systems

MDPI

Review

Uncertainty in Software Development Projects: A Review of Causes, Types, Challenges, and Future Research Directions

Mingqi Zhang [1], Maxwell Fordjour Antwi-Afari [2], Chonghui Wang [2], Weihao Sun [3,*], Saeed Reza Mohandes [1] and Sulemana Fatoama Abdulai [4]

[1] Department of Civil Engineering and Management, School of Engineering, University of Manchester, Manchester M13 9PL, UK; mingqi.zhang@postgrad.manchester.ac.uk (M.Z.); saeedreza.mohandes@manchester.ac.uk (S.R.M.)
[2] Department of Civil Engineering, College of Engineering and Physical Sciences, Aston University, Birmingham B4 7ET, UK; m.antwiafari@aston.ac.uk (M.F.A.-A.); c.wang21@aston.ac.uk (C.W.)
[3] Department of Construction and Real Estate, School of Civil Engineering, Southeast University, Nanjing 211189, China
[4] Department of Building and Real Estate, The Hong Kong Polytechnic University, Hung Hom, Hong Kong SAR, China; sulemana.abdulai@connect.polyu.hk
* Correspondence: weihaosun2023@seu.edu.cn

Abstract

In a rapidly evolving business landscape, the success of software development (SD) projects is increasingly impacted by uncertainty, which poses significant challenges for project managers. Despite the known influence of uncertainty on project outcomes, its types, causes, and challenges in software remain inadequately understood. This review conducts a systematic analysis of previous related SD projects and related research to clarify these aspects, ultimately identifying key research gaps and proposing future research directions. By adopting a mixed-methods review that integrates scientometric analysis and systematic review methods, this study analysed 60 articles from the Scopus database. The results reported nine causes, six types, and nine challenges associated with uncertainty in SD to provide insights for project managers and researchers in understanding and managing uncertainty more effectively. Additionally, this study proposes four areas for further research to enhance focus and innovation in SD project management.

Keywords: SD projects; uncertainty analysis; systematic review; scientometric review; project management

check for updates

Academic Editors: Fahim Ullah, Ahsen Maqsoom, Khurram Iqbal Ahmad Khan and Amir Ghanbaripour

Received: 10 June 2025
Revised: 23 July 2025
Accepted: 24 July 2025
Published: 1 August 2025

Citation: Zhang, M.; Antwi-Afari, M.F.; Wang, C.; Sun, W.; Mohandes, S.R.; Abdulai, S.F. Uncertainty in Software Development Projects: A Review of Causes, Types, Challenges, and Future Research Directions. *Systems* **2025**, *13*, 650. https://doi.org/10.3390/systems13080650

1. Introduction

In today's business landscape, rapidly advancing technologies have shortened product and service life cycles, while a constantly evolving market and increasing security threats have intensified competition [1]. Project management has become central to business process management [2], with strong project management capabilities now seen as critical for building competitive advantages [3]. In such an environment, companies must continually enhance their ability to innovate and adapt, keeping pace with a shifting business context by strengthening project management practices. Despite theoretical and practical advancements, however, projects frequently face an array of challenges, including time and cost overruns, unmet business objectives, stakeholder dissatisfaction, and sustainability issues [4]. Since project management was first applied in industries such as defence, aerospace, construction, and pharmaceuticals in the 1970s, scholars and practitioners have examined how to manage projects effectively from various perspectives [5].

A central focus in project management has long been the successful completion of tasks within specified time, budget, and quality constraints—often termed the "iron triangle" [6]. Extensive studies have highlighted reasons for project failure. For example, according to Ika and Pinto [7], there are no clear guidelines for evaluating success, and the meaning of the organisation's strategic objectives is not widely spread and has not reached a consensus within and outside the organisation. Furthermore, there is no clear distinction between project success and subsequent earnings, and definitions of success are rarely updated to reflect current realities [7]. Achieving success and satisfying stakeholders has thus become a primary research focus within project management [6]. According to relevant studies, approximately 50% of software projects fail to meet the satisfaction of CEOs and are therefore deemed unsuccessful [8]. Although numerous studies have investigated the factors contributing to project success [9], the issue of project success remains significant. Experts and scholars have called for further research in this field [10]. However, uncertainty and risk are frequently overlooked by project managers and practitioners, often leading to project failure [11].

Uncertainty arises from the very inception of a project and persists until its completion, particularly when there is significant ambiguity surrounding the scope and delivery mechanisms [12]. Ref. [13] defines uncertainty as the discrepancy between the data wanted and the data actually produced. Uncertainty is an inherent characteristic of projects, manifesting within technological, organisational, and social contexts [14]. Uncertainty is negatively correlated with 24 July 2025 project success [15], meaning that higher levels of uncertainty significantly impact success and often lead to project failure. When SD reaches a certain level of complexity and uncertainty, the use of various techniques, practices, and project management tools is prevalent in software engineering. Yet, despite the development of various tools aimed at managing complex projects, numerous instances of deviations from project plans and project failures persist. For instance, in a study conducted by scholars [14] surveying 5400 software projects, findings revealed that over 50% of projects exceeded their budgets, more than half of the projects reduced their scope, and nearly 20% of projects failed to achieve the desired outcomes.

Uncertainty is often conflated with risk management, with some even regarding them as synonymous [11]. However, a distinction exists: uncertainty is generally more challenging to analyse and control than risk. Uncertainty represents what remains after risk has been identified; though it poses a potential threat, no action can be taken until it materialises—otherwise, it can be regarded as a risk [16]. As described in the PMBOK, uncertainty affects project activities, leading to various outcomes [17]. A simplified understanding would posit that uncertainty can only be termed as risk when it materialises into adverse consequences.

Scholars have proposed different approaches to addressing uncertainty in software projects from diverse perspectives. According to [18], uncertainty in software projects can be tackled through a four-step process: constraint, modification, classification, and resolution. However, the effectiveness of this approach is uncertain, as it lacks empirical validation. Although [11] developed a method for categorising, identifying, perceiving, and managing uncertainties in software projects and conducting semi-structured interviews, their process is complex and lacks internal consistency. In software projects, sources of uncertainty are diverse, which has led scholars to investigate both internal and external organisational factors [19] or to focus on specific aspects [20]. Meanwhile, risk management is a scientific process to manage all the unexpected results during the development of new software. Several review studies have systematically examined risk management practices across SD methodologies. Ref. [21] extracted and classified 148 risk factors according to the SD taxonomy developed by the Software Engineering Institute. More recently, ref. [22]

conducted a systematic literature review of risk management in SD projects, synthesising key practices, tools, and research gaps. These reviews reinforce the close relationship between risk and uncertainty in the software domain. Although risk and uncertainty are conceptually distinct—risk involving known probabilities and uncertainty involving unknowns—many scholars and practitioners even use the terms interchangeably. However, there is a noticeable gap in the literature regarding review studies that specifically explore the uncertainty in the context of SD projects.

Herein, there is an urgent need for a framework to enhance the overall success of software projects by effectively managing the uncertainty that may arise throughout the project lifecycle. Before doing this, foundational work is required to investigate the source, types, and challenges of uncertainty in software projects and their impact on project outcomes. Therefore, this review study aims to conduct a systematic literature review and scientometric analyses to discuss the causes, types, challenges, and future research directions of uncertainty in SD projects. The findings are intended to provide software project managers with a comprehensive understanding of uncertainty, ultimately improving project success rates. To this end, this study refines its research questions as follows: (1) What are the current causes and types of uncertainty in SD projects? (2) What kinds of uncertainty challenges are critically examined to facilitate the reduction in uncertainty in the SD projects? (3) What are the research gaps and potential future directions in uncertainty research for SD projects?

This review study adopts a systematic literature review method by following the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) guidelines, combined with scientometric analysis, to investigate the causes, types, challenges, and future research direction associated with uncertainty in SD projects. A systematic literature review is a suitable method to understand the current research status and future research direction within a given field [23]. It is a structured, reproducible step to search, classify, organise, and synthesise results from the existing literature. Furthermore, systematic literature reviews can effectively identify future research trends and synthesise a large body of research findings [24]. The systematic literature review method can scientifically select the literature according to certain steps [25], rather than allowing researchers to pick relevant articles based on their preferences [26], which is the traditional literature review method. Ref. [27] posited that conducting a literature review is like constructing a factory: it involves clearing away what is unnecessary and arranging insights from similar studies to build knowledge. So, in order to know the causes, types, and challenges in the SD projects, this study adopts a systematic literature review. The research objectives are hence as follows: (1) To explore keywords, publication trends, and countries contributing to uncertainty research in the field of SD projects. (2) To identify and analyse the current causes and types of uncertainty in SD projects. (3) To examine and critically assess the challenges posed by uncertainty in the software industry to facilitate its reduction. (4) To identify research gaps and suggest potential future research directions related to uncertainty in SD projects.

Following the introduction, this review study is structured as follows. The second section outlines the research methodology, introducing the rationale behind the mixed review methods in this review study and detailing each procedural step. The third part presents the main findings, including the main results of the studies reviewed. The fourth section is the discussion, which critically examines the main findings, summarises the types, causes, and challenges of uncertainty in SD projects. The final part draws conclusions based on the main content of findings, identifying research gaps, and suggesting directions for future research on uncertainty in software projects.

2. Research Methodology

This paper adopts a combination of a systematic literature review (SLR) and the scientometric analysis method. The systematic literature review is followed by the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA). Scientometric analysis uses the bibliometric data to make a figure illustrating the relationship between the different papers with the VOSviewer 1.6.18. By mixing the results of SLR and scientometric analysis, this review paper will discuss the causes, types, challenges, and future research direction in the SD projects. The main methodological steps are illustrated in Figure 1.

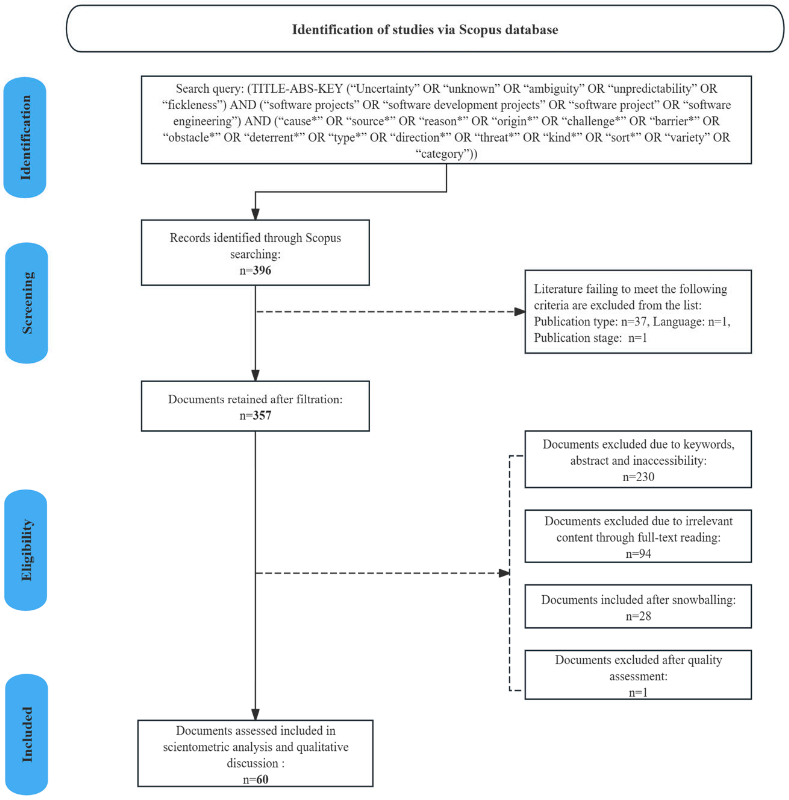
Identification of studies via Scopus database
Identification
Search query: (TITLE-ABS-KEY ("Uncertainty" OR "unknown" OR "ambiguity" OR "unpredictability" OR "fickleness") AND ("software projects" OR "software development projects" OR "software project" OR "software engineering") AND ("cause*" OR "source*" OR "reason*" OR "origin*" OR "challenge*" OR "barrier*" OR "obstacle*" OR "deterrent*" OR "type*" OR "direction*" OR "threat*" OR "kind*" OR "sort*" OR "variety" OR "category"))
Screening
Records identified through Scopus searching: n=396
Literature failing to meet the following criteria are excluded from the list: Publication type: n=37, Language: n=1, Publication stage: n=1
Documents retained after filtration: n=357
Eligibility
Documents excluded due to keywords, abstract and inaccessibility: n=230
Documents excluded due to irrelevant content through full-text reading: n=94
Documents included after snowballing: n=28
Documents excluded after quality assessment: n=1
Included
Documents assessed included in scientometric analysis and qualitative discussion : n=60

Figure 1. The mixed review framework follows the PRISMA protocol.

2.1. The Rationale for the Mixed Review Method

In Section 2, after reviewing the definition of uncertainty, as well as a detailed exploration of uncertainty in SD projects and prior articles, this review study adopts a mixed review methodology based on the approach of [28]. This approach combines scientometric analysis and systematic review methods to address the research questions of this study.

