

Prioritizing Use Cases: A Systematic Literature Review

Yousra Odeh ^{1,*} and Nedhal Al-Saiyd ²¹ Faculty of Information Technology, Philadelphia University, Amman 19392, Jordan² Faculty of Technical Engineering, Uruk University, Baghdad 10001, Iraq; nedhal.abdul-majied@uruk.edu.iq

* Correspondence: yodeh@philadelphia.edu.jo

Abstract: The prioritization of software requirements is necessary for successful software development. A use case is a useful approach to represent and prioritize user-centric requirements. Use-case-based prioritization is used to rank use cases to attain a business value based on identified criteria. The research community has started engaging use case modeling for emerging technologies such as the IoT, mobile development, and big data. A systematic literature review was conducted to understand the approaches reported in the last two decades. For each of the 40 identified approaches, a review is presented with respect to consideration of scenarios, the extent of formality, and the size of requirements. Only 32.5% of the reviewed studies considered scenario-based approaches, and the majority of reported approaches were semiformally developed (53.8%). The reported result opens prospects for the development of new approaches to fill a gap regarding the inclusive of strategic goals and respective business processes that support scenario representation. This study reveals that existing approaches fail to consider necessary criteria such as risks, goals, and some quality-related requirements. The findings reported herein are useful for researchers and practitioners aiming to improve current prioritization practices using the use case approach.

Keywords: use case prioritization; systematic literature review; use case model; requirements prioritization; requirement engineering; project management



Citation: Odeh, Y.; Al-Saiyd, N. Prioritizing Use Cases: A Systematic Literature Review. *Computers* **2023**, *12*, 136. <https://doi.org/10.3390/computers12070136>

Academic Editor: Paolo Bellavista

Received: 25 May 2023

Revised: 30 June 2023

Accepted: 1 July 2023

Published: 6 July 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Software requirements prioritization is one of the early and crucial processes conducted in software management to produce high-quality software, optimize investment in resources, and reduce potential risks. Requirements prioritization (RP) is an iterative process whereby a set of requirements is ranked from highest to lowest priority in relation to their importance to the development of the most critical requirements for the earliest release of a project [1]. RP is a complex, multicriteria decision-making process [1], contributing to increased customer satisfaction, even in cases in which a project is overbudget or behind schedule, as the highly prioritized requirements are delivered early in the project.

Software engineers strive to develop highly flexible, reusable, and maintainable systems. A well-known paradigm in this context is the component-based paradigm. In this paradigm, prioritization is applied to use cases that are user-centered functionality artifacts [2]. The research community has reported its efforts to develop requirements prioritization techniques whereby requirements are specified in a textual natural language form [3]. Although natural language text-based specified requirements are vital, prioritization cannot be omitted. However, natural language requirements lack some necessary characteristics. First, specifying precise and unambiguous requirements written in natural language is difficult. Second, a natural language depends on the reader's and the writer's understanding and skills, as there is no standardized format for writing natural requirements statements. Unlike standardized visuals, misinterpretation is significantly reduced. These visual models assist different stakeholders in arriving at a common understanding of system requirements, facilitating the prioritization process. For example, a use case diagram assists in visualizing user-centered requirements that show the scope of the system.

Third, specifying natural language requirements is time-consuming, resulting in the need for RP, especially for requirements that lack clarity. Fourth, finding the dependencies between natural language requirements and their change impact is difficult unless they are labelled, automated, and fulfilled within a requirements traceability matrix, which cannot easily detect semantic tradeoffs. Natural language requirements inevitably lack a unique interpretation. Therefore, to minimize the impact of these issues while prioritizing requirements, some researchers have proposed a data-entity-focused linguistic pattern that is extracted from use cases as reusable artifacts [4]. Use cases are feasible for requirements prioritization. In the past two decades, researchers have developed use-case-based prioritization approaches and investigated their effectiveness in eliciting and prioritizing essential requirements and scenarios [5–7]. Tools have also been developed to assist analysts in collecting requirements from essential use cases [8]. The use case concept has been employed for emerging subjects, such as bridging the gap between business and systems [9]. The contributions of this research are as follows:

- Summary and organization of all use case prioritization approaches in one repository to open prospects for improvement in terms of usefulness and effectiveness with respect to emerging technologies;
- Acting as a roadmap by classifying all use case prioritization approaches according to the inclusion of scenarios, size, and formality to assist stakeholders in selecting an appropriate approach;
- Identifying shortcomings regarding necessary but missing criteria that may lead to the integration of other research areas, for example, goal-oriented and business process models.

