business success. Thus, Software Product Management (SPM) has also gained prominence as an area of software engineering research [17]. Still, despite this increased popularity, existing research remains fragmented with a focus on isolated themes such as specific challenges [10], [17], [18], best practices [19], or new technologies' integration into decision-making within products [16].

Notably, few systematic literature reviews have attempted to explore the SPM domain, and the few that exist are narrow in scope or highly technical. The most recent comprehensive SPM literature review was in 2021 [4], so there is a lack of up-to-date, broad insights that take into account the rapid pace of development in the field, particularly concerning agile methodologies, AI-driven tools, and cross-functional team engagement. Besides, most of the earlier reviews lack the critical intersection of SPM with Software Engineering (SE) where practice, tool, and responsibility converge to deliver customer value.

This study fills a clear research gap by offering a systematic review that is holistic in its approach to SPM, addressing key themes such as: the convergence of SPM and SE concepts, the evolving roles of software product managers, the application of ontologies and frameworks in guiding SPM practice, and best practices that allow effective product management. To our knowledge, no comprehensive literature review has addressed these dimensions as a single entity, offering an integrated view of the SPM field. This research will not only contribute to academic research, but also to product management professionals and alignment of software practice with strategy. The results will have implications for future research and building scalable, adaptive SPM frameworks suitable to today's development environments.

The remainder of this paper is structured as follows. Section 2 discusses existing literature, Section 3 discusses the research methods, Section 4 presents research results, Section 5 presents a discussion, and Section 6 concludes the paper by suggesting future research.

.

## II. Literature Review

### A. Evolution of SPM concept

Product management practices have grown significantly in popularity and importance within the software industry. Software product management strategies have been adopted and put into practice by numerous prosperous firms, including market leaders such as Microsoft, IBM, and Google, to improve their product development procedures and increase their market performance [10]. Considering the "Big Tech" companies' success stories, many IT and software firms are updating their project-based strategies to product-based objectives [18].

The early pioneers of SPM were often software developers who recognized the need for a more holistic approach to software development. They began to apply product management principles, such as market analysis and customer research, to software development and created the foundation for the modern practice of software product management. As per the current practices, a software product manager's work is key in determining a product's commercial viability and the overall success of the firm that invests in the project [5]. The institutionalization of this job has been shown to increase project success rates and produce more favourable results dramatically [10]. To date, SPM is a well-established discipline with its own best practices, frameworks, and tools. It continues to evolve and adapt to changing technological trends and business needs

### B. SPM and SE integration

SPM is strongly linked with SE, highlighting an essential link between customer requirements, business strategy, and technical reliability. Whereas SE is concerned with systematic methods for developing, operating, and maintaining software systems, SPM is responsible for ensuring the correct product is developed, aligning with user requirements and business objectives [20]. Therefore, SE competence is needed for product success, and vice versa as technically sufficient products can fail to deliver value without proper product management [13]. High-quality software products, as expressed in high-quality codebases, maintainability, scalability, and team productivity, are the cornerstone of successful software product delivery. Product managers contribute to achieve software quality by establishing clear product goals, prioritising features based on customer value, and keeping technical decisions aligned with market needs. Agile, DevOps, and Continuous Integration/Continuous Deployment (CI/CD) are some examples of SE paradigms that enable successful product management involvement, particularly in rapid iteration and decision-making [21].

Conceptually, Requirement Engineering, architecture design, release planning, and quality assurance are SE processes requiring constant coordination with product managers [13]. Product managers are intended to translate software quality attributes (e.g., performance, security, usability) and transform them into executable requirements, facilitating cooperation in multidisciplinary teams. Several frameworks and ontologies have been proposed to facilitate this SE and SPM integration. The ISPMA (International Software Product Management Association) framework offers a structured method outlining the key activities and responsibilities of software product managers across the product lifecycle [22]. The SEMAT (Software Engineering Method and Theory) Essence framework [23] also offers a foundation for modeling similarities in SE practices. By SEMAT Essence framework product managers and engineers share a common understanding of development activities and progress [23].

These best practices to support product managers include collaborative roadmapping, value-based feature prioritization, and active participation in Agile and DevOps practices [20]. Moreover, Value-Based Software Engineering (VBSE) aims at the optimisation of product value through integrated engineering and management decisions [24]. In today's rapidly changing environments, technically aware product managers play a crucial role in making trade-offs between user requirements, technical feasibility, and business objectives. Recent research still emphasizes the need for VBSE, where feature implementation, technical debt, and design choices are led by the expected customer value. In this case, product managers are not only strategists but also part of the technical discussions as active contributors to balance innovation, expense, and customer satisfaction [25].

As software systems increase in complexity, the synergy of SE and SPM becomes increasingly important. New trends such as AI-supported development, platform ecosystems, and software sustainability require product managers with a deep understanding of both business and engineering to lead with agility, clarity, and impact.

### C. Systematic Literature Reviews in SPM domain

Several research studies have been undertaken (for instance, [17], [26], [27], [28], [29]), each focused on the literature review of different product management strategies in various domains such as manufacturing, marketing, engineering design, product road mapping, and product line management. However, literature presents a very limited number of research studies specifically focusing on literature reviews in software product management.

Ebert and Brinkkemper [20] conducted their study to discuss about an overview of software product management practices and tools. It compiles lessons learned through introducing, enhancing, and deploying the product manager role in the IT industry. Based on the results, they conclude that software product managers are the glue that holds operations, marketing, and engineering together. Product managers must maintain a 360-degree view of their company's operational activities while maintaining a strategic perspective on product prospects and strategy. Given these duties, the skill sets of product managers are wide, thorough, and hard. Investing in the development of these skill sets is a top priority.

Hyrynsalmi, et al. [4] conducted their research study to review the evolution of Software Product Management research. This study was conducted based on a bibliographic approach focusing on the publications available in Scopus. The identified research was created by a small group of authors and institutions, most of which are based in Europe. The study also suggests that the SPM field may lack an intellectual foundation and new opportunities because of the decentralized research focus. The authors propose creating a single research agenda for the Software Product Management area to prevent this.

Demirel [18] in their study reviews the literature related to SPM to provide a comprehensive overview of the product management profession. They investigated the similarities and differences between these two professions and specifically focused on transforming the project manager's role into the product manager's role. The study concluded that the software product manager’s role is ever evolving because of the digital revolution. Software product managers lack a well-defined structure and even an educational background and must adapt to numerous difficulties across multiple disciplines. Accordingly, the study highlights that, combined with an academic approach and practical experience, it is necessary to understand the actual needs of productization in software organizations.

Springer and Miler [10] investigated the difficulties that affect the software product management process and their perceived frequency and severity. They used articles from five databases that described research conducted within software development companies. They identified 95 unique software product management problems from their review and further filtered them using a survey conducted with industry practitioners. Results highlighted that software product managers' major problem is determining the true value of the product that the customer needs. This is one of the most frequent problems that product managers face, and it greatly impacts the effectiveness of product managers’ job roles.

Parikh [16] conducted a review study to identify the use of generative AI in software product management practices. The findings highlight that SPM can integrate generative AI into several phases of the product development process, from ideation to execution, such as idea generation, UI/UX design, market research, product design, and requirement elicitation. However, it is crucial to understand and address any possible drawbacks and difficulties associated with applying generative AI solutions to software product management.

TABLE 1
SUMMARY OF RELATED LITERATURE

| Reference | Objective | Limitation |
|---|---|---|
| [20] | Provide an overview of software product management practices and tools, highlighting the role of product managers as a bridge between operations, marketing, and engineering while emphasizing the need for skill development | Limited to general lessons learned in IT industry practices, lacks specific strategies for addressing identified challenges |
| [10] | Investigation of SPM challenges and their frequency/severity through a literature review and industry survey. | Limited to identified problems without providing detailed solutions or actionable strategies for overcoming these challenges. |
| [16] | Review of generative AI applications in SPM, highlighting its potential in ideation, UI/UX design, market research, and product design. | Focused on generative AI applications, leaving broader SPM integration and other emerging technologies unaddressed. |
| [4] | Review of the evolution of Software Product Management research using a bibliographic approach. | Focused only on European research outputs and lacks practical guidelines for building a cohesive research agenda. |
| [18] | Provide a comprehensive overview of the product management profession and address challenges due to the digital revolution. | Emphasizes the evolving role without providing concrete solutions or frameworks for addressing structural and educational gaps in the profession. |
| [30] | Conduct a systematic review of software product managers' roles and responsibilities. | Frameworks are theoretical and require validation or application in diverse organizational contexts. |
| [17] | A systematic review of software product manager’s role in software startups. | Identified several software product manager’s critical tasks related to software startup success. |

Parikh [30] conducted a systematic review to explore the roles and responsibilities of software product managers in the software development lifecycle. They present frameworks that outline the comprehensive job of software product managers: the Software Product Manager RACI Framework (SPM-RF) and the Software Product Manager

Life Cycle Model (SPM-LCM). These frameworks organize product managers' tasks, responsibilities, and accountabilities, showing how they engage with cross-functional groups and give a thorough grasp of their duties.