As discussed in the previous section, according to [23], systematic literature reviews as a qualitative analysis method can help researchers understand the current state of research in a specific field [29]. A systematic literature review follows the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) method to search, select, and synthesise studies [30]. Scientometric analysis is a relatively objective method that can present unbiased results through scientific mapping [31]. This method of extensively visualising and analysing the literature can significantly aid researchers in gaining insights into the knowledge and future trends within a research field [32]. Adopting a systematic literature review combined with a scientometric analysis method integrates knowledge from the existing literature in a descriptive manner [33]. It enables a deeper understanding of the research topic, leveraging the advantages of various methods while avoiding the limitations of a single approach [34,35]. Alternative methods were excluded due to time constraints, ethical concerns with human subjects, and limited data availability. For instance, case studies require detailed data and strict protocols, which are unsuitable for capturing the broad uncertainties in SD projects.

2.2. Systematic Literature Review

The systematic literature review comprises four main sections including identification, screening, eligibility, and included articles and quality assessments. These steps are explained in more detail in the following subsections.

2.2.1. Identification

Before conducting a formal search for articles closely related to the research topic, it is essential to select an appropriate search engine or database to identify the relevant papers, as this choice is crucial for the research process [36].

Commonly used databases include Web of Science, Google Scholar, and Scopus. This review study proposes to use the Scopus database as the primary database for the search, for several reasons. Firstly, Scopus is a leading global academic database with a comprehensive data source [37]. Additionally, its extensive database content covers most of the material available in other databases [38]. Scopus is one of the largest online literature databases and has a significant influence, offering access to the most recent publications, making it suitable for scientometric analysis [39]. On the other hand, other databases like Google Scholar lack a bulk export function, which makes the subsequent extensive article search and selection process time-consuming and less efficient. Therefore, Scopus is chosen for article retrieval as it encompasses the most extensive collection of peer-reviewed journals [38].

Following the PRISMA step 1, the initial search in Scopus employed the following keywords: (“Uncertainty” OR “unknown” OR “ambiguity” OR “unpredictability” OR “fickleness”) AND (“software projects” OR “software development projects” OR “software project” OR “software engineering”) AND (“cause*” OR “source*” OR “reason*” OR “origin*” OR “challenge*” OR “barrier*” OR “obstacle*” OR “deterrent*” OR “type*” OR “direction*” OR “threat*” OR “kind*” OR “sort*” OR “variety” OR “category”). Any papers with titles, abstracts, or keywords matching these criteria were advanced to the next screening stage. To ensure a substantial data volume in the initial search, synonyms are used as comprehensively as possible. The complexity in selecting keywords arises because some of the literature confuses these concepts. To better research the content, these keywords are included, and they will be distinguished during detailed reading at a later stage. Following this procedure, 396 documents were retrieved from Scopus.

2.2.2. Screening

According to PRISMA Step 2, the next phase after identifying the literature is screening, which involves the initial filtering of results obtained from the first step. The main content is to look through the journal type, language, and publication status to determine whether the articles should be included or not.

The screening of the 396 articles identified from Scopus was carried out based on the following criteria: (1) Only journal articles, and conference papers excluding books and review articles; (2) The language of the articles had to be English, as it is the primary language of our research; (3) Articles had to be in their final publication stage to avoid issues with pre-published articles being retracted. Following these criteria, 39 articles were excluded, resulting in 357 articles proceeding to the next phase.

2.2.3. Eligibility

According to PRISMA Step 3, the subsequent phase is the eligibility check, which ensures that the selected articles are highly relevant to the research question.

Since the abstract and title of an article reflect the main content of the publication [40], the eligibility check critically involves assessing the relevance of the articles to the research questions to ensure their credibility and reproducibility. In the first step of the eligibility check, the titles and abstracts of the 357 articles were reviewed, and articles that were not closely related to the research questions on uncertainty issues in SD projects were removed. This step eliminated 200 articles, leaving 157 articles to proceed to the next phase. Before carefully reading the full text, these 157 articles need to be downloaded first, but 30 were either non-public or unavailable, resulting in their exclusion. In the second step of the eligibility check, a full-text review of the remaining articles was conducted to ensure their relevance to the research questions of this review study. After a thorough review, 94 articles were found to be irrelevant and were excluded, leaving 33 articles for the next stage. To ensure comprehensive coverage of relevant studies, a manual snowballing approach was adopted to continuously search for the relevant literature by checking and recording references and citation searches or similar work [28]. This process added 28 relevant studies, resulting in 61 articles proceeding to the next step.

2.2.4. Included Articles and Quality Assessment

To ensure the reliability and high quality of the selected articles, a quality assessment was conducted on the 61 remaining articles.

In most of the previous studies, multiple authors independently reviewed articles to identify those of low quality, and the results were then consolidated to exclude such articles [28]. However, due to the characteristics of this review study, the quality assessment was conducted by two researchers independently (MZ and MFAA). Particularly, discrepancies between the two reviewers were collaboratively assessed with the aim of reaching a joint consensus. In cases where agreement could not be achieved, a third reviewer (WS) was consulted to make the final determination. This assessment focused on several aspects: first, the writing of the literature review section, ensuring it meets the needs of its research questions; second, the clarity and appropriateness of the theoretical framework provided in the article; third, the methodological section, which includes the clarity of methodological steps, and the suitability and rationality of the methods used to explore the research questions. Regarding data collection, the assessment checked the steps of data collection, standards, sample size (if applicable), number of participants, and the selection criteria of participants (if applicable). After the above steps, only one low-quality article was excluded, because it was too short and had almost no content (e.g., [41]). Although the article talked about uncertainty management in the enterprise resource planning software used by an organisation and was relevant to our research problem, it could hardly be analysed and used. As a result, the final analysis contains 60 articles in this review study.

2.3. Scientometric Analysis and Qualitative Discussion

The scientometric analysis method was used to create figures for the bibliometric data of the articles filtered by the SLR to obtain the relationship and features of the uncertainty of the SD projects. After the scientometric analysis, the qualitative discussion was conducted to analyse and summarise the causes, types, challenges, and future research directions in the SD projects.

2.3.1. Scientometric Analysis

According to relevant research [42], scientometric refers to “all quantitative aspects of science and scientific research”. This method investigates the relationships between citations to evaluate research impact and extracts knowledge from academic databases within a specific field [40]. However, various software (e.g., CiteSpace, Vosviewer) may have different influences on the results of making bibliometric maps. Vosviewer was selected to make a scientometric analysis because of its free accessibility and extensiveness [43]. By using the Vosviewer, the 60 articles were analysed to make scientometric maps in terms of keywords co-occurrence analysis, countries/regions co-occurrence analysis, and document analysis.

2.3.2. Qualitative Discussion

After obtaining the results of the scientometric analysis, the qualitative discussion was conducted. This section aims to summarise the causes, types, and challenges in SD projects and help identify the research gaps and ultimately summarise the future research directions in the SD projects.

3. Results

This section carries out a detailed analysis of the selected articles, including the results of the scientometric analysis. The scientometric analysis primarily addresses research objective 1 of this review, while the remaining research objectives are examined through systematic literature review analysis. In the scientometric analysis, the distribution map of the publication year of these 60 articles is drawn first.

3.1. Analysis of Publications Chronological Distribution

Figure 2 shows trends in the number of articles published in the field of SD projects from 1994 to 2024, with some years showing no publications. Although the number of related articles on uncertainty research in SD projects shows a fluctuating trend year by year, it is on the rise as a whole and has declined after the research peak around 2017. From 2000 to 2008, only one or two articles were published annually, likely reflecting the initial stage of research on uncertainty in SD projects. However, in 2017, it is noticed a research peak was noticed, which may be attributed to the rise in emerging computer technologies such as machine learning, big data, etc., which triggered the academic community's research interest in software design. The publication count for 2024 remains incomplete. These findings suggest that the research on uncertainty in SD projects has gradually gained research attention from less attention, and the academic community has gradually paid attention to and studied the uncertainty problems in SD projects.

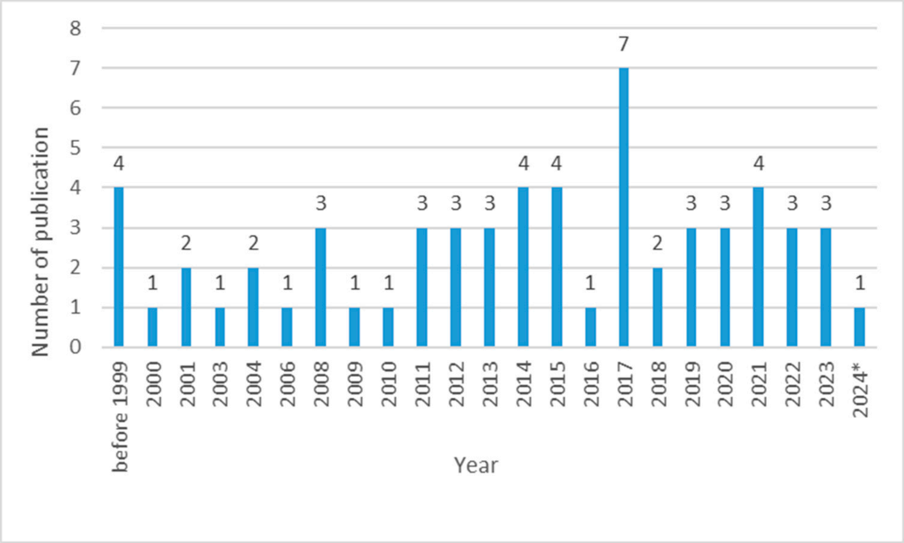

Figure 2. Publications over the year. Note: "*" indicates that data collection in 2024 ended early in July 2024.

3.2. Analysis of Geographical Distribution

Conducting a regional distribution analysis and a country collaboration network analysis of the included literature helps people quickly understand the collaborative network among countries in the research of uncertainties in SD projects, as well as identify the countries or regions with the most publications in this field. Table 1 shows the top ten countries' publication analysis of publications on uncertainty research in SD projects. Overall, the publication landscape in this field exhibits a clear dominance by one country, with the United States leading significantly with 15 publications, far surpassing other countries. Other countries have around 6 publications each. This indicates a certain regional imbalance in the research on uncertainties in SD projects. In Asia, India follows the United States with 7 publications, ranking second, while China has 5 related publications, reflecting both countries' attention to uncertainties in SD projects. Europe also shows significant interest in this field, with the United Kingdom, Germany, and Iceland publishing 6, 3, and 2 articles, respectively. The detailed information on the documents published by different countries is shown in Table 1.

Table 1. Top 10 countries publication analysis.

Country	Documents	Citations	Total Link Strength
United States	15	1977	11
India	7	124	1
Brazil	6	41	0
United Kingdom	6	279	4
China	5	189	1
Canada	4	12	5
Malaysia	4	60	3
Germany	3	21	5
Saudi Arabia	3	21	3
Iceland	2	210	2

Analysing the number of publications from each country alone may not intuitively reveal the collaborative relationships among different countries in this research field. To address this, VOSviewer was employed to create a co-authorship network map of countries, as shown in Figure 3. This figure illustrates the research collaborations among different countries in the analysis of uncertainties in SD projects.

The network analysis database divided these countries into four clusters, each represented by differently coloured circles, indicating groups of countries with close research relationships. Countries without connections to others, such as India, were excluded. In this figure, the size of the circles represents the number of publications from each country—the larger the circle, the greater the publication output. The lines of different colours indicate research relationships with countries in different clusters.

From Figure 3, it is obvious that the group of countries centred around the United States has research connections with the other three groups. The United States, as the country with the highest number of publications, collaborates with other countries such as the United Kingdom, Canada, and Ireland. Other countries within their respective clusters also exhibit strong collaborative research relationships, such as those between the United Kingdom and China or Canada and Germany. It is noteworthy that some countries, despite having a high publication frequency, are excluded from the international research collaboration network, such as India and Brazil. This indicates that these countries, while having a certain number of publications on uncertainties in SD projects, engage in minimal international collaboration. Nonetheless, in terms of both publication frequency and research collaboration with other countries, the United States remains the leading country in the study of uncertainties in SD projects.