The use case concept was introduced in the mid-1990s as a tool to assist in eliciting user and functional requirements in software engineering [10–13], acting as a driver for test cases through the application of forward engineering [14,15]. The rich specification of a use case is used to estimate effort and cost [16,17], as well as in planning and deriving test cases [18,19]. A significant role of use cases has been reported in bridging the gap between business and software systems [9,20,21].

Whether a traditional or agile software engineering process is adopted, RP is conducted during each phase (i.e., requirements engineering, design, implementation, testing, and maintenance) [22] and during the planning of each sprint or iteration with the obligatory involvement of the customer and relevant stakeholder classes [23]. Therefore, owing to the necessity of prioritization, RP occupies a considerable place in the literature as researchers have reported the prioritization methods for large, medium, and small projects [3,24–26]. Researchers have recognized the criticality of RP not only in software development but also in industry and the Internet of Things [27–30].

The research community has reported a significant lack of reviews of current use-case-based prioritization approaches within recent decades [31]. Only five approaches have been reported in a small review. Table 1 shows a summary of the gap between the work reported in [31] and this work, indicating the need to review current approaches to derive the current state of the art and identify and discuss gaps to facilitate development in various fields that employ the concept of use cases, such as business/IT alignment, artificial intelligence, big data, health informatics, and the Internet of Things. The current review can verify the sustainability of the use case concept for further evolution in IT-based projects.

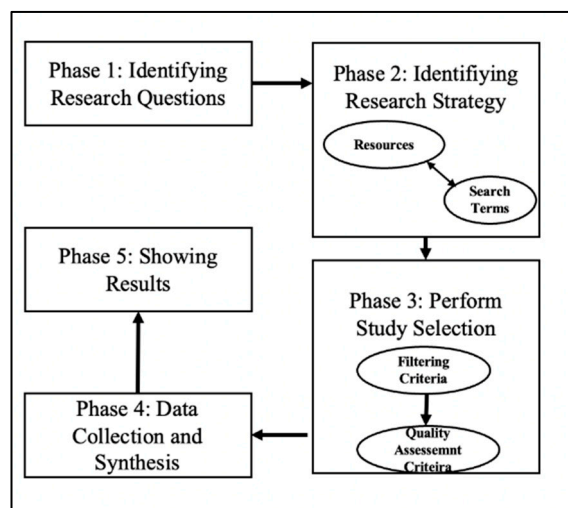
The remainder of this paper is structured in five sections. In Section 2, we present the research methodology. Results and answers to the specified research questions are presented in Section 3. A critical discussion is presented in Section 4. Finally, Section 5 concludes the work.

Table 1. A Summary of the gap between the current literature and this research work on use case prioritization.

Comparison Factor	Current Literature [31]	This Research
Number of reviewed approaches	5	40
Number of scenario-based approaches	2	13
Prioritization criteria	Goals, usage of use case, safety, security, infrastructure, dependency, frequency, revenues, availability, reliability, density of scenario path, structural aspects, number of actors, number of included use cases in a use case, number of use cases that include a use case, and customer prioritization	All criteria in [31] and more
Domains	Fuzzy logic systems	The majority in software engineering. The work reports domains such as industry and emerging technologies.

2. Materials and Methods

The research methodology of this systematic literature review was designed as a protocol-based process. The process was adopted from the medical domain and redesigned and adapted to software engineering research. The protocol process was previously in other reviews [3,32,33]. We refined, blueprinted, and applied the process in this research, as shown in Figure 1. Each rectangle represents a main phase in the protocol-based process: identifying a specific set of research questions that stem from the research problem; designing the research strategy; performing study selection; conducting data collection and analysis; and, finally, producing review results. The following subsections present each phase in the designed methodology.

**Figure 1.** Research methodology.

2.1. Identifying Research Questions

Having a clear research problem facilitates the derivation of a research aim and questions. Given the lack of literature reviews summarizing research on use case prioritization in the last two decades, with the use case concept related to emerging technology topics, all reported attempts need to be reviewed to align them with the current growth era in use-case-centric software development [34–36].