Pattyn [17] conducted a systematic review of the software product manager's role in software startups. They identified several software product manager's critical tasks related to software startup success. The authors use findings as a starting point to develop a framework that guides the product manager's role in software startups.

A summary of the existing literature, along with their limitations, is presented in Table 1. According to these reviews, further research is needed to address knowledge gaps, consolidate existing knowledge, and establish a cohesive intellectual foundation for SPM. Additionally, there is a growing recognition of the importance of investing in developing product managers' skills and leveraging emerging technologies to enhance product management practices effectively.

## III. METHODS

This study conducted an in-depth review of SPM and software product managers' roles published between 2003 and 2023 to systematically analyze recent literature on SPM, providing insights into its challenges, opportunities, and frameworks to support future research and practical advancements. This study addresses the following questions.

RQ 1. How is SPM integrated with software engineering concepts?
RQ 2. What is the role of a product manager in a software development project?
RQ 3. What are the frameworks/ontologies available to support software product manager's work?
RQ 4. What are the challenges the software product managers face when performing their role?
RQ 5. What are the solutions suggested to overcome these challenges?

### *A. Search strategy*

The search strategy included querying databases, including SCOPUS, IEEE Xplore, ACM Digital Library, ScienceDirect and ProQuest Central. The search terms were defined by aligning the research questions with all related search categories to the research questions using PICO criteria [31]. In PICO, the Population refers to the target entity and group of the study. Intervention refers to the action or focus of the study. Comparison refers to any alternatives being compared, and Outcome refers to the impact of the study. Table 2 presents the keywords used in search query.

TABLE 2
INCLUSION AND EXCLUSION CRITERIA

| Criteria | Keywords |
|---|---|
| P (Population) | Software Product Manager, software Engineer |
| I (Intervention) | Product Manager role, integration |
| C (Comparison) | Software Product Management, Software engineering |
| O (Outcome) | Software product management frameworks, software product management challenges, software product management trends, software product management ontology, software value |

Then, as mentioned below, a comprehensive search query was built by combining the keywords with "AND" and "OR" operators:

("software engineering" AND "software product management" AND "integration") OR ("software product manager" AND "product management role") OR ("software product management" AND "software product manager") OR ("software product management" AND "software product management frameworks") OR ("software product management" AND "software product management ontology") OR ("software product management" AND "software product management challenges") OR ("software product management" AND "software product management trends") OR ("software product management" AND "software value").

TABLE 3
INCLUSION AND EXCLUSION CRITERIA

| Criteria | Description |
|---|---|
| IC1 | The article is available in full text in the academic databases we selected for this study. |
| IC2 | Articles published in a journal, conference proceedings, books, a workshop co-related with a conference. |
| IC3 | Full articles that discuss software product management, software engineering and software product management integration, software product manager professionals' role and their responsibilities, software product management frameworks, ontologies, and guidelines, challenges in software product management, and trends in software product management |
| EC1 | Articles that are not published in English. |
| EC2 | Inaccessible Articles (due to firewall and payment restrictions) |
| EC3 | Short papers, work-in-progress proposals, keynotes, editorials, and non-peer-reviewed articles |
| EC4 | Duplicative literature |

### *B. Inclusion criteria and Exclusion criteria*

After retrieving the search query, we listed the inclusion criteria (IC) and exclusion criteria (EC). With the help of these criteria, listed in Table 3, 579 articles that are more relevant to the study were picked.

### *C. Article quality assessment and filtration process*

The Preferred Reporting Items for Systematic Review and Meta-analysis (PRISMA) statement was followed to conduct this study. This method was specifically selected because it offers revised reporting guidelines for systematic reviews, reflecting advancements in study identification, selection, evaluation, and synthesis methods [32]. Fig 1 shows the article filtration process of this study in a PRISMA flow diagram.

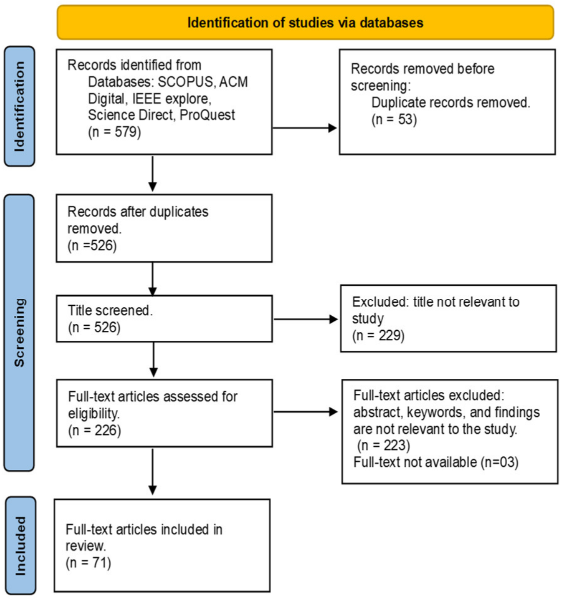
Identification of studies via databases
Identification
Screening
Included
Records identified from Databases: SCOPUS, ACM Digital, IEEE explore, Science Direct, ProQuest (n = 579)
Records removed before screening: Duplicate records removed. (n = 53)
Records after duplicates removed. (n =526)
Title screened. (n = 526)
Excluded: title not relevant to study (n = 229)
Full-text articles assessed for eligibility. (n = 226)
Full-text articles excluded: abstract, keywords, and findings are not relevant to the study. (n = 223) Full-text not available (n=03)
Full-text articles included in review. (n = 71)

Fig. 1 Article Inclusion and Exclusion Process as per PRISMA 2000 Statement

On the flow diagram, "n" represents the number of records. PRISMA 2000 statement describes three important steps, namely, 1) identification, 2) screening, and 3) inclusion, that help increase transparency in systematic literature review reporting purposes. This study initially identified 579 articles from five (05) academic databases, namely, SCOPUS (n=149), IEEE Explorer (n=118), ACM Digital (n=109), Science Direct (n=112), and ProQuest (n=91). All identified 579 articles were screened in four criteria: (1) duplicate articles, (2) availability of full text, (3) relevance of the paper title to this study, and (4) absence of clear parameters in keywords, abstract, and findings. 508 articles were removed during the screening stage due to duplication (n=53), irrelevance of the article (n=229), absence of clear parameters in keywords, abstract, and findings (n=23), and unavailability of full text (n=03). At the end of the screening stage, 71 articles were selected for rigorous literature review. As illustrated in Fig 2, most of the filtered articles are from IEEE Explore (n=21), and ACM Digital Library (n=18). The remaining 32 articles were distributed as SCOPUS (n=12), ScienceDirect (n=11), and ProQuest (n=09).

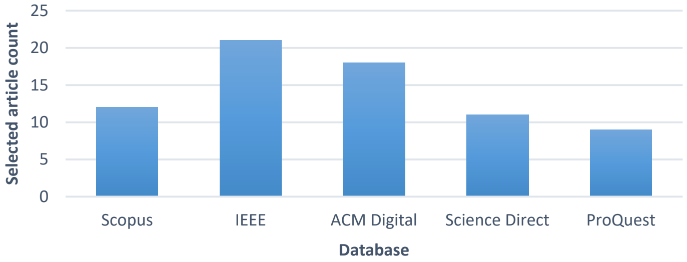

Fig. 2 Selected Article Counts per Database

## IV. Results

### *A. Summary of the studies*

Out of the 71 articles, 51 (61%) were in journals, 16 (20%) were in conference proceedings, 9 (12%) were published in workshop proceedings, and the remaining 6 (7%) were in books or book chapters. Fig 3 presents these distributions. This analysis concludes that the journals discuss the software product management topics more than the other sources.