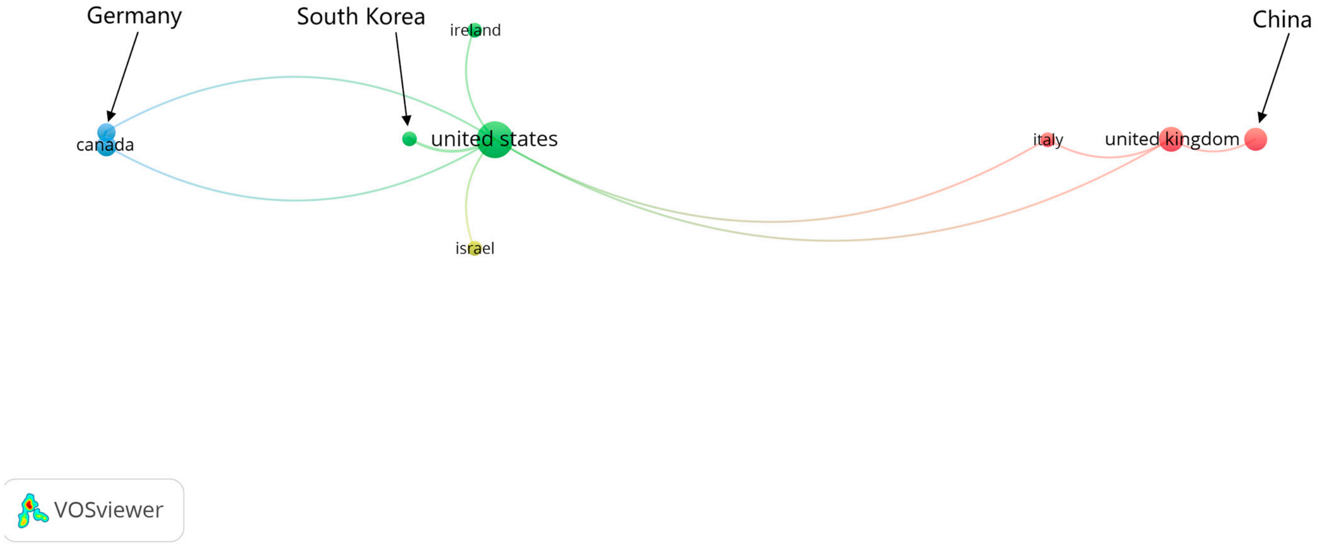

Figure 3. Co-authorship network among countries. Note: 1. China, South Korea, and Germany, denoted in Figure 3, are positioned outside the original mapping to reduce font overcrowding.

3.3. Journal Sources

An analysis of the journal sources of the included articles was conducted to identify the primary journal sources contributing to research on uncertainty in SD projects. The number of source journals for the included articles was counted, summarising the top eight journal titles, the number of included articles, and the citation scores, which are updated early in 2024. Except for a few conference papers that did not have citation scores, most journal citation scores exceeded 8, with some reaching as high as 13.8, indicating the significant influence of these source journals.

The journal with the most included articles is the "IEEE Transactions on Software Engineering" with a total of four articles. This journal focuses on the field of software engineering, covering a wide range of topics such as software-related subjects and the latest software science and technology, including artificial intelligence and machine learning. Notably, it frequently publishes articles focused on the SD process and boasts a high citation score of 9.7.

Other notable journals include "Journal of Systems and Software" and "ACM International Conference Proceeding Series" as well as "International Journal of Project Management", each contributing three articles. The "Journal of Systems and Software" is a journal with a wide range of sources related to software and software systems, encompassing nearly all aspects of software engineering, and it has a citation score of 8.6, demonstrating the high quality of the source articles. The "International Journal of Project Management" is a leading journal in project management research, covering almost all aspects of the field of project management. What is more, its high citation score of 12.3 indicates its high influence in the field of project management.

Journals contributing two articles include the "International Journal of Cognitive Computing in Engineering", "Empirical Software Engineering", "Lecture Notes in Computer Science", and "Proceedings—International Conference on Software Engineering". The first two are journal papers, while the latter two are conference papers. The "International Journal of Cognitive Computing in Engineering" focuses on articles related to self-learning systems, data mining, pattern recognition, and natural language processing, being a more specialised journal in software engineering or systems, with a citation score as high as 13.8.

It is an open-access, authoritative, peer-reviewed journal. Similarly, “Empirical Software Engineering” is a journal that covers a broad range of topics within the field of software engineering. Although the conference paper sources do not have citation scores as a reference for the number of citations, they have also contributed to many research articles.

From the above analysis and Table 2, it can be seen that the included articles come from both leading journals in software engineering and the field of project management. The high citation scores indicate that the articles they contain are frequently cited, demonstrating the high quality of the articles filtered through the PRISMA process, laying the foundation for further analysis in the subsequent sections.

Table 2. Top 8 sources of the articles on uncertainty in SD projects.

Rank	Source Title	Number	Cite Score *
1	IEEE Transactions on Software Engineering	4	9.7
2	ACM International Conference Proceeding Series	3	None
2	Journal of Systems and Software	3	8.6
2	International Journal of Project Management	3	12.3
3	International Journal of Cognitive Computing in Engineering	2	13.8
3	Empirical Software Engineering	2	8.5
3	Lecture Notes in Computer Science	2	None
3	Proceedings—International Conference on Software Engineering	2	None

Note: “*” indicates that the cite score is updated as of 15 July 2024.

3.4. Keywords Co-Occurrence Analysis

After analysing the overall publishing trend of articles including time dimension, the number of published articles in different countries, and the research relationship network among countries, this section continues to use VOSviewer to draw common keywords among the literature to identify the ongoing research hotspots of uncertainty in the current research field of SD projects. Because common keywords represent and condense a simple, comprehensive summary of an article [44]. VOSviewer is employed to generate a visual representation of keywords appearing at least twice, thereby clarifying current research interests. Figure 4 shows the co-occurrence keywords of the included articles. A total of 551 keywords were identified, of which 101 met the inclusion criteria and were visualised. In the graph, larger circles represent higher keyword frequencies. The dataset includes 101 keywords, 905 links, and a total link strength of 1159.

Table 3 presents the top 10 keywords according to frequency. As can be seen from Table 3, high-frequency keywords are not limited to software projects, software development, uncertainty, software design, project management, software engineering, etc. Software engineering and project management, software design, software projects, uncertainty, and uncertainty analysis are the keywords with the highest frequency, indicating that the article selection process of this paper is in line with the theme of exploring uncertainty in SD projects. At the same time, it can be seen from Figure 4 that there are many other keywords from the perspective of software projects and management, such as risk management, software project management, rapid development, project success, computer software, machine learning, information technology, etc. This indicates a multi-disciplinary approach in the study of uncertainties in software project development. Especially in recent years, the perspective of uncertainty research in SD projects has begun to combine with some emerging computer technologies such as machine learning.

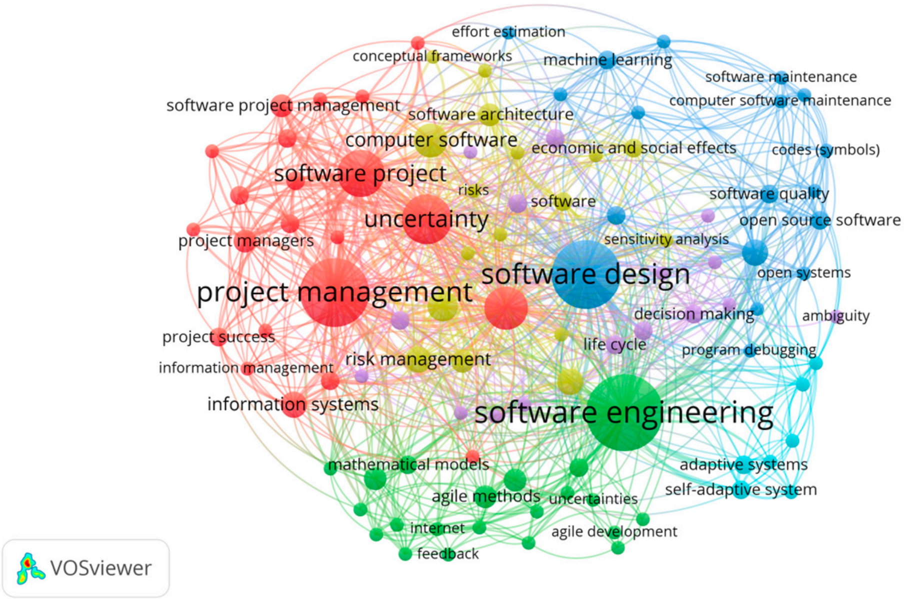

Figure 4. Keyword co-occurrence network analysis visualising research themes in uncertainty in SD projects.

Table 3. Top 10 co-keywords of included articles.

Keyword	Occurrence	Average Publication Year	Link	Total Link Strength
Software engineering	22	2012.14	78	135
Project management	19	2009.74	63	127
Software design	19	2015.21	72	124
Software project	11	2015.45	52	90
Uncertainty	12	2017.33	49	81
Uncertainty Analysis	10	2012.90	52	72
Computer software	7	2014.86	27	35
Software development projects	6	2007.83	30	39
Risk management	5	2009.20	32	40
Requirements engineering	5	2014.00	29	39

Among all the keywords, the research is divided into six clusters, with their average publication times detailed in Table 4. After organising and examining the clustering information via VOSviewer, the six clusters were named as follows: adopting project perspective in SD, agile methods in SD projects, uncertainty in software design, risk management in SD projects, mix perspective in SD projects, and focus on self-adaptive systems. Each cluster is represented by a different colour and named according to its most frequently occurring keywords and central themes. Within the same cluster, research topics often share close connections, such as issues related to uncertainty in the software design process within Cluster 3 (blue). Based on Tables 3 and 4 and Figure 4, the following mainstream research directions in SD project uncertainties emerge:

- Agile methods in SD projects (cluster 2 green)-This cluster highlights research into agile methodologies within SD projects, where keywords frequently include development methods, development processes, and agile development. For instance, ref. [45] investigated uncertainties encountered in the agile development process not only in software but also in product development.
- Risk management in SD projects (cluster 4 yellow)-In the early stage, risks associated with uncertainty in SD projects first gained research attention. Studies around 2015 identified requirements uncertainty as a significant factor jeopardising the success of

SD projects [46]. This uncertainty was often linked to software architecture and its associated risks [47].

- Adopting Project Perspective in SD projects (cluster 1 red)-As research matured, scholars began systematically examining various aspects of uncertainty from a project management perspective, as shown in Cluster 1 (red). During this period, research on uncertainty in software projects started to integrate with numerous project management perspectives. The study of project management uncertainty evolved to encompass time and cost estimation [48], project success and failure [49] and information management and information systems management [50,51], eventually developing into research on software project management.
- Uncertainty in Software Design (cluster 3 blue) -This cluster investigates uncertainty from a software design perspective, with research keywords frequently related to open systems [52], software quality, software coding [53], and software maintenance and support [54]. Many articles in this cluster are highly specialised and may not be fully comprehensible to non-professional software developers.
- Mix Perspective in SD Projects (cluster 5 purple) -In contrast, cluster 5 (purple) represents a more diverse set of topics, reflecting a wide range of studies without a single prominent focus, such as decision-making processes in development projects [55], and uncertainties in workload and budget [56,57]. The variety of topics within this cluster highlights the multifaceted nature of uncertainties in software projects.
- Focus on Self-adaptive System (cluster 6 light blue)-The last cluster, cluster 6 (light blue), represents the most recent articles on average publication date, reflecting the latest trends in SD projects, namely the development of adaptive systems. Research in this cluster focuses on model design, design time, interaction issues [58], and ensuring operational performance [59], as well as emerging technologies like blockchain development [60].

Table 4. The cluster of the co-keywords.

Average Publication Year	Cluster ID	Cluster Topic	Size
2014	1	Adopting project perspective in SD	22
2009	2	Agile methods in SD projects	21
2017	3	Uncertainty in software design	18
2015	4	Risk management in SD projects	17
2014	5	Mix perspective in SD projects	16
2019	6	Focus on self-adaptive systems	7

Based on the clustering analysis and co-occurrence network, six thematic clusters provide a holistic view of the ongoing research directions in uncertainties within SD projects. The clusters reflect both foundational studies and emerging trends, illustrating how the field has evolved to integrate traditional project management approaches with advanced technological frameworks. While some overlap exists among the research clusters, they can generally be categorised into distinct thematic groups. The clustering analysis using keywords elucidates the development trends and the latest research directions in uncertainty in SD projects, providing direct evidence for a better understanding of research in this field. In addition, the clustering analysis not only categorises research into distinct thematic areas but also highlights interdisciplinary connections and the evolution of research topics. By examining keyword co-occurrence and their relationships, the analysis provides a roadmap for understanding both established themes and emerging trends in addressing uncertainties in SD projects.

3.5. Documents Analysis

Highly cited articles are often indicative of significant academic contributions [43]. To further analyse all the articles with more citations, that is, articles with more extensive influence that have been cited and referred to by many scholars, this section selects the document that has been cited at least 75 times, and the top ten cited documents were selected to be shown in Table 5. These documents provide important references in the study of software project uncertainty. Among these ten articles, some focused on the management of SD projects at the macro level, such as [49,61,62]. These works extensively discuss uncertainty management. Other articles delve into specific aspects of uncertainty, offering diverse perspectives on its implications in SD projects.

Table 5. Top 10 most cited documents.