In this research, we conducted a systematic literature review to summarize the existing literature on use-case-based prioritization approaches in software development. In this

review, we consider research dimensions related to use case prioritization, prioritization criteria, the size of the project use cases or requirements, contributions, and limitations. The following research questions were formulated to be answered through the literature review:

RQ1: *What is the significance of use cases in requirements prioritization?*

RQ2: *Which reported studies have employed use the case concept, modeling, or scenarios for requirements prioritization?*

RQ3: *Which stakeholder classes are involved, and what are their views on use-case-based prioritization?*

RQ4: *What gaps exist in the literature that must be addressed to enhance existing use-case-based prioritization approaches?*

2.2. Research Strategy

In the first phase, we identified relevant digital libraries and search strings to answer the four research questions in related articles, conference proceedings, technical reports, books, book chapters, patents, theses, and journals. We searched eight databases: IEEE Xplore, ACM, Google Scholar, Elsevier (Science Direct), Semantic Scholar, Springer, Research Gate, and Academia. These digital libraries were selected because they encompass the most relevant work published in software engineering, in addition to the advanced search options provided on their websites [36]. Other digital libraries were considered for this research, such as EBSCO and Pro-Quest; however, they were excluded because they returned no results regarding the selected search strings, possibly highlighting a shortage of use-case-based RP publications.

The search strings were derived from the research problem. Core and common searching strings were “Use case prioritization” AND/OR “Prioritizing use cases”, including different UK/US spellings and syntax. The following strings were used:

1. Significance OR Importance of prioritizing use case OR use case prioritization (RQ1);
2. Use case prioritization methods OR techniques OR algorithms OR approaches (RQ2);
3. Prioritizing use cases (RQ2);
4. “Use Case” prioritization (RQ2);
5. Criteria OR Factors used in prioritizing use cases OR use case prioritization (RQ2);
6. Benefits OR advantages OR Contributions of use case prioritization (RQ2);
7. Size of use cases in use case prioritization method OR technique OR algorithm OR approach (RQ2);
8. Stakeholders in use case prioritization (RQ2 and RQ3);
9. Limitations OR Challenges in use case prioritization OR prioritizing use cases (RQ2 and RQ4);

The above strings paved the way to answer the research questions indicated next to each search string.

2.3. Study Selection Strategy

In this stage, we applied inclusion and exclusion criteria, followed by the quality assessment criteria (QAC) checklist. A total of 40 studies were included in this research after filtering and selection.

Inclusion Criteria

- Studies published in English;
- A relevant answer to at least one research question within the publication;
- The paper includes strings with any UK/US spelling of “use case prioritization”, “scenario prioritization”, and “use-case prioritization”;
- Technical reports, white papers, and research studies from industry domain experts that do not necessarily employ use case prioritization in software engineering but in other areas, such as IoT and big data.

Exclusion Criteria

- Studies published in languages other than English;
- No relevant answer to at least one research question in the publication’s content;

- Redundant studies.

QAC were formulated as a checklist comprising five criteria with associated scoring points, as shown in Table 2. The QAC were used for filtering after applying the exclusion criteria. The Five QACs were applied to publications titles, abstracts, content, and conclusions. Any publication that attained fewer than 5 points was excluded. A total of 40 studies were selected for this research. The pie chart in Figure 2 shows a quantitative classification of the publications selected after applying the inclusion criteria and QAC.

Table 2. The quality assessment criteria.

QAC No.	Question	Point Scores of Answers
1	Is the aim of the research defined and addressed?	
2	Does the research scope focus on use case prioritization?	
3	Is the presented approach/algorithm/technique clearly explained?	Yes = 2, moderately = 1, no = 0.
4	Is the use-case-based prioritization evaluation technique clearly explained using a case study or datasets?	
5	Is the result of the research clearly explained?	