All the included articles were published between 2003 and 2023, and Fig 4 displays the distribution of publication counts in each year. Except for 2004, 2005, 2007, 2016, and 2021, all other years have at least one publication. The peak time of the publications is 2010 and 2022, with 11 publications per year. Overall, we have 2009, 2010, 2011, and 2022 with 8 or more publications and 2006, 2012, and 2018 with 6 or more publications.

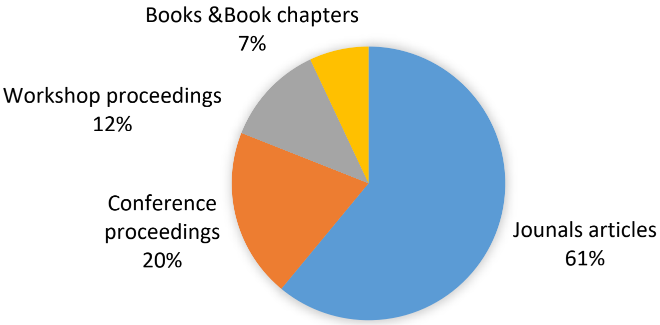

Fig. 3 Categorization of Selected Articles According to The Publication Type

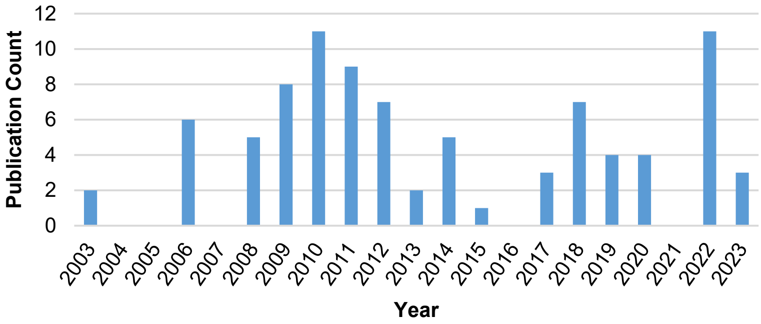

Fig. 4 Distribution of Publication Counts in Each Year

### *B. Integration of software product management with software engineering (RQ 1)*

Software Engineering (SE) and SPM are two of the most vital functions in a software organization; thus, both software engineer and product manager positions are essential to organizational success [18]. Whether the company is just getting started or has escalated quickly, maintaining alignment between software engineering and product management is critical to long-term success [33].

Van de Weerd and Katchow [34] explored the relationship between SPM and defect management in a distributed setting. They introduced a conceptual model combining four domain concepts: issue, market requirement, product requirement, and software defect to assist product managers and defect management practices. This conceptual model can be used as the starting point for generating SPM support tools.

Van de Weerd, et al. [35] conducted a study on the evolution of incremental methods in global software product management. The researchers applied the collection of method increment approaches they identified in a previous study based on method engineering principles [36] into an ERP implementation case study. They identified several points for the success of global-level software product management companies: changing business strategy, critical shared infrastructure, and global involvement.

Delivering valuable software to clients is the top priority, according to the first principle of the Agile Manifesto [37]. In this light, several research studies have discussed applying product management to agile software development approaches. Vlaanderen, et al. [38] discussed applying agile SCRUM practices to product management practices. The researchers have extended the SCRUM process with the "agile requirements refinery" named agile SPM, which gives product managers the ability to handle complicated requirements in an agile development environment. The introduced new process was applied to a real-world case study, and effective agile SPM practices were identified, such as the task size, backlog structure, and willingness to keep the backlog updated. Kittlaus [39] explained the conflicts between SPM and agile development and solutions that can address the identified conflicts. The researchers suggest that their findings can be used to enhance the productivity of an organization. Moreover, they highlight that the SPM Framework in the ISPMA Book of Knowledge is highly beneficial for analyzing a situation in an organization and outlining particular solutions.

In a background of accomplishing the alignment of SPM with software architecture (SA) design was scientifically unknown, Lucassen, et al. [40] outline the most crucial steps for SPM and SA alignment: collecting requirements and refining. The study highlights that the success of these steps requires effective communication backed up by high-level architectural perspectives. Based on their findings, the researchers propose the Accurate Architectural Models Approach (AAMA) to prevent architectural model divergence and assist the collaboration between software product managers and architects.

Table 4 summarizes the focus areas, methodologies, and key findings of notable studies addressing the integration of software product management with various aspects of software engineering.

TABLE 4
SUMMARY OF KEY STUDIES ON SPM AND SE ALIGNMENT

| Study | Focus Area | Key Findings |
|---|---|---|
| [34] | SPM and defect management | Introduced a model integrating issues, market requirements, product requirements, and software defects. |
| [35], [36] | Incremental methods in global SPM | Identified factors for global SPM success: strategy changes, infrastructure, and global involvement. |
| [37], [38] | Agile SPM | Developed an "agile requirements refinery" to manage complex requirements; identified key agile SPM practices. |
| [39] | SPM and agile development conflict resolution | Highlighted the ISPMA SPM Framework as a key tool to resolve conflicts and enhance productivity |
| [40] | SPM and software architecture alignment | Suggested AAMA to prevent divergence and improve collaboration between product managers and architects. |

### C. *Role of a product manager (RQ 2)*

Product managers are responsible for playing a crucial role in the success of software product-oriented companies [41], [42]. Software product managers are responsible for managing the development of software products, ensuring that they meet customer needs and align with the company's business goals [2], [43]. As Table 5, a software product manager’s role is a cross-discipline that combines company strategic objectives identification, understanding customer problems and needs, and available technology applications. Thus, a product manager needs sound knowledge of three domains: business analysis, UI/UX engineering [9], and human-computer interaction [44]. Moreover, they require a deep understanding of the market, customer needs, and the competitive landscape to succeed in the software product manager's role [45]. Effective communication, collaboration, and project management skills are essential for managing cross-functional teams and stakeholders throughout the product lifecycle [46].

Companies must have a comprehensive product strategy in place to produce their products/services in an environment that is becoming more complex. The need to create an end-to-end flow between client demand and the quick delivery of a good or service is increasing because of the users' ever-changing expectations [9]. The Product Owners (PO) in agile software development, such as Scrum, oversee this flow. POs map customer demands and ensure that the final product is profitable by converting business requirements into workable software specifications, gathering and prioritizing requirements, and authorizing the software development before it is released to customers [47]. However, the PO’s role is not sufficient to map client demands. Fitzgerald and Stol[48] explain that there should be a manager who is dedicated to identifying features that optimize the value of the product in a systematic way and promptly inspecting the features' delivery, delivery costs, customer usage, and real return on investment. A product manager's responsibility is constantly building product portfolios and connecting them to consumer demand [49], [50]. As explained in the Software Product Management Body of Knowledge (SPMBoK) the PMs are responsible for product planning, product strategy development, product development orchestration, marketing, sales, and support.

Even though the product manager's role is a mix of technical and managerial responsibilities, most product managers in software and IT come from a technical background and evolve into this field without any formal education [7], [51]. Product managers in Europe and Asia typically progress from technical roles such as technical project manager. There is a lack of standard educational pathways to software product management [52].

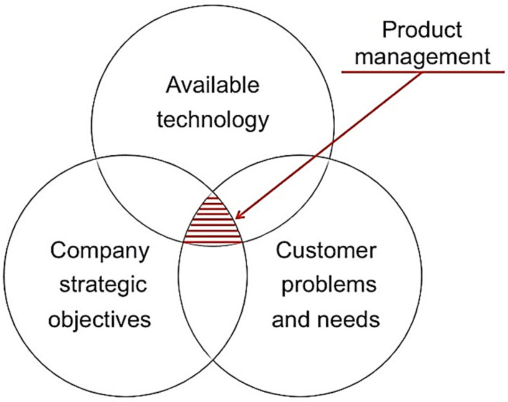

Fig. 5 Cross-discipline Nature of Software Product Managers' Role (Source: [1])

Product management duties are critical in software firms because they enable decision-making and the development of products per the corporate strategy. Product management is distinct from project management by definition; yet, in many software organizations, the responsibilities of product manager and project manager are merged [53]. Simply, the product manager defines what should be done in the project, and in contrast, the project manager defines how should the work be carried out [54]. Product managers, like project managers, must have strong management abilities because they frequently lead teams, projects, or even departments. Communication skills are required to effectively manage stakeholders and collaborate with development teams to implement product roadmaps [2]. Table 5 presents a summary of the product manager's responsibilities.