Authors	Document Title	Total Citations
[49]	A survey study of critical success factors in agile software projects	631
[63]	Technology Novelty, Project Complexity, and Product Development Project Execution Success: A Deeper Look at Task Uncertainty in Product Innovation	457
[64]	The Effect of Coordination and Uncertainty on Software Project Performance: Residual Performance Risk as an Intervening Variable	404
[61]	Rules and Tools for Software Evolution Planning and Management	170
[65]	Two's company, three's a crowd: A case study of crowdsourcing software development	165
[66]	Managing the sources of uncertainty: Matching process and context in software development	133
[67]	Relationships among interpersonal conflict, requirements uncertainty, and software project performance	119
[47]	Standardisation, requirements uncertainty, and software project performance	113
[62]	A model of critical success factors for software projects	98
[68]	Agile software development: Methodologies and trends	95

Ref. [49] is the most cited article in the selected literature, with 631 citations. Probably due to its early publication, it covers many important management factors in SD projects. This paper points out that in the field of SD, special attention should be paid to the uncertainty in the four aspects of organisation, people, process, and technology. Its extensive coverage of these areas has contributed to its wide citation. Similarly, ref. [68] are also talking about problems in the process of rapid development. Ref. [63] ranked the second most cited document with a total of 457 citations. The paper analyses 120 product development projects in depth, emphasising the impact of task uncertainty in the development process. The results show that the characteristics of different projects should be analysed specifically, but the technical novelty and project complexity will affect the time to market.

In contrast, other papers, such as [47,64,67], discussed the uncertainty in SD projects from the perspective of performance-related problems in SD projects, but their specific emphases were different. Ref. [64] focused on the impact of coordination and uncertainty, while [67] mainly explored the impact of requirement uncertainty and interpersonal conflict on the performance of SD projects. Ref. [47] studied the relationship between standardisation, requirement uncertainty, and SD projects' performance. Other published papers focused on crowdsourced SD projects [65] and managing sources of uncertainty in SD projects [66]. The research can provide a good reference for answering the research questions and achieving the research objectives.

4. Discussion

Following the scientometric analysis of the selected literature, this section endeavours to conduct a thorough qualitative discussion by closely reading all the included articles. Unlike risk, which involves known probabilities of outcomes, uncertainty refers to situations where outcomes or probabilities are unknown or ill-defined. For example, a risk might be a potential delay due to an identified shortage of developers, with a known likelihood based on past data. In contrast, uncertainty could involve unpredictable changes in user requirements during development, where neither the change nor its impact is foreseeable. Hence, this section aims to achieve the research objectives by summarising the causes, types, and challenges of uncertainties in the field of SD, as well as identifying future research directions.

4.1. Causes of the Uncertainty in SD Projects

First, it is essential to investigate the causes of uncertainty in SD projects. While both uncertainty and risk affect project outcomes, their causes differ fundamentally. The causes of risk typically stem from identifiable and measurable factors, such as known resource limitations, historical delays, or well-documented technical issues, making them easier to anticipate and model. In contrast, the causes of uncertainty often involve incomplete information, ambiguity, or unpredictable external conditions. For example, uncertainty may arise from vague client requirements, rapid changes in technology, or evolving regulatory environments—factors that are difficult to quantify or forecast. Through careful reading of the included literature, it is obvious that the causes of uncertainty in SD projects cannot be classified by a single standard. Instead, there are multiple classification methods. Subsequently, this section categorises all causes into three main types (see Table 6): external causes, internal causes, and other types, analysing their details to help SD project managers better understand the causes of uncertainty from different perspectives.

4.1.1. External Causes

In SD project activities, the overall environment can be categorised into internal and external environments. Within the external environment, numerous sources of uncertainty arise, including market uncertainty, environmental uncertainty, technological uncertainty, and socio-human uncertainty. each of these causes is discussed in detail below:

- **Market uncertainty—**This indicates the novelty of the final product for the market, consumers, and potential users [69]. The familiarity of consumers with a product significantly influences its acceptance [11]. However, different consumers and markets have varying preferences and behaviours. When SD teams fail to accurately understand user needs or preferences, market uncertainty escalates [70]. Incremental updates to established products typically result in predictable market reactions [69]. However, radical innovations often provoke strong, unpredictable responses. Continuous market evolution compels development teams to adjust strategies dynamically, adding to market-related uncertainties [18,71].
- **Technological uncertainty—**This type of external uncertainty is linked to the adoption of novel technologies. Emerging technologies often lack maturity, and their potential consequences are difficult to predict [72]. The level of uncertainty also depends on the organisation's accessibility to specific technologies and their readiness to implement them [73]. Challenges often become apparent after the technology is deployed, introducing delays and unforeseen risks. For example, when the new technology

was employed in developing the software, it would give rise to unexpected security vulnerabilities or even complex integration challenges. This form of uncertainty is particularly impactful in software projects reliant on cutting-edge innovations.

- **Environmental uncertainty**—Environmental uncertainty arises from the actions or demands of external entities, such as suppliers, competitors, governments, and shareholders. For instance, failing to anticipate the actions of competitors can leave a company at a disadvantage [69]. Moreover, cultural differences or conflicting requirements from stakeholders across geographic regions can introduce substantial uncertainty during project execution.
- **Socio-human uncertainty**—This category involves individuals within or interacting with the organisation and their interpersonal dynamics, including differing value systems and beliefs. Such differences can disrupt team collaboration and adversely affect project outcomes [11]. Misalignments in thought processes and decision-making approaches due to socio-cultural factors further exacerbate uncertainty.

4.1.2. Internal Causes

From the internal perspective of an SD project, uncertainties may arise at different stages, including planning and control, development, testing, and maintenance. This section explores these causes in greater detail.

- **Planning and control stage**—At the early stages of a project, planning and estimating resources, costs, and schedules inevitably introduce uncertainty. Evaluation and estimation activities often come with high levels of unpredictability [74]. As projects progress, initial plans may become obsolete due to changes in scope, timeline, or budget, resulting in significant risks. Underestimations of workload or duration can lead to abandoned tasks, rework, or even severe errors [61,75]. These estimation inaccuracies can cause time constraints and financial overruns, destabilising project management.
- **Development stage**—As the core phase of software creation, development is inherently complex, with uncertainties stemming from technology, tools, and methodologies. Established approaches tend to provide reliable outcomes, whereas newer methods often demand additional investments in training and implementation, introducing design uncertainty [48]. Developers must select tools and programming languages, whose impacts are often unpredictable, making it challenging to trace the consequences of their choices [52].
- **Test stage**—The testing phase frequently uncovers previously unknown issues, such as bugs. The process of identifying and prioritising these bugs creates inherent uncertainty, as the time and resources required to address them can vary significantly [76]. Bug resolution often depends on the expertise of assigned personnel and the complexity of the errors, introducing unpredictability into development schedules [77].
- **Maintenance and support stage**—After deployment, the software enters the maintenance phase, where uncertainties persist. Accumulated information from earlier stages may be incomplete or disorganised, complicating the process of classification and resolution [53]. Furthermore, understanding existing code during maintenance requires significant effort, and unexpected complexities often necessitate code refactoring [55]. These challenges add to the uncertainty surrounding time and resource allocation for ongoing support.

4.1.3. Other Causes

Aside from the above factors, several additional elements contribute to uncertainty in SD projects. Because those causes could arise from the internal and external factors, it is essential to separate them to call for more attention:

- **Human capital**—The diverse skills and efficiency levels of team members significantly influence the progress of projects. Variations in learning and working capabilities introduce human resource uncertainty (Özdamar and Alanya, 2001) [78].
- **Inter-project dependencies**—In organisations managing multiple projects simultaneously, uncertainties can arise from interdependencies and overlaps. Poor communication between projects and competing resource demands can disrupt workflows [79].
- **Communication gaps**—Misaligned goals and misunderstandings within teams or across departments further exacerbate uncertainty. Clear communication is crucial, yet it is often disrupted by organisational silos or cultural differences [11,62,80].

Table 6. Causes of uncertainty in SD projects.

Causes	Specific Aspect	References
External Causes	Market uncertainty	[11,18,69–71]
	Technological uncertainty	[72,73]
	Environmental uncertainty	[69]
	Socio-Human uncertainty	[11]
Internal Causes	Planning and control	[57,61,74,75]
	Development stage	[48,52]
	Test stage	[76,77]
	Maintenance and support stage	[53,55]
Other Causes	Other projects within the organisation, human capital	[62,78,81]

4.2. Types of Uncertainty in SD Projects

After discussing the causes of uncertainty in SD projects, this section summarises the common uncertainty types within these projects. It is important to distinguish these from the types of risk, as the two concepts differ not only in probability but also in nature. Risk types are generally well-defined and quantifiable—for example, schedule risk due to delayed component delivery or budget risk from underestimated development costs. In contrast, uncertainty types are more ambiguous and difficult to measure. For instance, technological uncertainty may arise from using emerging tools with unknown stability; market uncertainty can stem from unpredictable user preferences in a fast-changing digital environment. Drawing on the included articles and additional references, six major types of uncertainty are categorised, aiding researchers and practitioners in quickly identifying the uncertainties in their projects and taking appropriate actions to resolve them. Each type is elaborated below.

4.2.1. Requirement Uncertainty

Generally speaking, uncertainties can be divided into two main categories [82]: epistemic and aleatory. Epistemic uncertainty arises from a lack of knowledge, while aleatory uncertainty results from inherent variability. Among these, requirement uncertainty is the most prominent type encountered in SD projects.

Capturing customer needs accurately is fundamental to the development of any software system. This task, typically undertaken during the early stages of a project, plays a critical role in setting the foundation for subsequent development [60]. However, numerous studies have highlighted the challenges posed by requirement uncertainty in SD projects [57,83].

The customer’s requirements constitute the goals of the SD team, yet identifying, representing, and refining these requirements throughout the project often proves challenging [83]. Since requirements are usually articulated in natural language, ambiguities, multiple meanings, linguistic errors, or a lack of detail can lead to misinterpretations, resulting in software that fails to meet customer needs [84].

Moreover, customer requirements are rarely static; they may frequently change during the project due to market dynamics or new ideas, posing further challenges for SD [85]. Additionally, there is often a discrepancy between the customers' real needs and the requirements perceived by the project team, leading to extended planning times and impacting the project timeline as developers strive to bridge this gap [18].

Inadequate requirements are also a leading cause of software defects [86]. As the SD project progresses and the software matures, identifying and fixing defects becomes increasingly costly and time-consuming. In extreme cases, the associated costs can render such efforts infeasible [87]. Therefore, requirements uncertainty is identified as the primary category of uncertainty in SD projects.

4.2.2. Technological Uncertainty

The second type of uncertainty in SD projects is technological uncertainty. While it can also be considered a source of uncertainty, technological uncertainty is significant enough to be recognised as a distinct category within SD projects. Technologies employed by developers often vary across different projects, resulting in diverse and unpredictable outcomes. A lack of familiarity with specific software tools or platforms can significantly influence a project's trajectory, making it challenging to foresee the consequences with precision [48].

Even in scenarios where customer requirements are well-defined and the testing team is equipped to validate software against these requirements, the technical team may encounter difficulties in delivering viable solutions. In such instances, the technical expertise of the team may not align with the functional demands of the project, further complicating its progress [88]. Scholars have noted that the degree of technological uncertainty varies across projects, with higher uncertainty associated with the adoption of new technologies [89].

The inherent complexity of technology in SD projects amplifies risks during both the development and testing phases, consequently diminishing overall project performance [51]. However, technological uncertainty extends beyond individual developers. Even when agile development methods or standard technical processes are employed, uncertainty can only be mitigated, not eradicated entirely. Collaborative interactions among team members, for instance, can add additional layers of unpredictability [18].

Overall, technological uncertainty is closely linked to the number of two main factors: the organisation's access to mature, reliable technologies and the introduction of emerging, often unproven, technologies. Given its central role in the core activities of SD projects, addressing technological uncertainty should be a priority [70].

4.2.3. Human-Related Uncertainty

The third major category of uncertainty in SD projects pertains to human-related uncertainties. According to the research by [90], humans represent one of the greatest sources of uncertainty in project management. Externally, this is particularly evident among users, who, in the context of SD projects, represent the company's customers. Beyond the previously discussed uncertainty regarding requirements, there is the potential for user resistance to change [91]. Users may exhibit various forms of resistance, such as inaction, delayed responses, insistence on previous unsatisfactory behaviours, or consistently expressing opposing viewpoints, all of which are unpredictable [92].

Internally, within the project development team, various human-related uncertainties also exist. Firstly, communication uncertainties can arise within the team, manifested in groupthink and communication complexity [69]. This means that different team members may have different ideas about a particular issue, leading to misunderstandings during communication and resulting in work being conducted based on incorrect assumptions, causing deviations from the intended direction [88].

Moreover, individuals within the team have varying levels of capability, learning new technologies, and work efficiency. Assigning tasks to individuals with the appropriate experience can reduce uncertainties related to human capital [93]. However, in the context of global SD projects, effectively coordinating communication among personnel from different countries and assigning them appropriate tasks is a significant challenge [94].

4.2.4. Management Uncertainty

The fourth major category of uncertainty can be summarised as managerial uncertainty. Throughout the lifecycle of an SD project, project managers need to plan and control the project, which often generates a series of uncertainties. A critical aspect of SD projects is the estimation of the workload, time requirement, scheduling, and quality of the deliverable software product [57]. However, the relevant data on workload and staffing in the dynamic SD environment is complex at different stages [57].

Consequently, estimating the overall cost of the software project is inherently uncertain, leading to cost estimation uncertainty [95,96]. Additionally, the scheduling must be adjusted as time progresses. Managing the entire project also highlights resource uncertainty, an important aspect of managerial uncertainty [97], such as the lack of skilled personnel or insufficient funding.

The choice of project management methods also contributes to managerial uncertainty. In recent years, agile development methods have been favoured due to their numerous advantages over traditional waterfall methods in planning, requirements analysis, design, testing, and maintenance phases [46]. However, each method has its own challenges and potential pitfalls, making it critical for project managers to select the most suitable approach for their specific project.

4.2.5. Environment Uncertainty

SD projects are simultaneously influenced by both internal and external environments, which can introduce significant uncertainties. In the external environment, market uncertainty is particularly prominent. This type of uncertainty is heightened when customers have limited familiarity or experience with the product, making it challenging to define requirements and proceed with development effectively [66].

Additionally, when collaboration with external entities is required, such as with suppliers, uncertainties can arise, for example, if suppliers provide incorrect products or services, leading to losses in primary equipment [98]. These external factors highlight the complex and dynamic nature of environmental uncertainty in SD projects.

4.2.6. Professional Uncertainty

The final major category is professional uncertainties, which are specific to the field of SD. These uncertainties often emerge during the software design-to-delivery process, manifesting in unique challenges to these projects.

One such uncertainty is code design uncertainty during the design process, particularly in the time required for coding and testing new code until it is fully functional [78]. Issues during this phase often result in iterative cycles of rework, delaying project progress until all development goals are achieved. Moreover, software design frequently involves upgrading and modifying previous software versions, introducing platform uncertainty. This type of uncertainty correlates with the extent of changes made to the software architecture in the new version [66]. For instance, reviewing the system architecture of existing software and redesigning it introduces significant uncertainty [99].

Even after design completion, uncertainties persist during the software testing and maintenance stages, with bug-related uncertainties being particularly significant [76]. These uncertainties involve the unpredictable occurrence of bugs, their categorisation, and the assignment of appropriate personnel to address them [76]. Such factors make it difficult to forecast the resolution timeline and the resources required, further complicating the project [100].

Following the detailed discussion and classification above, this thesis categorises the uncertainties in SD projects into six major categories and fourteen specific aspects, summarised in Table 7. This classification provides a more intuitive understanding of the types of uncertainties and their Manifestations in SD projects.

Table 7. The types of uncertainty in software projects.

Types	Specific Aspects	Manifestations	References
Requirement uncertainty		• Ambiguity and ambiguity in the expression of customer needs • Frequent changes in requirements	[57,60,83–85,101]
Technical uncertainty		• The unknown results after the application of new technology • Different development technologies bring different development processes • Technical ability is difficult to meet the development needs	[18,48,51,70,88,89]
Human related Uncertainty	Customer resistance	• Oppose a proposal • deliberately delay • express a contrary opinion or delay a decision	[91,92]
	Communication Uncertainty	• Misinterpreting the words of others • Making simple problems complicated	[69,88]
	Team uncertainty	• The people in the team have different abilities • Frequent turnover of personnel	[93]
	Culture sensitivity	• People with different cultural backgrounds have different habits and customs	[94]
Management uncertainty	Resource Uncertainty	• Not enough or competent staff • Lack of financial support	[97]
	Schedule Uncertainty	• Change the schedule and milestones according to the situation	[95,96]
	Develop method uncertainty	• New development methods bring different results and explore new standard process	[46]
Environment uncertainty	Market Uncertainty	• The changeable market trend • Not clear market definition	[66]
	Supplier Uncertainty	• Interrupted supply • Wrong offer for the company	[98]
Professional Uncertainty	Code modelling Uncertainty	• Complexity of writing code • Variable modelling methods • Difficult to estimate duration of coding time	[78]
	Platform uncertainty	• The uncertainty of the extent to upgrade the software	[66]
	Bug-related Uncertainty	• Whether there will be bugs • Not clear about how many bugs will arise • Not clear how much time it will take to fix the bug	[76,100]

4.3. *Challenges of Uncertainty in SD Projects*

After discussing the causes and types of uncertainty in SD projects, this section will identify the challenges in researching uncertainty in SD projects.

4.3.1. Challenges of Research in Defining Uncertainty

A critical challenge in researching uncertainty lies in the lack of a unified definition, as highlighted in the literature review. The definition of uncertainty might seem like a simple issue, but it is fundamental to all research about uncertainty. From a broad project management perspective, starting with PMBOK's broad definition: "A lack of understanding and awareness of issues, events, paths to follow, or solutions to pursue", to scholars emphasising uncertainty in the organisational environment or equating it with ambiguity, all reflect the divergence in defining uncertainty within the extensive field of project management. Specifically, in SD projects, some of the literature does not even discuss the definition of uncertainty but directly conducts research, neglecting this fundamental issue [50].

Even when researchers attempt to define uncertainty, they often do so using a traditional project management lens, without adapting the concept to the specific contexts of SD projects. For instance, some research projects claim to study the relationship between risk and software project performance, but the measured dimensions primarily reflect uncertainty. Therefore, it is challenging to conduct research without a widely accepted common understanding, whether in broad project management or SD projects [102]. From a management perspective, uncertainty can be viewed as the deviation of project outcomes from original goals caused by unpredictable or poorly understood factors, such as ambiguous client requirements, rapidly changing technologies, or evolving stakeholder expectations in dynamic environments. In contrast, risk refers to events or conditions that have known probabilities and measurable impacts, whether positive or negative. For example, a project team might identify the risk of delayed component delivery due to a supplier's track record and allocate buffer time accordingly. However, if a new regulatory policy is suddenly introduced during the project lifecycle—without prior indication—this represents uncertainty, as neither its occurrence nor its impact could have been reliably predicted.

Moreover, the definition of uncertainty influences research methodologies and management strategies. For example,

- **Traditional project management** often treats uncertainty as one kind of risk to be minimised or avoided, because most time uncertainty would bring unexpected results. For example, when the SD projects participants do not know what would happen if they adopted the artificial intelligence tools in coding the software. This can be one aspect of technical uncertainty. But once the artificial intelligence tools are applied in the coding process and crash the software, it is a risk.
- **Agile methodologies**, on the other hand, view uncertainty as inevitable and address it through iterative development and feedback loops [45].

Specifically, in SD projects, uncertainty shows in diverse ways, such as

- Requirement instability, where project objectives shift over time.
- Technological uncertainty, stemming from emerging tools or methods.
- Team-related challenges, such as varying expertise and communication issues.

These factors affect not only project timelines and deliverables but also budgets and team dynamics, often leading to cost overruns or failures [103]. Despite the extensive impact of uncertainty, many studies fail to converge on a shared framework or terminology, limiting the practical relevance and utility of their insights. Addressing this definitional ambiguity is essential for advancing both research and practice in the field.

4.3.2. Challenges in Measurement, Monitoring, and Prediction of Uncertainty

Effective management of uncertainty in SD requires measurement, monitoring, and prediction, yet significant challenges persist in these areas.

- **Subtle Signs**—As described in the previous section, there is generally a lack of a unified definition of uncertainty complicates its measurement. A clear definition is fundamental for detecting, monitoring, and predicting uncertainty; only by understanding what needs to be measured can the corresponding activities be conducted effectively. However, early signs of uncertainty are often weak and intermittent and can have either positive or negative impacts on ongoing activities [104]. While early identification through semantic understanding can help managers recognise uncertainty [98], the detection of early signs often encounters issues like information overload and organisational filtering. For instance, managers might ignore certain signals due to personal perception biases and selective attention [105]. Even when some information is observed, it might not be correctly interpreted.

- **Complexity of the Real World**—Some researchers propose addressing uncertainty in different project stages by evaluating unforeseen uncertainties through a structured approach: identifying problems, classifying these problems, assessing the categorised uncertainties, and finally implementing various methods to address them [106]. However, in the real-world project environments, information is often ambiguous, incomplete, and inaccurate. Additionally, the urgency to address current problems frequently overshadows the need for long-term uncertainty monitoring, especially in SD, where delayed delivery incurs penalties [107]. Often, there are financial penalties for delayed delivery to clients or suppliers [108].
- **Ambiguity and Vagueness**—Ambiguity is another significant challenge in the measurement and monitoring of uncertainty [83]. During the drafting of SD requirements, developers and requirements engineers often encounter issues of low performance and efficiency, leading to approximately 82% of applications being returned for rework. Many companies need to allocate additional resources to address unclear requirements [66]. Ambiguity in requirements is unavoidable, being an inherent characteristic of natural language [101]. Although scholars have developed methods to address ambiguity in software requirements, each has its limitations. Machine learning has been suggested for detecting ambiguity [56,84], but its accuracy and handling of different types of ambiguity need improvement. Although ambiguity detection tools have been developed, they are often limited to lexical, syntactic, or grammatical ambiguities [109]. Recent advancements, including large language models like GPT-4, have shown promise in improving the understanding of ambiguous requirements, but their limitations in addressing specific contextual factors persist [84].
- **Large Data Challenges**—Daily project activities generate vast amounts of data, presenting an opportunity for applying data analysis software to measure and predict uncertainties [110]. However, making accurate early estimates of workload and time for project activities is challenging [57]. Techniques such as data mining [111], data classification, and regression analysis have been developed, but their prediction accuracy still requires improvement [112]. These large datasets require pre-processing and cleaning to enable accurate predictions [57].

4.3.3. Challenges About Tackling Uncertainty

Tackling uncertainty in SD projects involves multiple challenges, spanning psychological, technical, financial, and human resource domains. This section delves into the core challenges faced by project teams when addressing uncertainty.

- **Negative Emotions**—When uncertainty arises, it can generate negative emotions throughout the SD project team, such as frustration, restlessness, and even health-related stress [50]. These negative emotions can cause team members to feel inadequate or disillusioned about their contributions. Some team members may consistently feel that their solutions are insufficient and of poor quality, leading to dissatisfaction because no one knows which solution is the best one [50].

- **Rapid Technological Changes—**The continuous evolution of technology presents a significant challenge. While emerging methods and tools promise to eliminate operational errors, they often introduce unexpected performance issues and technical complications [55]. However, most software still faces operational interruptions or performance issues, accompanied by numerous unexpected errors. This presents technical difficulties in addressing uncertainty.
- **Limited Budget and Time**—Constraints in budget and time exacerbate challenges in tackling uncertainty. Studies have found that 75% of SD projects fail due to a lack of time rather than other reasons [113]. Under constraints of limited time and budget, focus must be placed on arranging primary tasks, leaving minimal room for progress estimation, project compatibility, and monitoring changes to address potential uncertainties [48]. As a result, managing uncertainties often takes a back seat, increasing the likelihood of project setbacks.
- **Ineffective Resource Allocation—**Effective allocation of human resources is a critical but challenging aspect of managing uncertainty. As previously mentioned, there are many human-related sources of uncertainty, and human resources are the primary resource in SD projects [114]. This requires clear identification of established tasks, dependencies between tasks, and the software manager's estimation of the workload for each task, along with available employees, their salaries, and their skill levels. During the planning of typical software projects, it is often assumed or initially estimated that the workload and skills required for a task are known and will not change arbitrarily [115]. If no uncertainty arises during project execution, it should proceed according to plan. However, project funds are limited [116], necessitating better allocation across various activities within the software project. The complexity of real-world environments often disrupts resource plans, necessitating dynamic and flexible approaches [117].
- **Stakeholder Resistance**—Uncertainty frequently demands immediate interventions, but these are often met with resistance from stakeholders both within and outside the organisation [118]. Some researchers believe that resistance is an inherent human trait [119]. User resistance is the most obvious form, manifesting as individual or group resistance to changes related to the software project [91]. Effectively managing such resistance is a crucial yet complex task, requiring project teams to balance stakeholder concerns with the need for timely action.

4.4. Future Research Directions of Uncertainty in SD Projects

Identifying future research directions for uncertainty in SD projects requires addressing existing gaps in the literature. This section discusses potential areas of investigation, starting with managing requirement uncertainty.

4.4.1. Methods of Managing Requirement Uncertainty

Requirement uncertainty is a key challenge in SD projects. While various scholars have conducted studies on managing requirement uncertainty using cognitive approaches [18], machine learning [84], and others, no research has explored the connections between different methods for managing requirement uncertainty or determined which method is comparatively more effective in minimising uncertainty during the requirements phase.

So, there is a lack of quantitative empirical studies to compare the effectiveness of methods. Therefore, future research could compare different methods for managing requirement uncertainty or investigate specific types and quantities of requirement uncertainties. For example, a comparative experiment can be conducted to verify which requirements management method is more effective in controlling uncertainties. Different indicators (such as schedule expectation rate, number of occurrences of uncertainties, and number of target deviations) can be established, and an indicator scoring method can be used to determine the effectiveness of the methods based on the scores from high to low.