TABLE 5
SUMMARY OF PRODUCT MANAGER'S ROLES AND RESPONSIBILITIES

| Article | Role/ Responsibility | Description |
|---|---|---|
| [2], [34], [44], [45] | Match products to customer needs and corporate goals | Confirm that the developed products satisfy market demand and conform to corporate goals |
| [9], [44] | Leverage cross-disciplinary expertise | Apply business analysis, UI/UX engineering, and human-computer interaction expertise to product design |
| [46] | Lead cross-functional teams and stakeholders | Use communication, cooperation, and project management skills throughout the life of the product |
| [40], [41] | Lead product portfolios | Construct and realign product portfolios on an ongoing basis in accordance with consumer demand |
| [48], [50], [51] | Optimize product value | Establish high-value product attributes, track delivery, report ROI, and respond to actual usage data |
| [9], [47] | Convert business and customer requirements into actionable plans | Convert requirements to software specifications, feature prioritization, and product delivery plan |
| [53], [54] | Make strategic product decisions | Decide what to develop and deliver in alignment with organizational strategy, independent of project execution |

### *D. SPM frameworks and ontologies (RQ 3)*

To achieve a more mature SPM process, a number of theoretical frameworks and models have been suggested over time. These models, not only guide the practical application of SPM activities but also provide an underlying framework for SPM analysis and improvement in different organizational settings.

#### *1) Tradional Process Improvement Models*

The Capability Maturity Model (CMM) [55] and Capability Maturity Model Integration (CMMI) [56] are generally accepted models that were originally developed for software process improvement. While these models offer structured guidance for process enhancement, their rigid, resource-intensive design limits their applicability to the volatile and often resource-constrained environments of small and medium-sized enterprises (SMEs) [57], [58]. In addition, CMMI as well as other standards like ISO/IEC 15504 (SPICE) are also often criticized for being excessively complex and static in nature in dynamic product-oriented environments [59], [60], [61]. Thus, although they are conceptually robust, these models face practical adoption barriers in agile and product-driven organizations

*2) Evolution of SPM-Specific Frameworks*

Recognizing these shortcomings, Van De Weerd, et al. [62] proposed one of the earliest specific SPM frameworks. The model breaks down the activities of the software product manager into four broad areas of work—portfolio management, product planning, release planning, and requirements management—and eight market-oriented sub-functions, being able to effectively cast the software product manager's internal process interaction with market-oriented activities. This framework marked a milestone for the standardization of the SPM position and organizational function transition of SPM.

Kittlaus and Fricker[5] expanded the work by Van De Weerd, et al. [62] into a framework that included concepts from the Pragmatic Marketing Framework [63] and product lifecycle management guidelines adopted from Ebert [50], thereby creating a more robust framework that considers product lifecycle, market influence, and strategic alignment. This consolidation centers on the role clarity, in particular the differentiation and coordination of product managers and product marketing managers and improves the strategy component of SPM. Fricker's framework [51] complements this by being a road map of key tasks and giving an overlapping vocabulary within product teams required for cross-functional cooperation in agile environments.

*3) Knowledge Infrastructures and Maturity Models*

To enable ongoing improvement, Van de Weerd and Brinkkemper [64] proposed the Product Software Knowledge Infrastructure (PSKI), an reusable knowledge base of best practices and case-based process knowledge. This concept, further developed by Vlaanderen, et al. [65], enabled organizations to capture and analyze SPM activities using meta-modelling tools like MetaEdit+, which resulted in process assessment automation. Bekkers, et al. [66] and Bekkers, et al. [67] addressed SPM capability measurement by creating a competence model and a maturity matrix grounded in empirical knowledge gained from 62 case studies. These tools allow organizations to chart weaknesses (e.g., blocking levels/questions) and support progress through evidence-based SPM improvement approaches. Quantitative validation of these models gives them greater credibility and generalizability, particularly in application settings.

In order to add additional structure and conceptual clarity, Botzenhardt and Maedche [68] began developing an SPM domain ontology, outlining a formalized, shared vocabulary for SPM components. Their ontology—influenced by the model proposed by Van De Weerd, et al. [62] —was further elaborated in an all-inclusive model by Botzenhardt, et al. [69], which organized the domain into four basic components: activity, artifact, organization, and role. All of them were divided into subcomponents to organize systematically the richness of SPM activities. While the original ontology was developed when the SPM field was not yet mature, current development of the field suggests the promise of refining this model as an even more flexible and general one.

Table 6 presents an overview of the discussed SPM ontologies and frameworks, emphasizing their focus areas and contributions to advancing SPM practices.

TABLE 6
KEY ONTOLOGIES AND FRAMEWORKS FOR ADVANCING SPM

| Ontology/ Framework | Focus Area | Contributions |
|---|---|---|
| [55], [56] | Software process enhancement | Discusses how the general software development process can be formalized. |
| [62] | SPM framework design | Identified four main areas (portfolio management, product planning, release planning, requirements management) and sub-functions aligned with stakeholders and designed the SPM framework. |
| [5], [51] | Integrated SPM framework | Unified Van De Weerd's framework with Pragmatic Marketing and lifecycle management concepts, offering a blueprint for product teams. |
| [64] | Process improvement | Provided an infrastructure (Product Software Knowledge Infrastructure (PSKI)) to support product firms with process-deliverable diagrams and customizable recommendations based on case studies and procedural data. |
| [66], [67] | SPM maturity assessment | Enabled evaluation of organizational practices and identification of best practices; refined through quantitative analysis of 62 case studies. |
| [68] | SPM domain ontology | Defined core components of the SPM domain (activity, artifact, organization, and role) in terms of SPM ontology. |
| [69] | Expanded SPM ontology | Extended the ontology into a comprehensive model, proposing refinements to reflect SPM's current maturity. |
| [65] | SPM knowledge preservation | MetaEdit+ Process-Deliverable Diagrams: Incorporated SPM processes into process-deliverable diagrams for structured evaluation and reuse. |

### *E. Solutions suggested to overcome the SPM challenges (RQ 5)*

Research on Software Project Management (SPM) challenges has generally focused on managing issues related to decision-making across the SPM lifecycle, requirement prioritization, managing releases and increments in the product delivery process, and adopting effective change management strategies.

Release planning, in which features are assigned to releases, is a critical step in software product management [70]. Stakeholder preferences, volatile features, and resource availability have all been identified as factors that lower the quality of releases in SPM. AlBourae, et al. [71] offer a lightweight replanning process model in which old and new features are compared using the Analytical Hierarchy Process (AHP). The researchers used a replan algorithm to identify the most promising features in response to changing market-driven product demands.

SPM relies heavily on decision-making, which includes deciding on requirements priority and the content of upcoming releases [72]. Many algorithms have been introduced for release planning and prioritization, in which individuals carry out a sequence of actions to generate a decision with or without the assistance of a machine. Some examples are binary search tree algorithm [73], cumulative voting algorithm [74], priority grouping algorithm [75], and APH algorithm [76]. Rather than using a particular algorithm to find an acceptable solution to a decision problem Regnell and Kuchcinski [77] suggest modelling SPM decision-making as a Constraint Satisfaction Problem (CSP). In CSP, they suggest identifying the requirement priorities relating to variables such as feature priorities, stakeholder preferences, and resource constraints. Saltan, et al. [78] highlight that the decision-making practices in SPM should be more transparent and evidence based. They suggest that moving from intuition-based decision-making to data-driven decision-making is the best practice to eliminate the requirement of prioritizing issues in the SPM process. Several research studies highlighted that there is a significant gap between data analysis and data-driven decisions [79], [80], [81], [82]. Product managers must extract value from analysis and effectively communicate insights thus, Lin [83] suggests that data storytelling—using narrative and data visualization to communicate your insights and influence business decisions—is the finest method for bridging the gap. Parikh [16] evaluates generative AI's applications, benefits, and constraints in SPM. The study specifically focuses on the role of generative AI in requirement elicitation, software product design, and product development. They highlight that software product managers can use generative AI technologies to automate time-consuming repetitive tasks, make data-driven decisions, and improve overall product quality. Moreover, Liu, et al. [84] has discussed how smart enabling technologies such as IOT, AI, Cloud computing, Big data analytics, and Digital twins can be used for product lifecycle management and the issues of applying these technologies.