4.4.2. Data Mining Applications for Dealing with Uncertainty in SD Projects

After reviewing articles on the use of data mining techniques to manage uncertainty in SD projects, it is revealed that data mining is primarily applied in managing requirement uncertainty and in project estimation and budgeting. Osman and [120] used a machine learning model to detect requirement uncertainty issues in the Malay language, achieving an accuracy rate of nearly 90%, showcasing the potential of such techniques in mitigating uncertainty. Ref. [121] adopted the neural network to estimate and refine the model's uncertainty for trustworthy decision-making, improving the model's trustworthiness in downstream decision-making tasks. But it is still unknown if the data mining techniques are used together with new information technology like cloud computing, BIM, to deal with the uncertainty in making new software.

Just like the data mining application in the construction project [122,123], the future research direction of data mining technology could explore how does the data mining combine with other information technology like BIM, internet of things to create a virtual model to prevent the uncertainty or failure in advance.

4.4.3. Uncertainty and SD Project Performance

Although scholars have conducted early studies on the general impact of uncertainty analysis on project performance in SD projects. The uncertainty usually has an influence on the SD project through the risk. For example, ref. [102] studied the impact of six dimensions of risk (essentially the impact of uncertainty) on project performance. These studies only examined simple relationships between different aspects and project performance. Moreover, in these studies, only one participant completed the risk assessment, whereas different stakeholders perceive risks in software projects differently [102,103]. For example, in a public medical software project, the government prioritises stability and cost-effectiveness, whereas the vendor is more concerned with profitability and meeting customer demands. As a result, each party focuses on different types of uncertainty. What is more, ref. [67] found that cultural differences, such as those between collectivist and individualist cultures, significantly influence interpersonal dynamics within teams. These cultural factors can further affect the way team members perceive and respond to uncertainties, influencing overall project performance.

Future research can bridge these gaps by exploring the complex relationships between uncertainty and project performance in more depth. Researchers could examine how different uncertainties influence software project performance by employing questionnaire surveys and other empirical methods. Structured Equation Modelling (SEM) could be particularly useful for establishing the causal relationships between various uncertainties and project outcomes. And studies could explore how different stakeholders—such as developers, managers, and clients—behave when confronted with uncertainties.

4.4.4. Research Framework for Uncertainty in Self-Adaptive Systems

Self-adaptive system involves designing a system with self-monitoring and self-adjusting capabilities to continuously monitor its state and the surrounding environment. It belongs to one specific kind of software. When the operating conditions change, the system can automatically modify its structure and behaviour to operate as efficiently as possible towards the defined goals. There are three research gaps in Self-Adaptive systems. First, Scholars acknowledge that quantifying uncertainty in self-adaptive systems is inherently challenging [124]. Unlike traditional systems, these frameworks often require real-time assessment of dynamic variables. Second, it remains unclear how the prioritisation of uncertainty management in self-adaptive systems compares to traditional software projects. Finally, studies have proposed methods for managing uncertainty in self-adaptive systems, but little is known about the challenges of implementing these methods and the outcomes they produce [83].

To summarise the previous discussion of research gap analysis, this review study proposes a framework diagram for future research, as shown in Table 8. It should be noted that these four aspects of future research are the majority of the uncertainties in many research SD projects, and although they involve different aspects, they are all important aspects in SD projects. These future research directions are carried out based on existing research, and expansion is not completely innovative.

Several proposed future studies on uncertainty in SD projects include, but are not limited to, the following:

1. Compare the applicability of different demand uncertainty processing tools and their advantages and disadvantages, considering the specific type and quantity of fuzziness in demand uncertainty to determine the appropriate management method.
2. Update and expand the application of advanced computer techniques, such as machine learning, in detecting project activities to carry out prediction uncertainties, such as deviation from baseline prediction.
3. Add situational factors such as different cultural backgrounds, performance at different stages, and risk tolerance to the study of uncertainty and project performance.
4. Explore types of uncertainty and quantified effects and the priority of dealing with uncertainty in adaptive systems. Need to make sure about challenges and the results evaluation of different methods of managing uncertainty.

Table 8. Research themes and trends.

Research Themes	Research Trend
Method of managing requirement uncertainty	Comparative study of different management uncertainty methods Determining the appropriate requirement uncertainty method considering the types and number of ambiguities
Data mining applications in uncertainty	Monitoring the whole process activities and predicting the deviation
Uncertainty and SD Project Performance	A comprehensive study about uncertainty considering different cultural backgrounds, different phases of the project, and various risk tolerance capabilities
uncertainty in self-adaptive system	The framework for studying the types and effects of uncertainty and the evaluation of management methods

5. Conclusions

This review study aims to investigate the causes, types, and challenges of the uncertainties in SD projects and to identify the future research directions. By following the PRISMA protocol and using a mixed review method, which consists of a systematic literature review and scientometric analysis, it selected 60 articles published from 1995 to mid-2024 from the Scopus database.

Scientometric analysis through VOSviewer provides the basic research situation concerning the uncertainty in SD projects. First, there has been an increasing number of research articles in the past 10 years, although the number of research articles has fluctuated in some years. Among all the sources of the included articles, IEEE Transactions on Software Engineering is the journal with the most published articles. Refs. [49,63] are the most influential authors, with the highest number of citations.

From a global perspective, there is a disparity in the number of publications produced by different countries and regions. The leading countries in researching uncertainty in SD projects are the United States, Brazil, India, the United Kingdom, and China. According to the keywords co-occurrence analysis, the main research focuses are on software projects, software design, uncertainty/risk management, and requirement management.

The results of the systematic review respond to the other main objectives, like causes, types, challenges, and future research directions. In terms of causes of uncertainty in the SD projects, the main causes were identified as external, internal, and others in general, specifically in 8 aspects, except for the other causes. There are basically 6 different types of uncertainties, which are requirement uncertainty, technological uncertainty, human-related uncertainty, management uncertainty, environment uncertainty, and professional uncertainty, with a total of 12 specific uncertainties. Requirement uncertainty is the most common type in SD projects. Meanwhile, there are various kinds of challenges in SD projects. This review study summarises them into three types, which are challenges in defining uncertainty, challenges in measurement, monitoring, and prediction of uncertainty, and challenges in dealing with the uncertainty.

The future research trend could be in four different themes. First, as for the management methods for requirement uncertainty, it could conduct a comparative study of different methods for managing uncertainty to determine appropriate management methods for requirement uncertainty based on the types and quantities of requirement ambiguities. Second, about the application of data mining in dealing with uncertainty, research can focus on data mining techniques with other promising techniques to monitor activities throughout the entire process and predict deviations that may be caused by uncertainty. Third, for the uncertainty and SD projects' performance, it can conduct a comprehensive study on uncertainty, taking into account factors such as different cultural backgrounds, project phases, and risk-tolerance capabilities, to understand its impact on the performance of SD projects. Finally, as for the uncertainty in self-adaptive systems, future direction could establish a research framework to study the types and impacts of uncertainty in self-adaptive systems and evaluate the corresponding management methods.

5.1. Study Implications and Contributions

Meanwhile, this review study has not only the theoretical but also the practical contributions to the uncertainty research in SD projects.

5.1.1. Theoretical Contributions

This study conducts a systematic literature review and scientometric analysis of causes, types, challenges, and future research directions. Although some studies have explored some aspects of uncertainty in SD projects, it is vital to systematically review the causes, types, and challenges within it. This study systematically categorises the causes, types, and challenges of uncertainty in SD projects, offering a foundation for targeted academic exploration. And through the scientometric analysis offering keywords, countries/regions, and document analysis, scholars can quickly find the development of the research relationship. Suggestions for future research provide a roadmap for addressing identified gaps, encouraging more scholars to contribute to this field, and expanding research developments.

5.1.2. Practical Contributions

This review is the state-of-the-art review study to offer the SD project managers guidance. Project managers can use the detailed categorization of the causes, types, and challenges to quickly identify, understand, and address the uncertainty during the development of the new software. This helps in the proactive mitigation of negative effects and enhances decision-making. Furthermore, it will also encourage SD projects practitioners to leverage technologies, such as data mining and machine learning, or other information technologies, to improve uncertainty monitoring and enhance project success rates.

5.2. Limitations and Future Studies

Although this review study employs a mixed review method, this study has inherent limitations that could impact its findings and pave the way for future research. Although Scopus was chosen for its breadth, its single-database approach may limit the study's comprehensiveness. Second, the inclusion of English-only publications excludes potentially significant findings in other languages, narrowing the research scope and excluding diverse perspectives. Finally, the review articles may not include the latest articles after July 2024. Future research could incorporate multiple databases (e.g., Web of Science, Google Scholar) and include more recent and multilingual studies to improve data coverage and reduce bias.

Funding: This research received no external funding.

Acknowledgments: The authors are grateful to the Department of Civil Engineering and Management, University of Manchester, UK, for supporting this research. The authors are also grateful to the editor and reviewers for their comments in improving the quality of this paper. This paper forms part of the first author's research project, where similar research backgrounds but different scopes and/or analyses may have been published.

Conflicts of Interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

1. Ramasesh, R.V.; Browning, T.R. A conceptual framework for tackling knowable unknown unknowns in project management. *J. Oper. Manag.* **2014**, *32*, 190–204. [CrossRef]
2. Plattfaut, R. On the Importance of Project Management Capabilities for Sustainable Business Process Management. *Sustainability* **2022**, *14*, 7612. [CrossRef]
3. Haverila, M.; Haverila, K.C.; Twyford, J.C. Critical variables and constructs in the context of project management: Importance-performance analysis. *Int. J. Manag. Proj. Bus.* **2021**, *14*, 836–864. [CrossRef]
4. Love, P.E.D.; Sing, M.C.P.; Ika, L.A.; Newton, S. The cost performance of transportation projects: The fallacy of the Planning Fallacy account. *Transp. Res. Part A Policy Pract.* **2019**, *122*, 1–20. [CrossRef]
5. Levitt, R.E. Towards project management 2.0. *Eng. Proj. Organ. J.* **2011**, *1*, 197–210. [CrossRef]
6. Shahzad, K.; Iqbal, R.; Nauman, S.; Shahzadi, R.; Luqman, A. How a Despotic Project Manager Jeopardizes Project Success: The Role of Project Team Members' Emotional Exhaustion and Emotional Intelligence. *Proj. Manag. J.* **2023**, *54*, 875697282211458. [CrossRef]
7. Ika, L.A.; Pinto, J.K. The 're-meaning' of project success: Updating and recalibrating for a modern project management. *Int. J. Proj. Manag.* **2022**, *40*, 835–848. [CrossRef]
8. Keil, M.; Rai, A.; Cheney Mann, J.E.; Zhang, G.P. Why software projects escalate: The importance of project management constructs. *IEEE Trans. Eng. Manag.* **2003**, *50*, 251–261. [CrossRef]
9. Rosińska, A.K.; Iwko, J. Stakeholder Management—One of the Clues of Sustainable Project Management—As an Underestimated Factor of Project Success in Small Construction Companies. *Sustainability* **2021**, *13*, 9877. [CrossRef]
10. Unterhitzenberger, C.; Bryde, D.J. Organizational Justice, Project Performance, and the Mediating Effects of Key Success Factors. *Proj. Manag. J.* **2019**, *50*, 57–70. [CrossRef]
11. Marinho, M.; Sampaio, S.; Moura, H. Managing uncertainty in software projects. *Innov. Syst. Softw. Eng.* **2017**, *14*, 157–181. [CrossRef]
12. Saunders, F.C.; Gale, A.W.; Sherry, A.H. Conceptualising uncertainty in safety-critical projects: A practitioner perspective. *Int. J. Proj. Manag.* **2015**, *33*, 467–478. [CrossRef]
13. Drosg, M. *Dealing with Uncertainties*; Springer: Berlin/Heidelberg, Germany, 2007; ISBN 3-3-540-29606-9.
14. Böhle, F.; Heidling, E.; Schoper, Y. A new orientation to deal with uncertainty in projects. *Int. J. Proj. Manag.* **2016**, *34*, 1384–1392. [CrossRef]