Literature has discussed five common SPM challenges: the long release cycle, unavailability of the matrix for evaluating work, maintaining collaboration between organizations and customers, short-term thinking, and instant changes in requirements. Springer, et al. [19] based on focused group discussions and survey findings, propose general solutions to these challenges. They highlight that managing the workflow, using value to identify the performance indicators, investing more time in product analysis, developing a long-term strategy based on excellence areas, and introducing simple and incremental changes would facilitate the identified challenges. Maglyas, et al. [85] discuss how lean principles can mitigate the same five common problems. Product management and lean practices share characteristics such as the importance of value and attention to client needs [86]. In this light, the researchers highlighted that lightweight lean practices for SPM allow businesses to focus on the most important and simple product management practices while constantly improving. Table 8 summarizes the solutions discussed in this section.

TABLE 8
SUMMARY OF THE SOLUTIONS FOR SPM CHALLENGES

| Article | Solution | Description |
|---|---|---|
| [67], [68], [69], [70], [71], [72], [87], [88], [89], [90], [91] | Data-Driven Decision-Making | Data analytics, algorithms (e.g., AHP, CSP, cumulative voting), and formal decision models can be used to support prioritization and release planning. These enable the shift from experience-based decision making to data-driven decision-making. |
| [83] | Communication & knowledge Sharing | Use of data storytelling (with the help of narratives or visualization tools) help to bridge the data analysis–business decision gap, helping product managers to convince stakeholders. |
| [16], [84] | Smart & Emerging Technologies | Manual activities such as product development, requirements gathering, and lifecycle management can be automated by adopting AI, IoT, cloud, digital twins, and generative AI. |
| [85], [86] | Lean & Agile Practices | Adopting lean philosophy and minimalistic workflows into software process with emphasis on value delivery, customer requirements, and incremental and continuous improvement. |
| [19] | General SPM Best Practice | Alignment of KPIs to everyday workflow, improvement of product analysis, long-term strategy development, and monitoring frequently changing requirements. |

## V. Discussion

### *A. Integration of SPM with software engineering concepts (RQ 1)*

Integrating SPM and SE is crucial in delivering value-driven software solutions. Theoretical models can provide helpful frameworks ([34], [38], [39]); however, implementing these frameworks often has challenges. For example, aligning theoretical frameworks and a company's long-term goals with software development processes can be difficult because of conflicting short-term goals [35]. This highlights the need for practically acceptable tools that support real-time collaboration while keeping the software's overall goals and objectives on track. Agile development methods are commonly used in software engineering that offer benefits and challenges in achieving this balance. However, the iterative practices of Agile methods can sometimes clash with the more structured and goal-oriented approach of SPM [38]. To overcome this conflict, a balanced approach that combines the adaptable nature of agile methods with the strategic focus of SPM should be introduced. For example, agile practices can be improved with periodic strategic reviews, to ensure that short-term decisions align with the company's long-term objectives.

### *B. The role of a product manager in a software development project (RQ 2)*

The role of the product manager is crucial to the software development process but is frequently misunderstood or underestimated [41], [42]. To perform the product managers' role effectively, they must have technical expertise [9], [44], market understanding [45], [49], [50], and stakeholder management skills[46]. However, many software development organizations do not provide sufficient training or recognition for product managers' responsibilities [7], [51], [52].

Product Managers must collaborate with software technical teams, business stakeholders, and customers to ensure that software products are functionally accurate, relevant to the market, and aligned with the customer organization's goals and objectives [2], [53]. Offering professional development programs that focus on the unique skills required for product management can increase the efficiency of their job role and its impact.

### *C. Frameworks/ontologies available to support software product managers' work (RQ 3)*

The SPM-related frameworks presented in the literature *(e.g.:* [62], [64], [68]) are often difficult to adapt to organizational contexts. For example, these frameworks are too complex to work with or unnecessarily resource-consuming for small and medium-sized enterprises (SMEs) (e.g.: [66], [67]). On the other hand, large-scale organizations may expect scalable frameworks to avoid excessive bureaucracy. Developing more flexible frameworks that adapt to specific organizational needs could support avoiding these issues. Such frameworks will make SPM practices applicable to a wide range of organizations [69]. Furthermore, integrating SPM with the organizational strategic planning processes will help to align product decisions with broader business goals.

### *D. Challenges that software product managers face when performing their role (RQ 4)*

A major challenge in SPM is decision-making related to unclear project scope creeps [87], [88], misaligning with stakeholder expectations [89], or limitations of resources [90], [91]. Even though data-driven decisions are considered more accurate and effective, many organizations struggle to analyse and apply their data correctly [10].

In many instances, SPM is treated as a support function rather than a strategic driver [9]. This consideration restricts its integration with organizational goals accurately. Additionally, as organizations frequently adopt emerging technologies, several ethical concerns, such as AI system biases, appear. Less availability of skills development programs specific to product managers' job role expectations also has become a major challenge.

### *E. Solutions suggested to overcome these challenges (RQ 5)*

Using technologies combining advanced analytics with user-friendly interfaces could be a possible solution to help product managers quickly and effectively analyse data and make appropriate decisions [73], [74], [75]. Emerging technologies like machine learning and artificial intelligence (AI) can transform current human-oriented SPM practices [16]. Product managers can focus more on strategic planning by using AI to assess customer feedback or prioritise software requirements. However, practising these technologies raises ethical issues, such as biases in AI systems [79], [80], [81], [82]. Most importantly, it is necessary to highlight that these technologies are used only to enhance human judgement rather than replace humans.

A key for enhancing SPM practices is realigning organisational attitudes, acquiring the necessary expertise, and using new tools, technologies, and techniques [19]. By making investments in the knowledge, resources, and organisational frameworks that enable product managers, SPM's influence can be greatly increased and its potential as a vital source of innovation and value can be realised [85]. Its efficacy can also be increased by encouraging a culture that sees SPM as an important factor in company performance rather than merely an expense.

### *F. Limitations and threats to the validity of the results of this study*

This study has several limitations. We could receive additional literature to review by expanding the number of digital libraries. However, as we discovered with the Science Direct, IEEE, and ACM Digital libraries, much of the research would be duplicated. Thus, increasing the number of libraries would not significantly improve the number of articles. The databases selected include scholarly journals but have fewer industrial experience reports from practitioners such as software product managers and business analysts. Although industry reports are particularly useful in identifying the foundations of a study, they were removed from this research in favor of scientific journals to ensure the consistency of this research study.

Various factors have compromised the research's validity. The very first issue is with the search conditions used. Each of the databases has different search options. As a result, we customized the searches for each database to make the search as similar as possible. Furthermore, database search engines operate differently, and verifying that all the papers have been collected is impossible. We minimized this risk by utilizing five digital libraries expecting that the papers would be discovered by at least one search engine. The presence of publication bias is the second issue. We may have missed important articles since our search criteria were based on broad keywords like software product management and product management challenges. We tried to reduce this risk by using as many appropriate keywords as possible. The final issue is associated with the data extraction method. It might be done in various ways, so we explained the extraction method used in the research as much detail as possible in this article.

## VI. CONCLUSIONS

This research gives a comprehensive and thorough synthesis of the Software Product Management body of research with specific focus on its integration with Software Engineering practice. The key contribution of this work is the development of a holistic view of contemporary SPM practices through a systematic literature analysis—highlighting how SE affects SPM, changing nature of product managers' roles and responsibilities, ontologies and frameworks applicable for SPM domain, and challenges to the SPM practice. The available literature provides fragmented findings and contents under these topics without providing an integrated perspective. This research is filling in this literature gap undoubtedly, by focusing on a holistic view of SPM domain.

The findings of this study highlight that SPM provides tools and techniques to help an organization accomplish its objectives and enhance the predictability and profitability of software product development. Ontologies and frameworks are essential for improving SPM techniques. Moreover, integrating SPM and SE practices plays a major role in delivering valuable software solutions. Though theoretical frameworks are available in the literature to provide this integration, implementing them often has challenges. Research can focus on developing adaptable and user-friendly SPM frameworks that match both medium-scale and large-scale organizational expectations.

The role of the product manager is crucial to the software development process. To perform the product managers' role effectively, they must have technical expertise, market understanding, and stakeholder management skills. However, they lack sufficient organizational support to enrich their skills and knowledge. To improve decision-making and ensure better alignment of SPM with business objectives, organizations have to enhance product managers' capabilities by leveraging emerging technologies.

A key to enhancing SPM practices is realigning organizational attitudes, acquiring the necessary expertise, and using new tools, technologies, and techniques. By investing in the knowledge, resources, and organizational frameworks that enable product managers, SPM's influence can be further increased, and their potential as a source of innovation and value can be realized. Encouraging an organizational culture that sees SPM as an important factor in company performance rather than merely an expense can support aligning SPM well with the software lifecycle.

There are several implications for practice suggested by this study. One is realigning the organizational culture to better recognize the strategic value that SPM provides to software organization as well as to client organizations. The second is promoting the alignment between SPM and SE practices among the software practitioners. Designing practical and scalable SPM frameworks and ontologies and leveraging the alignment between SPM and emerging technologies will support such coordination among SPM and SE practices. Moreover, organizational readiness can be encouraged to use enhanced SPM practices in the software development lifecycle.

Future research should focus on (1) empirically defining and developing adaptive SPM frameworks aligned with diverse software engineering paradigms (e.g., Agile, DevOps); (2) developing role-based competency models and training curricula to improve product managers' technical and strategic skills; (3) studying the application of AI and data analytics tools towards supporting product decision-making and prioritization; (4) studying organizational change strategies to promote SPM maturity in small and medium-sized businesses; and (5) evolving more detailed SPM domain ontologies to better facilitate knowledge sharing, process automation, and decision support systems.

**Author Contributions:** *Chalani Oruthotaarachchi*: Conceptualization, Methodology, Writing - Original Draft, Writing - Review & Editing. *Janaka Wijayanayake:* Conceptualization, Supervision, Review & Editing.

All authors have read and agreed to the published version of the manuscript.

**Funding:** This research received no specific grant from any funding agency

**Conflicts of Interest:** The authors declare no conflict of interest.

**Data Availability:** Data sharing does not apply to this article as no new data were created or analyzed in this study.

**Informed Consent:** There were no human subjects.

**Institutional Review Board Statement:** Not applicable.

**Animal Subjects:** There were no animal subjects.

**ORCID**:
Chalani Oruthotaarachchi: https://orcid.org/0000-0003-1594-3688
Janaka Wijayanayake: https://orcid.org/0000-0002-9523-5384

## REFERENCES

[1] L. Gorchels, *The Product Manager's Handbook : The Complete Product Management Resource*. McGraw-Hill, 2022.

[2] O. Springer and J. Miler, "The Role of a Software Product Manager in Various Business Environments," in *2018 Federated Conference on Computer Science and Information Systems (FedCSIS)*, 2018, pp. 985–994.

[3] A. Maglyas, U. Nikula, K. Smolander, and S. A. Fricker, "Core software product management activities," *Journal of Advances in Management Research*, vol. 14, no. 1, pp. 23–45, Feb. 2017, doi: 10.1108/JAMR-03-2016-0022.

[4] S. Hyrynsalmi, A. Suominen, and M. Seppanen, "A Bibliographical Study of Software Product Management Research," in *2021 IEEE International Conference on Engineering, Technology and Innovation (ICE/ITMC)*, IEEE, Jun. 2021, pp. 1–8. doi: 10.1109/ICE/ITMC52061.2021.9570214.

[5] H.-B. Kittlaus, *Software Product Management: The ISPMA®-Compliant Study Guide and Handbook, 2nd Edition*. 2022. doi: 10.1007/978-3-662-65116-2.

[6] L. Gorchels, *The Product Manager's Handbook 4/E, 4th Edition*. McGraw-Hill, 2011.

[7] A. Maglyas, U. Nikula, and K. Smolander, "What are the roles of software product managers? An empirical investigation," *Journal of Systems and Software*, vol. 86, no. 12, pp. 3071–3090, Dec. 2013, doi: 10.1016/j.jss.2013.07.045.

[8] S. J. Konig, "Finance as a Stakeholder in Product Management," in *2009 Third International Workshop on Software Product Management*, IEEE, Sep. 2009, pp. 15–22. doi: 10.1109/IWSPM.2009.2.

[9] A. Tkalich, R. Ulfsnes, and N. Moe, "Toward an Agile Product Management: What Do Product Managers Do in Agile Companies?," 2022, pp. 168–184. doi: 10.1007/978-3-031-08169-9_11.

[10] O. Springer and J. Miler, "A comprehensive overview of software product management challenges," *Empir Softw Eng*, vol. 27, no. 5, p. 106, Sep. 2022, doi: 10.1007/s10664-022-10134-5.

[11] N. Moe, M. Berntzen, A. Barbala, and V. Stray, "Software Product Management in Large-Scale Agile," 2024, pp. 53–69. doi: 10.1007/978-3-031-61154-4_4.

[12] A. Tkalich, R. Ulfsnes, and N. B. Moe, "Toward an Agile Product Management: What Do Product Managers Do in Agile Companies?," in *Lecture Notes in Business Information Processing*, Springer Science and Business Media Deutschland GmbH, 2022, pp. 168–184. doi: 10.1007/978-3-031-08169-9_11.

[13] W.-T. Lee and C.-H. Chen, "Agile Software Development and Reuse Approach with Scrum and Software Product Line Engineering," *Electronics (Basel)*, vol. 12, no. 15, p. 3291, Jul. 2023, doi: 10.3390/electronics12153291.

[14] A. Fawzy, A. Tahir, M. Galster, and P. Liang, "Exploring data management challenges and solutions in agile software development: a literature review and practitioner survey," *Empir Softw Eng*, vol. 30, no. 3, p. 77, May 2025, doi: 10.1007/s10664-025-10630-4.

[15] S. A. Licorish, "Understanding the Effect of Agile Practice Quality on Software Product Quality," *IEEE Transactions on Software Engineering*, vol. 51, no. 2, pp. 650–662, Feb. 2025, doi: 10.1109/TSE.2025.3532502.

[16] N. Parikh, "Empowering Business Transformation: The Positive Impact and Ethical Considerations of Generative AI in Software Product Management – A Systematic Literature Review," Jul. 2023. doi: 10.48550/arXiv.2306.04605.

[17] F. Pattyn, "The Critical Role of Product Managers and Their Responsibilities in Software Startups: A Systematic Literature Review," *American Journal of Engineering and Technology Management*, vol. 9, no. 4, pp. 66–91, Oct. 2024, doi: 10.11648/j.ajetm.20240904.11.

[18] S. Demirel, "Project Management vs Product Management: A View on The Journey from Project to Product in Software Development," Jul. 2022.

[19] O. Springer, J. Miler, and Michałr. Wróbel, "Strategies for Dealing With Software Product Management Challenges," *IEEE Access*, vol. 11, pp. 55797–55813, 2023, doi: 10.1109/ACCESS.2023.3282605.

[20] C. Ebert and S. Brinkkemper, "Software product management – An industry evaluation," *Journal of Systems and Software*, vol. 95, pp. 10–18, Sep. 2014, doi: 10.1016/j.jss.2013.12.042.

[21] I. Kolawole and A. Fakokunde, "Improving Software Development with Continuous Integration and Deployment for Agile DevOps in Engineering Practices," *International Journal of Computer Applications Technology and Research*, Jul. 2024, doi: 10.7753/IJCATR1401.1002.

[22] H.-B. Kittlaus and S. Fricker, "Management of Software as a Business," in *Software Product Management: The ISPMA-Compliant Study Guide and Handbook*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2017. doi: 10.1007/978-3-642-55140-6.

[23] I. Jacobson, P.-W. Ng, P. E. McMahon, I. Spence, and S. Lidman, "The essence of software engineering," *Commun ACM*, vol. 55, no. 12, pp. 42–49, Dec. 2012, doi: 10.1145/2380656.2380670.

[24] P. Grasserbauer and R. Ploesch, "Value Based Prioritization of Requirements in Software Engineering Education," in *2023 IEEE 35th International Conference on Software Engineering Education and Training (CSEE&T)*, IEEE, Aug. 2023, pp. 11–20. doi: 10.1109/CSEET58097.2023.00012.