15. Jiang, J.J.; Klein, G.; Ellis, T.S. A Measure of software development Risk. *Proj. Manag. J.* **2002**, *33*, 30–41. [CrossRef]
16. Cleden, D. *Managing Project Uncertainty*; Gower: London, UK, 2009; p. 8407. ISBN 9780566088407.
17. Barghi, B. Qualitative and quantitative project risk assessment using a hybrid PMBOK model developed under uncertainty conditions. *Heliyon* **2020**, *6*, e03097. [CrossRef]
18. Haleem, M.; Farooqui, M.F.; Faisal, M. Cognitive impact validation of requirement uncertainty in software project development. *Int. J. Cogn. Comput. Eng.* **2021**, *2*, 1–11. [CrossRef]
19. Kutsch, E.; Hall, M. Intervening conditions on the management of project risk: Dealing with uncertainty in information technology projects. *Int. J. Proj. Manag.* **2005**, *23*, 591–599. [CrossRef]
20. Jalonen, H. The uncertainty of innovation: A systematic review of the literature. *J. Manag. Res.* **2012**, *4*, E12. [CrossRef]
21. Menezes, J.; Gusmão, C.; Moura, H. Risk factors in software development projects: A systematic literature review. *Softw. Qual. J.* **2019**, *27*, 1149–1174. [CrossRef]
22. Valério, K.G.O.; da Silva, C.E.S.; Neves, S.M. Risk management in software development projects: Systematic review of the state of the art literature. *Int. J. Open Source Softw. Process.* **2020**, *11*, 1–22. [CrossRef]
23. Rezaei, S. *Quantitative Tourism Research in Asia. Perspectives on Asian Tourism*; Springer Nature: Singapore, 2019. [CrossRef]
24. Hallinger, P.; Kovačević, J. A Bibliometric Review of Research on Educational Administration: Science Mapping the Literature, 1960 to 2018. *Rev. Educ. Res.* **2019**, *89*, 335–369. [CrossRef]
25. Gurbuz, H.G.; Tekinerdogan, B. Model-based testing for software safety: A systematic mapping study. *Softw. Qual. J.* **2017**, *26*, 1327–1372. [CrossRef]
26. Briner, R.B.; Walshe, N.D. From Passively Received Wisdom to Actively Constructed Knowledge: Teaching Systematic Review Skills As a Foundation of Evidence-Based Management. *Acad. Manag. Learn. Educ.* **2014**, *13*, 415–432. [CrossRef]
27. Weed, M. Sports Tourism Research 2000–2004: A Systematic Review of Knowledge and a Meta-Evaluation of Methods. *J. Sport Tour.* **2006**, *11*, 5–30. [CrossRef]
28. Zong, H.; Yi, W.; Antwi-Afari, M.F.; Yu, Y. Fatigue in construction workers: A systematic review of causes, evaluation methods, and interventions. *Saf. Sci.* **2024**, *176*, 106529. [CrossRef]
29. Anwer, S.; Li, H.; Antwi-Afari, M.F.; Umer, W.; Wong, A.Y.L. Evaluation of physiological metrics as real-time measurement of physical fatigue in construction workers: State-of-the-art review. *J. Constr. Eng. Manag.* **2021**, *147*, 03121001. [CrossRef]
30. Liberati, A.; Altman, D.G.; Tetzlaff, J.; Mulrow, C.; Gotzsche, P.C.; Ioannidis, J.P.A.; Clarke, M.; Devereaux, P.J.; Kleijnen, J.; Moher, D. The PRISMA statement for reporting systematic reviews and meta-analyses of studies that evaluate healthcare interventions: Explanation and elaboration. *BMJ* **2009**, *339*, b2700. [CrossRef]
31. Antwi-Afari, M.F.; Li, H.; Chan, A.H.S.; Seo, J.; Anwer, S.; Mi, H.-Y.; Wu, Z.; Wong, A.Y.L. A science mapping-based review of work-related musculoskeletal disorders among construction workers. *J. Saf. Res.* **2023**, *85*, 114–128. [CrossRef]
32. Salihu, C.; Hussein, M.; Mohandes, S.R.; Zayed, T. Towards a comprehensive review of the deterioration factors and modeling for sewer pipelines: A hybrid of bibliometric, scientometric, and meta-analysis approach. *J. Clean. Prod.* **2022**, *351*, 131460. [CrossRef]
33. Paul, J.; Criado, A.R. The art of writing literature review: What do we know and what do we need to know? *Int. Bus. Rev.* **2020**, *29*, 101717. [CrossRef]
34. Liu, X.; Antwi-Afari, M.F.; Li, J.; Zhang, Y.; Manu, P. BIM, IoT, and GIS integration in construction resource monitoring. *Autom. Constr.* **2025**, *174*, 106149. [CrossRef]
35. Sun, W.; Antwi-Afari, M.F.; Mehmood, I.; Anwer, S.; Umer, W. Critical success factors for implementing blockchain technology in construction. *Autom. Constr.* **2023**, *156*, 105135. [CrossRef]
36. Mohandes, S.R.; Karasan, A.; Erdoğan, M.; Ghasemi Poor Sabet, P.; Mahdiyar, A.; Zayed, T. A comprehensive analysis of the causal factors in repair, maintenance, alteration, and addition works: A novel hybrid fuzzy-based approach. *Expert Syst. Appl.* **2022**, *208*, 118112. [CrossRef]
37. Bakkalbasi, N.; Bauer, K.; Glover, J.; Wang, L. Three options for citation tracking: Google Scholar, Scopus and Web of Science. *Biomed. Digit. Libr.* **2006**, *3*, 7. [CrossRef]
38. Mongeon, P.; Paul-Hus, A. The Journal Coverage of Web of Science and Scopus: A Comparative Analysis. *Scientometrics* **2016**, *106*, 213–228. [CrossRef]
39. Rose, M.E.; Kitchin, J.R. pybliometrics: Scriptable bibliometrics using a Python interface to Scopus. *SoftwareX* **2019**, *10*, 100263. [CrossRef]
40. Ghaleb, H.; Alhajlah, H.H.; Bin Abdullah, A.A.; Kassem, M.A.; Al-Sharafi, M.A. A Scientometric Analysis and Systematic Literature Review for Construction Project Complexity. *Buildings* **2022**, *12*, 482. [CrossRef]
41. Marinho, M.; Macedo, K.; Santos, S.; Beecham, S. Uncertainty Management to Coordinate and Control an ERP Project: A Case Study. In Proceedings of the 2018 11th International Conference On The Quality Of Information And Communications Technology (QUATIC), Coimbra, Portugal, 4–7 September 2018; pp. 295–296. [CrossRef]
42. Sengupta, I.N. Bibliometrics, Informetrics, Scientometrics and Librametrics: An Overview. *Libri* **1992**, *42*, 75–98. [CrossRef]

43. Mu, X.; Antwi-Afari, M.F. The applications of Internet of Things (IoTs) in industrial management: A science mapping review. *Int. J. Prod. Res.* **2024**, *62*, 1928–1952. [CrossRef]
44. Su, H.-N.; Lee, P.-C. Mapping knowledge structure by keyword co-occurrence: A first look at journal papers in Technology Foresight. *Scientometrics* **2010**, *85*, 65–79. [CrossRef]
45. Schmidt, T.S.; Chahin, A.; Kößler, J.; Paetzold, K. Agile development and the constraints of physicality: A network theory-based cause-and-effect analysis. In Proceedings of the DS 87-4 Proceedings of the 21st International Conference on Engineering Design (ICED 17) Vol 4: Design Methods and Tools, Vancouver, BC, Canada, 21–25 August 2017; pp. 199–208, ISBN 978-1-904670-92-6.
46. Liu, J.H.; Xia, H.X.; Zhang, H.B. A Research into the UML Legend in the Waterfall Model Development. *Appl. Mech. Mater.* **2014**, *519–520*, 322–328. [CrossRef]
47. Nidumolu, S.R. Standardization, requirements uncertainty and software project performance. *Inf. Manag.* **1996**, *31*, 135–150. [CrossRef]
48. Suliman, S.M.A.; Kadoda, G. Factors that influence software project cost and schedule estimation. In Proceedings of the 2017 Sudan Conference on Computer Science and Information Technology (SCCSIT), Elnihood, Sudan, 17–19 November 2017; pp. 1–9. [CrossRef]
49. Chow, T.; Cao, D.-B. A survey study of critical success factors in agile software projects. *J. Syst. Softw.* **2008**, *81*, 961–971. [CrossRef]
50. Taipalus, T.; Seppänen, V.; Pirhonen, M. Uncertainty in information system development: Causes, effects, and coping mechanisms. *J. Syst. Softw.* **2020**, *168*, 110655. [CrossRef]
51. Wang, Q.Z.; Liu, J. Project Uncertainty, Management Practice and Project Performance: An Empirical Analysis on Customized Information Systems Development Projects. In Proceedings of the 2006 IEEE International Engineering Management Conference, Salvador, Brazil, 17–20 September 2006; pp. 341–345. [CrossRef]
52. Famelis, M.; Ben-David, N.; Di Sandro, A.; Salay, R.; Chećhik, M. MU-MMINT: An IDE for Model Uncertainty. In Proceedings of the 2015 IEEE/ACM 37th IEEE International Conference on Software Engineering, Florence, Italy, 16–24 May 2015; pp. 697–700. [CrossRef]
53. Levin, S.; Yehudai, A. Boosting Automatic Commit Classification Into Maintenance Activities By Utilizing Source Code Changes. *arXiv* **2017**, arXiv:1711.05340. [CrossRef]
54. de Lemos, R.; Garlan, D.; Ghezzi, C.; Giese, H.; Andersson, J.; Litoiu, M.; Schmerl, B.; Weyns, D.; Baresi, L.; Bencomo, N.; et al. Software Engineering for Self-Adaptive Systems: Research Challenges in the Provision of Assurances. In *Software Engineering for Self-Adaptive Systems III. Assurances*; Lecture Notes in Computer Science; Springer: Cham, Switzerland, 2017; pp. 3–30. [CrossRef]
55. Mkaouer, M.W.; Kessentini, M.; Cinnéide, M.Ó.; Hayashi, S.; Deb, K. A robust multi-objective approach to balance severity and importance of refactoring opportunities. *Empir. Softw. Eng.* **2016**, *22*, 894–927. [CrossRef]
56. Haris, M.; Chua, F.-F.; Lim, A.H.-L. An Ensemble-Based Framework to Estimate Software Project Effort. In Proceedings of the 2023 IEEE 8th International Conference On Software Engineering and Computer Systems, Penang, Malaysia, 25–27 August 2023; pp. 47–52. [CrossRef]
57. Pospieszny, P. Software estimation: Towards prescriptive analytics. In Proceedings of the 27th International Workshop on Software Measurement and 12th International Conference on Software Process and Product Measurement, Gothenburg, Sweden, 25–27 October 2017; pp. 221–226. [CrossRef]
58. Camara, J.; Troya, J.; Vallecillo, A.; Bencomo, N.; Calinescu, R.; Cheng, B.; Garlan, D.; Schmerl, B. The Uncertainty Interaction Problem in Self-Adaptive Systems. *Softw. Syst. Model.* **2022**, *21*, 1277–1294. [CrossRef]
59. Haleem, M.; Farooqui, M.F.; Faisal, M. A Critical Analysis of Software Product Failure: An Indian & Global Perspective. *Int. J. Eng. Adv. Technol.* **2019**, *8*, 106–114. [CrossRef]
60. Farooq, M.S.; Ahmed, M.; Emran, M. A Survey on Blockchain Acquainted Software Requirements Engineering: Model, Opportunities, Challenges, and Future Directions. *IEEE Access* **2022**, *10*, 48193–48228. [CrossRef]
61. Lehman, M.M.; Ramil, J.F. Rules and tools for software evolution planning and management. *Ann. Softw. Eng.* **2001**, *11*, 15–44. [CrossRef]
62. Purna Sudhakar, G. A model of critical success factors for software projects. *J. Enterp. Inf. Manag.* **2012**, *25*, 537–558. [CrossRef]
63. Tatikonda, M.V.; Rosenthal, S.R. Technology novelty, project complexity, and product development project execution success: A deeper look at task uncertainty in product innovation. *IEEE Trans. Eng. Manag.* **2000**, *47*, 74–87. [CrossRef]
64. Nidumolu, S. The Effect of Coordination and Uncertainty on Software Project Performance: Residual Performance Risk as an Intervening Variable. *Inf. Syst. Res.* **1995**, *6*, 191–219. [CrossRef]
65. Stol, K.-J.; Fitzgerald, B. Two's company, three's a crowd: A case study of crowdsourcing software development. In Proceedings of the 36th International Conference on Software Engineering—ICSE 2014, Hyderabad, India, 31 May–7 June 2014. [CrossRef]
66. MacCormack, A.; Verganti, R. Managing the Sources of Uncertainty: Matching Process and Context in Software Development. *J. Prod. Innov. Manag.* **2003**, *20*, 217–232. [CrossRef]
67. Liu, J.Y.-C.; Chen, H.-G.; Chen, C.C.; Sheu, T.S. Relationships among interpersonal conflict, requirements uncertainty, and software project performance. *Int. J. Proj. Manag.* **2011**, *29*, 547–556. [CrossRef]