[25] B. De Haaff, "What Is a Minimum Lovable Product? (Plus, MLP Vs. MVP)," https://www.aha.io/roadmapping/guide/plans/what-is-a-minimum-lovable-product?term=lovability.

[26] H. Cao and P. Folan, "Product life cycle: the evolution of a paradigm and literature review from 1950–2009," *Production Planning & Control*, vol. 23, no. 8, pp. 641–662, Aug. 2012, doi: 10.1080/09537287.2011.577460.

[27] V. Krishnan and K. Ulrich, "Product Development Decisions: A Review of the Literature," *Manage Sci*, vol. 47, pp. 1–21, Jul. 2001, doi: 10.1287/mnsc.47.1.1.10668.

[28] H. Gunjal and R. Belokar, "Systematic Review: Implementation of Product Lifecycle Management in Industries," 2023, pp. 263–279. doi: 10.1007/978-981-19-6107-6_19.

[29] J. Münch, S. Trieflinger, and D. Lang, "Product Roadmap - From Vision to Reality: A Systematic Literature Review," Jul. 2019. doi: 10.1109/ICE.2019.8792654.

[30] N. A. Parikh, "A Novel Software Product Manager's Framework: A Systematic Literature Review," *Open Journal of Business and Management*, vol. 12, no. 01, pp. 634–666, 2024, doi: 10.4236/ojbm.2024.121036.

[31] A. Carrera-Rivera, W. Ochoa, F. Larrinaga, and G. Lasa, "How-to conduct a systematic literature review: A quick guide for computer science research," *MethodsX*, vol. 9, p. 101895, 2022, doi: 10.1016/j.mex.2022.101895.

[32] N. R. Haddaway, M. J. Page, C. C. Pritchard, and L. A. McGuinness, "PRISMA2020: An R package and Shiny app for producing PRISMA 2020-compliant flow diagrams, with interactivity for optimised digital transparency and Open Synthesis," *Campbell Systematic Reviews*, vol. 18, no. 2, Jun. 2022, doi: 10.1002/cl2.1230.

[33] T. Williams, "Three Signs Showing You Engineering Organizations and Product Management Team Are Out of Sync and How to Remediate?," https://www.keypup.io/blog/out-of-sync-engineering-organization-and-product-management.

[34] I. de Weerd and R. Katchow, "On the integration of software product management with software defect management in distributed environments," in *2009 5th Central and Eastern European Software Engineering Conference in Russia, CEE-SECR 2009*, Jul. 2009, pp. 167–172. doi: 10.1109/CEE-SECR.2009.5501167.

[35] I. Van De Weerd, S. Brinkkemper, and J. Versendaal, "Incremental method evolution in global software product management: A retrospective case study," *Inf Softw Technol*, vol. 52, pp. 720–732, 2010, doi: 10.1016/j.infsof.2010.03.002.

[36] I. de Weerd, S. Brinkkemper, and J. Versendaal, "Concepts for Incremental Method Evolution: Empirical Exploration and Validation in Requirements Management.," Jul. 2007, pp. 469–484.

[37] H. Alahyari, R. Berntsson Svensson, and T. Gorschek, "A study of value in agile software development organizations," *Journal of Systems and Software*, vol. 125, pp. 271–288, Mar. 2017, doi: 10.1016/j.jss.2016.12.007.

[38] K. Vlaanderen, S. Jansen, S. Brinkkemper, and E. Jaspers, "The Agile Requirements Refinery: Applying SCRUM Principles to Software Product Management," *Inf Softw Technol*, vol. 53, pp. 58–70, Jul. 2009, doi: 10.1016/j.infsof.2010.08.004.

[39] H.-B. Kittlaus, "Software Product Management and Agile Software Development: Conflicts and Solutions," 2012, pp. 83–96. doi: 10.1007/978-3-642-31371-4_5.

[40] G. Lucassen, J. M. E. M. van der Werf, and S. Brinkkemper, "Alignment of software product management and software architecture with discussion models," in *2014 IEEE 8th International Workshop on Software Product Management (IWSPM)*, IEEE, Aug. 2014, pp. 21–30. doi: 10.1109/IWSPM.2014.6891064.

[41] D. Condon, *Software Product Management: Managing Software Development from Idea to Product to Marketing to Sales*. Aspatore Books, 2002.

[42] H.-B. Kittlaus and H. Mangipudi, *Software Product Management for Startups – The ISPMA-Compliant Study Guide and Handbook*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2023. doi: 10.1007/978-3-662-65116-2.

[43] V. Bureš, "A Process View on Product Management in Software Engineering Companies," *Journal of Software & Systems Development*, pp. 1–12, Jun. 2012, doi: 10.5171/2012.898139.

[44] M. U. Tariq, "User Centered Human-Computer Interaction and Agile Development: A Systematic Model for Useable Product Case Study," Jul. 2020.

[45] H. Kittlaus, *One Size Does Not Fit All: Software Product Management For Speedboats vs. Cruiseships*. Switzerland: Springer International Publishing, 2015.

[46] G. Geracie and S. D. Eppinger, "The Guide to the Product Management and Marketing Body of Knowledge," 2013.

[47] K. Ali, J. A. Khan, F. Aizaz, and M. Ahmed, "Software Requirements Prioritization in the context of Global Software Development," in *2021 International Conference on Frontiers of Information Technology (FIT)*, IEEE, Dec. 2021, pp. 13–18. doi: 10.1109/FIT53504.2021.00013.

[48] B. Fitzgerald and K.-J. Stol, "Continuous software engineering: A roadmap and agenda," *Journal of Systems and Software*, vol. 123, pp. 176–189, Jan. 2017, doi: 10.1016/j.jss.2015.06.063.

[49] M. Cagan, *Inspired: How to Create Products Customers Love*. 2008.

[50] C. Ebert, "The impacts of software product management," *Journal of Systems and Software*, vol. 80, no. 6, pp. 850–861, Jun. 2007, doi: 10.1016/j.jss.2006.09.017.

[51] S. A. Fricker, "Software Product Management," in *Software for People. Management for Professionals*, Berlin, Heidelberg: Springer, 2012, pp. 53–81. doi: 10.1007/978-3-642-31371-4_4.

[52] C. Ebert, "Managing software products in a global context," in *Proceedings of the 13th International Conference on Global Software Engineering*, New York, NY, USA: ACM, May 2018, pp. 69–76. doi: 10.1145/3196369.3196371.

[53] S. Haines, *The Product Manager's Desk Reference*, 3rd ed. McGraw-Hill Education, 2021.

[54] T. Tomaszewski, "Product Manager vs. Project Manager." Accessed: Nov. 17, 2023. [Online]. Available: https://productvision.pl/2013/product-manager-vs-project-manager/

[55] M. C. Paulk, B. Curtis, M. B. Chrissis, and C. V. Weber, "Capability maturity model, version 1.1," *IEEE Softw*, vol. 10, no. 4, pp. 18–27, Jul. 1993, doi: 10.1109/52.219617.

[56] C. P. Team, *Capability Maturity Model{\textregistered} Integration (CMMI\textsuperscript{SM}), Version 1.1 Continuous Representation (CMMI-SE/SW, V1.1, Continuous)*. 2002.

[57] M. A. Cusumano, *The Business of Software: What Every Manager, Programmer, and Entrepreneur Must Know to Thrive and Survive in Good Times and Bad*. New York: Free Press, 2004.

[58] B. K. Clark, "Quantifying the effects of process improvement on effort," *IEEE Softw*, vol. 17, no. 6, pp. 65–70, 2000, doi: 10.1109/52.895170.

[59] jean-pierre Kuilboer and N. Ashrafi, "Software process and product improvement: An empirical assessment," *Inf Softw Technol*, vol. 42, pp. 27–34, Jul. 2000, doi: 10.1016/S0950-5849(99)00054-3.

[60] D. J. Reifer, "The CMMI: it's formidable," *Journal of Systems and Software*, vol. 50, no. 2, pp. 97–98, Feb. 2000, doi: 10.1016/S0164-1212(99)00119-3.

[61] J. G. Brodman and D. L. Johnson, "What small businesses and small organizations say about the CMM," in *Proceedings of 16th International Conference on Software Engineering*, IEEE Comput. Soc. Press, 1994, pp. 331–340. doi: 10.1109/ICSE.1994.296795.

[62] S. Brinkkemper, R. Nieuwenhuis, L. Bijlsma, J. Versendaal, and I. van de Weerd, " Towards a Reference Framework for Software Product Management ," in *14th IEEE International Requirements Engineering Conference* , Los Alamitos, CA, USA: IEEE Computer Society, Sep. 2006, pp. 319–322. doi: 10.1109/RE.2006.66.

[63] P. Institute, "Pragmatic Marketing Framework." Accessed: Dec. 12, 2023. [Online]. Available: https://www.pragmaticinstitute.com/product/framework/

[64] I. Van De Weerd, J. Versendaal, and S. Brinkkemper, "A Product Software Knowledge Infrastructure for Situational Capability Maturation: Vision and Case Studies in Product Management," in *12th Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ'06)*, 2006, pp. 87–112.

[65] K. Vlaanderen, I. van de Weerd, and S. Brinkkemper, "Model-driven assessment in Software Product Management," in *2010 Fourth International Workshop on Software Product Management*, IEEE, Sep. 2010, pp. 17–25. doi: 10.1109/IWSPM.2010.5623868.

[66] W. Bekkers, I. van de Weerd, M. Spruit, and S. Brinkkemper, "A Framework for Process Improvement in Software Product Management," in *Systems, Software and Services Process Improvement*, A. O. R. T. S. M. R. Riel, Ed., Berlin, Heidelberg: Springer, 2010, pp. 1–12.

[67] W. B. S. V. D. B. L. F. and W. C. and V. D. W. I. Bekkers, "Evaluating the Software Product Management Maturity Matrix," in *2012 20th IEEE International Requirements Engineering Conference (RE)*, IEEE, 2012, pp. 51–60.

[68] A. Botzenhardt and A. Maedche, "Towards a performance measurement reference model for software product management," in *2010 Fourth International Workshop on Software Product Management*, IEEE, Sep. 2010, pp. 26–29. doi: 10.1109/IWSPM.2010.5623869.

[69] A. Botzenhardt, A. Maedche, and J. Wiesner, "Developing a domain ontology for software product management," in *2011 Fifth International Workshop on Software Product Management (IWSPM)*, IEEE, Aug. 2011, pp. 7–16. doi: 10.1109/IWSPM.2011.6046207.

[70] An Ngo-The and G. Ruhe, "Optimized Resource Allocation for Software Release Planning," *IEEE Transactions on Software Engineering*, vol. 35, no. 1, pp. 109–123, Jan. 2009, doi: 10.1109/TSE.2008.80.

[71] T. AlBourae, G. Ruhe, and M. Moussavi, "Lightweight Replanning of Software Product Releases," in *2006 International Workshop on Software Product Management (IWSPM'06 - RE'06 Workshop)*, IEEE, 2006, pp. 27–34. doi: 10.1109/IWSPM.2006.5.

[72] C. Li, M. van den Akker, S. Brinkkemper, and G. Diepen, "An integrated approach for requirement selection and scheduling in software release planning," *Requir Eng*, vol. 15, no. 4, pp. 375–396, Nov. 2010, doi: 10.1007/s00766-010-0104-x.

[73] I. and B. S. Bebensee Thomas and van de Weerd, "Binary Priority List for Prioritizing Software Requirements," in *Requirements Engineering: Foundation for Software Quality*, A. Wieringa Roel and Persson, Ed., Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 67–78.

[74] B. Regnell, M. Höst, J. N. och Dag, P. Beremark, and T. Hjelm, "An Industrial Case Study on Distributed Prioritisation in Market-Driven Requirements Engineering for Packaged Software," *Requir Eng*, vol. 6, no. 1, pp. 51–62, Feb. 2001, doi: 10.1007/s007660170015.

[75] P. Berander and A. Andrews, "Requirements Prioritization," in *Engineering and Managing Software Requirements*, Berlin/Heidelberg: Springer-Verlag, 2005, pp. 69–94. doi: 10.1007/3-540-28244-0_4.

[76] J. Karlsson, "Software requirements prioritizing," in *Proceedings of the Second International Conference on Requirements Engineering*, IEEE Comput. Soc. Press, 1996, pp. 110–116. doi: 10.1109/ICRE.1996.491435.

[77] B. Regnell and K. Kuchcinski, "Exploring Software Product Management Decision Problems with Constraint Solving—Opportunities for Prioritization and Release Planning," in *2011 Fifth International Workshop on Software Product Management (IWSPM)*, IEEE, 2011, pp. 47–56.

[78] A. Saltan, S. Jansen, and K. Smolander, "Decision-making in Software Product Management: Identifying Research Directions from Practice," Jul. 2018.

[79] E. Mendes, P. Rodriguez, V. Freitas, S. Baker, and M. A. Atoui, "Towards improving decision making and estimating the value of decisions in value-based software engineering: the VALUE framework," *Software Quality Journal*, vol. 26, no. 2, pp. 607–656, Jun. 2018, doi: 10.1007/s11219-017-9360-z.

[80] R. B. Briner, D. Denyer, and D. M. Rousseau, "Evidence-Based Management: Concept Cleanup Time?," *Academy of Management Perspectives*, vol. 23, no. 4, pp. 19–32, Nov. 2009, doi: 10.5465/AMP.2009.45590138.

[81] D. M. Rousseau, "Is there Such a thing as 'Evidence-Based Management'?," *Academy of Management Review*, vol. 31, no. 2, pp. 256–269, Apr. 2006, doi: 10.5465/amr.2006.20208679.

[82] R. P. L. Buse and T. Zimmermann, "Information needs for software development analytics," in *2012 34th International Conference on Software Engineering (ICSE)*, IEEE, Jun. 2012, pp. 987–996. doi: 10.1109/ICSE.2012.6227122.

[83] C. Lin, "Data driven product management," *IEEE Engineering Management Review*, vol. 46, no. 1, pp. 16–18, Mar. 2018, doi: 10.1109/EMR.2018.2810099.

[84] Y. Liu, Y. Zhang, S. Ren, M. Yang, Y. Wang, and D. Huisingh, "How can smart technologies contribute to sustainable product lifecycle management?," *J Clean Prod*, vol. 249, p. 119423, Mar. 2020, doi: 10.1016/j.jclepro.2019.119423.

[85] A. Maglyas, U. Nikula, and K. Smolander, "Lean Solutions to Software Product Management Problems," *IEEE Softw*, vol. 29, no. 5, pp. 40–46, Sep. 2012, doi: 10.1109/MS.2012.108.

[86] J. Womack and D. Jones, *Lean Thinking : Banish Waste and Create Wealth in Your Corporation*, vol. 48. 1996. doi: 10.1038/sj.jors.2600967.

[87] W. Bekkers, I. van de Weerd, S. Brinkkemper, and A. Mahieu, "The Influence of Situational Factors in Software Product Management: An Empirical Study," in *2008 Second International Workshop on Software Product Management*, IEEE, Sep. 2008, pp. 41–48. doi: 10.1109/IWSPM.2008.8.

[88] K. Wnuk, B. Regnell, and L. Karlsson, "What Happened to Our Features? Visualization and Understanding of Scope Change Dynamics in a Large-Scale Industrial Setting," in *2009 17th IEEE International Requirements Engineering Conference*, IEEE, Aug. 2009, pp. 89–98. doi: 10.1109/RE.2009.32.

[89] E. Bjarnason, K. Wnuk, and B. Regnell, "Overscoping: Reasons and consequences &#x2014; A case study on decision making in software product management," in *2010 Fourth International Workshop on Software Product Management*, IEEE, Sep. 2010, pp. 30–39. doi: 10.1109/IWSPM.2010.5623866.

[90] A. Maglyas, U. Nikula, and K. Smolander, "Software product management in the Russian companies," in *2011 7th Central and Eastern European Software Engineering Conference (CEE-SECR)*, IEEE, Oct. 2011, pp. 1–9. doi: 10.1109/CEE-SECR.2011.6188469.

[91] S. Jantunen and D. C. Gause, "Using a grounded theory approach for exploring software product management challenges," *Journal of Systems and Software*, vol. 95, pp. 32–51, Sep. 2014, doi: 10.1016/j.jss.2014.03.050.

**Publisher's Note:** Publisher stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.