68. Alsaqqa, S.; Sawalha, S.; Abdel-Nabi, H. Agile software development: Methodologies and Trends. *Int. J. Interact. Mob. Technol.* **2020**, *14*, 246–270. [CrossRef]
69. Marinho, M.; Sampaio, S.; Moura, H. An Approach Related to Uncertainty in Software Projects. In Proceedings of the 2013 IEEE International Conference On Systems, Man, and Cybernetics (SMC 2013), Manchester, UK, 13–16 October 2013; pp. 894–899. [CrossRef]
70. Marinho, M.; Sampaio, S.; Moura, H. Uncertainties in software projects management. In Proceedings of the 2014 XL Latin American Computing Conference (CLEI), Montevideo, Uruguay, 15–19 September 2014; pp. 1–10. [CrossRef]
71. Marinho, M.; Sampaio, S.; Luna, A.; Lima, T.; Moura, H. Dealing with uncertainties in software project management. In Proceedings of the 2015 IEEE International Conference on Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing, Liverpool, UK, 26–28 October 2015; pp. 889–894. [CrossRef]
72. Shenhar, A.J. From low- to high-tech project management. *RD Manag.* **1993**, *23*, 199–214. [CrossRef]
73. O'Connor, G.C.; Rice, M.P. A Comprehensive Model of Uncertainty Associated with Radical Innovation. *J. Prod. Innov. Manag.* **2013**, *30* (Suppl. 1), 2–18. [CrossRef]
74. Selby, R.W. Software engineering: The legacy of Barry W. Boehm. In Proceedings of the 29th International Conference on Software Engineering (ICSE'07 Companion), Minneapolis, MN, USA, 20–26 May 2007. [CrossRef]
75. Antoniol, G.; Di Penta, M.; Harman, M. A robust search-based approach to project management in the presence of abandonment, rework, error and uncertainty. In Proceedings of the IEEE International Software Metrics Symposium, Chicago, IL, USA, 11–17 September 2004; pp. 172–183. [CrossRef]
76. Jahanshahi, H.; Cevik, M.; Mousavi, K.; Başar, A. ADPTriage: Approximate Dynamic Programming for Bug Triage. *IEEE Trans. Softw. Eng.* **2023**, *49*, 4594–4609. [CrossRef]
77. Park, J.; Lee, M.-W.; Kim, J.; Hwang, S.; Kim, S. Cost-aware triage ranking algorithms for bug reporting systems. *Knowl. Inf. Syst.* **2015**, *48*, 679–705. [CrossRef]
78. Özdamar, L.; Alanya, E. Uncertainty Modelling in Software Development Projects (with Case Study). *Ann. Oper. Res.* **2001**, *102*, 157–178. [CrossRef]
79. Szelągowski, M.; Berniak-Woźny, J. BPM challenges, limitations and future development directions—A systematic literature review. *Bus. Process Manag. J.* **2024**, *30*, 505–557. [CrossRef]
80. Schmidt, C.; Dart, P.; Johnston, L.; Sterling, L.; Thorne, P. Disincentives for communicating risk: A risk paradox. *Inf. Softw. Technol.* **1999**, *41*, 403–411. [CrossRef]
81. Ribeiro, L.; Gusmão, C.; Feijó, W.; Bezerra, V. A case study for the implementation of an agile risk management process in multiple projects environments. In Proceedings of the PICMET'09-2009 Portland International Conference on Management of Engineering & Technology, Portland, OR, USA, 2–6 August 2009; pp. 1396–1404. [CrossRef]
82. Senge, R.; Bösner, S.; Dembczyński, K.; Haasenritter, J.; Hirsch, O.; Donner-Banzhoff, N.; Hüllermeier, E. Reliable classification: Learning classifiers that distinguish aleatoric and epistemic uncertainty. *Inf. Sci.* **2014**, *255*, 16–29. [CrossRef]
83. Muñoz-Fernández, J.C.; Knauss, A.C.; Castañeda, L.C.; Derakhshanmanesh, M.C.; Heinrich, R.; Becker, M.; Taherimakhsousi, N.C. Capturing Ambiguity in Artifacts to Support Requirements Engineering for Self-Adaptive Systems. In Proceedings of the RESACS: 3rd International Workshop on Requirements Engineering for Self-Adaptive & Cyber Physical System, Essen, Germany, 27 February 2017; Available online: https://paris1.hal.science/hal-01513011 (accessed on 12 July 2024).
84. Muhamad, F.N.J.; Ab Hamid, S.H.; Subramaniam, H.; Abdul Rashid, R.; Fahmi, F. Fault-Prone Software Requirements Specification Detection Using Ensemble Learning for Edge/Cloud Applications. *Appl. Sci.* **2023**, *13*, 8368. [CrossRef]
85. Jha, P.; Patnaik, K.S. Assessing Overall Software Defect-Based Risk Using Analytic Hierarchy Process. In Proceedings of the 4th International Conference on Microelectronics, Computing and Communication Systems; Lecture Notes in Electrical Engineering. Springer: Singapore, 2020; pp. 123–134. [CrossRef]
86. El Ghazi El Houssaïni, S.; Maskani, I.; Boutahar, J. A Security Requirement Engineering Case Study: Challenges and Lessons Learned. In *Intelligent Computing*; Lecture Notes in Networks and Systems; Springer: Cham, Switzerland, 2021; pp. 761–783. [CrossRef]
87. George, G.; Lakhani, K.R.; Puranam, P. What Has changed? the Impact of Covid Pandemic on the Technology and Innovation Management Research Agenda. *J. Manag. Stud.* **2020**, *57*, 1754–1758. [CrossRef]
88. Kumlander, D. Bridging uncertainties gaps in software development projects. In Proceedings of the 10th International Conference on Enterprise Information Systems, Barcelona, Spain, 12–16 June 2008; pp. 240–245. [CrossRef]
89. Shenhar, A.J.; Dvir, D. Managing technology projects: A contingent exploratory approach. In Proceedings of the Annual Hawaii International Conference on System Sciences, Wailea, HI, USA, 3–6 January 1995; Volume 3, pp. 494–503. [CrossRef]
90. de Bakker, K.; Boonstra, A.; Wortmann, H. Does risk management contribute to IT project success? A meta-analysis of empirical evidence. *Int. J. Proj. Manag.* **2010**, *28*, 493–503. [CrossRef]

91. Kim, H.-W.; Kankanhalli, A. Investigating User Resistance to Information Systems Implementation: A Status Quo Bias Perspective. *MIS Q.* **2009**, *33*, 567–582. [CrossRef]
92. Lapointe, L.; Rivard, S. A Multilevel Model of Resistance to Information Technology Implementation. *MIS Q.* **2005**, *29*, 461. [CrossRef]
93. Misra, S.C.; Kumar, V.; Kumar, U. Identifying some important success factors in adopting agile software development practices. *J. Syst. Softw.* **2009**, *82*, 1869–1890. [CrossRef]
94. Wiredu, G.O. Global Software Development and the Problem of Increased Uncertainties. *J. Glob. Inf. Manag.* **2012**, *20*, 1–24. [CrossRef]
95. Boehm, B.W.; Papaccio, P.N. Understanding and controlling software costs. *IEEE Trans. Softw. Eng.* **1988**, *14*, 1462–1477. [CrossRef]
96. Lee, G.-S.; Murata, T. A β-distributed stochastic petri net model for software project time/cost management. *J. Syst. Softw.* **1994**, *26*, 149–165. [CrossRef]
97. Jiang, J.; Klein, G. Software development risks to project effectiveness. *J. Syst. Softw.* **2000**, *52*, 3–10. [CrossRef]
98. Russo, R.D.F.S.M.; Sbragia, R.; Yu, A.S.O. Determining factors in the unforeseeable uncertainty management in innovation projects. In Proceedings of the 2013 Proceedings of PICMET'13: Technology Management in the IT-Driven Services (PICMET), San Jose, CA, USA, 28 July–1 August 2013; ISBN 623-634.978-1-890843-27-4.
99. Marples, D.L. The Decisions Of Engineering Design. *IRE Trans. Eng. Manag.* **1961**, *EM-8*, 55–71. [CrossRef]
100. Aung, T.W.W.; Wan, Y.; Huo, H.; Sui, Y. Multi-triage: A multi-task learning framework for bug triage. *J. Syst. Softw.* **2022**, *184*, 111133. [CrossRef]
101. Alomari, R.; Elazhary, H. Implementation of a Formal Software Requirements Ambiguity Prevention Tool. *Int. J. Adv. Comput. Sci. Appl.* **2018**, *9*, 424–432. [CrossRef]
102. Wallace, L.; Keil, M.; Rai, A. How Software Project Risk Affects Project Performance: An Investigation of the Dimensions of Risk and an Exploratory Model. *Decis. Sci.* **2004**, *35*, 289–321. [CrossRef]
103. Yang, B.; Hu, H.; Jia, L. A Study of Uncertainty in Software Cost and Its Impact on Optimal Software Release Time. *IEEE Trans. Softw. Eng.* **2008**, *34*, 813–825. [CrossRef]
104. Ansoff, H.I. Managing Strategic Surprise by Response to Weak Signals. *Calif. Manag. Rev.* **1975**, *18*, 21–33. [CrossRef]
105. Kull, T.J.; Oke, A.; Dooley, K.J. Supplier selection behavior under uncertainty: Contextual and cognitive effects on risk perception and choice. *Decis. Sci.* **2014**, *45*, 467–505. [CrossRef]
106. Loch, C.H.; Solt, M.E.; Bailey, E.M. Diagnosing Unforeseeable Uncertainty in a New Venture. *J. Prod. Innov. Manag.* **2007**, *25*, 28–46. [CrossRef]
107. Pham, H. A software cost model with imperfect debugging, random life cycle and penalty cost. *Int. J. Syst. Sci.* **1996**, *27*, 455–463. [CrossRef]
108. Hou, R.-H.; Kuo, S.-Y.; Chang, Y.-P. Optimal release times for software systems with scheduled delivery time based on the HGDM. *IEEE Trans. Comput.* **1997**, *46*, 216–221. [CrossRef]
109. Nigam, A.; Arya, N.; Nigam, B.; Jain, D. Tool for Automatic Discovery of Ambiguity in Requirements. *Int. J. Comput. Sci. Issues* **2012**, *9*, 350.
110. Hariri, R.H.; Fredericks, E.M.; Bowers, K.M. Uncertainty in big data analytics: Survey, opportunities, and challenges. *J. Big Data* **2019**, *6*, 44. [CrossRef]
111. López-Martín, C.; Abran, A. Neural networks for predicting the duration of new software projects. *J. Syst. Softw.* **2015**, *101*, 127–135. [CrossRef]
112. Kocaguneli, E.; Menzies, T.; Keung, J.W. On the Value of Ensemble Effort Estimation. *IEEE Trans. Softw. Eng.* **2012**, *38*, 1403–1416. [CrossRef]
113. Lehtinen, T.O.; Mäntylä, M.V.; Vanhanen, J.; Itkonen, J.; Lassenius, C. Perceived causes of software project failures–An analysis of their relationships. *Inf. Softw. Technol.* **2014**, *56*, 623–643. [CrossRef]
114. Barreto, A.; Barros, M.d.O.; Werner, C.M.L. Staffing a software project: A constraint satisfaction and optimization-based approach. *Comput. Oper. Res.* **2008**, *35*, 3073–3089. [CrossRef]
115. Shen, X.; Minku, L.L.; Bahsoon, R.; Yao, X. Dynamic Software Project Scheduling through a Proactive-Rescheduling Method. *IEEE Trans. Softw. Eng.* **2016**, *42*, 658–686. [CrossRef]
116. Lesca, N.; Caron-Fasan, M.-L. Strategic scanning project failure and abandonment factors: Lessons learned. *Eur. J. Inf. Syst.* **2008**, *17*, 371–386. [CrossRef]
117. Sommerville, I. *Software Engineering*, 9th ed.; Pearson Education Inc.: Boston, MA, USA, 2011; ISBN 0-13-703515-21.
118. Vrhovec, S.L.R.; Hovelja, T.; Vavpotič, D.; Krisper, M. Diagnosing organizational risks in software projects: Stakeholder resistance. *Int. J. Proj. Manag.* **2015**, *33*, 1262–1273. [CrossRef]
119. Laumer, S. Why Do People Reject Technologies—A Literature-Based Discussion of the Phenomena "Resistance to Change" in Information Systems and Managerial Psychology Research. 2011. Available online: https://aisel.aisnet.org/ecis2011/60 (accessed on 12 July 2024).

120. Osman, M.H.; Zahrin, M.F. Ambi detect: An ambiguous software requirements specification detection tool. *Turk. J. Comput. Math. Educ.* **2021**, *12*, 2023–2028. [CrossRef]
121. Thuy, A.; Benoit, D.F. Explainability through uncertainty: Trustworthy decision-making with neural networks. *Eur. J. Oper. Res.* **2024**, *317*, 330–340. [CrossRef]
122. Pan, Y.; Zhang, L. A BIM-data mining integrated digital twin framework for advanced project management. *Autom. Constr.* **2021**, *124*, 103564. [CrossRef]
123. Deng, Y.; Liu, Z.; Song, L.; Ni, G.; Xu, N. Exploring the metro construction accidents and causations for improving safety management based on data mining and network theory. *Eng. Constr. Archit. Manag.* **2024**, *31*, 3508–3532. [CrossRef]
124. Ismail, S.; Shah, K.; Reza, H.; Marsh, R.; Grant, E. Toward Management of Uncertainty in Self-Adaptive Software Systems: IoT Case Study. *Computers* **2021**, *10*, 27. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.