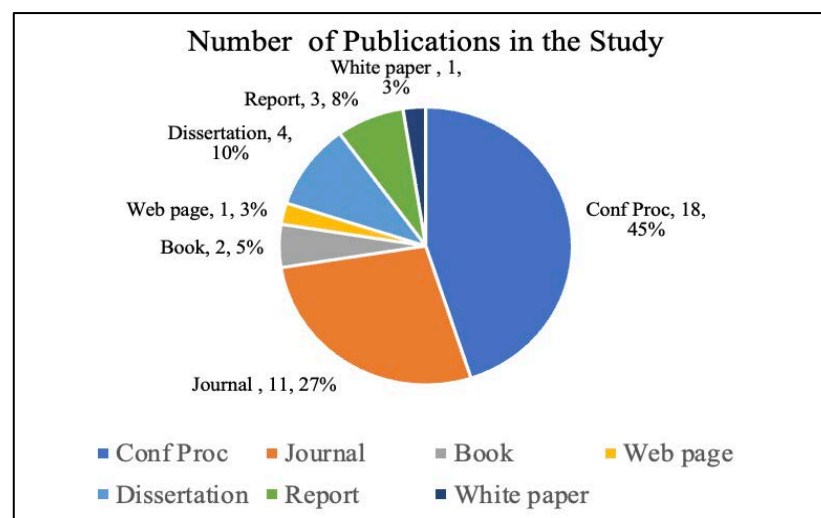


Figure 2. Quantitative Classification of Resulted Publications in this Research.

2.4. Data Collection and Synthesis

Each study was analyzed carefully to obtain relevant answers to the four research questions. Therefore, quantitative and qualitative data are presented. Any data clarifying the importance of prioritizing use cases is considered an answer to RQ1. Data related to use case prioritization approaches that involve prioritization criteria such as project requirement size and formality are considered an answer to RQ2. Information about the contributions and limitations of each use-case-based prioritization approach answers RQ2. Information about stakeholder viewpoints and involvement in use case prioritization answers RQ3. Identifying common gaps across reviewed and analyzed works assists in formulating answers to RQ4.

3. Answering the Research Questions

A total of 40 publications were selected after applying the inclusion and exclusion criteria, including journal articles, technical reports, dissertations, conference proceedings, books, white papers, and web pages. The included publication types and associated percentages are shown in Figure 2. The most prevalent type of publication was conference proceedings, followed by journal articles, accounting for 45% and 27%, respectively,

whereas dissertations and reports accounted for 10% and 8%, respectively; books accounted for 5%; and webpages and white papers accounted for 3% of included publications.

3.1. RQ1: Significance of Employing Use Cases in Requirements Prioritization

The importance of employing a use case diagram in prioritization stems from its distinctive visualization with rich embedded requirements. A use case diagram also presents requirements in textual form, using descriptions [37]. The concept of scenarios is another rich information source within use cases. The research community considers scenario prioritization within use cases to be an important early activity in project planning within the process of software development [37]. However, covering all scenarios with limited time, budget, and resources is difficult [37,38]. Use cases represent an important aspect of big data innovation and digitalization. A scoring model is used to generate data-driven prioritized use cases to achieve a desired business value [34,39]. In big data, use case prioritization is necessary to discover use cases with significant potential impact on an organization [34]. A data-based scoring model quickly presents data related to a use case, permitting data scientists to decide whether to proceed with a case in the next release with minimal potential risks.

Moisiadis provided an explicit answer to this question and indicated the need to prioritize use cases, particularly their scenarios, during the requirements engineering process [40,41] in order to expand and elaborate the most important functionalities or scenarios to the market to improve quality and reduce development time and cost, whereas less important use cases are not considered in the elaboration of the current release. The same approach was applied in [42,43], where prioritized use cases were employed early in each iteration of agile development, concentrating on the increment to be delivered, taking into account a quality delivery with minimal possible cost and development time [43].

In agile development, use case prioritization plays a significant role in the planning of iterations [44]. In agile practice, developers prioritize old and new use cases at the beginning of each iteration in backlog, allowing them to review important functionalities for the next release early if needed and to shorten agile development time between launched releases, leading to increased user satisfaction [45]. The consistency of customers' logical judgment is represented by prioritization [46]. In agile development, use cases are employed to increase team osmotic communication. Some research has highlighted the importance of representing features as prioritized use cases to improve understanding of the system being developed and its estimated effort according to use-case-driven cost, resource, and time management plans [31,38,47–50]. In Scrum, sprints are planned based on prioritized use cases. Personas are employed to identify user requirements representing use cases for sprint planning [44]. Use cases were found to be appropriate in terms abstractness to present user requirements, acting as an intermediate model between abstract persona analysis and elaborated scenario models [44].

Scenarios have assisted in prioritizing test cases using a scenario-based traceability matrix, facilitating early detection of bugs [19,38,50], and contributing to a reduction in testing and maintenance costs [31,51]. In component-based systems, use cases and their scenarios facilitated the identification of the centrality value of an identified use case node for the sake of improved component interaction management [52]. Prioritization cannot be addressed without considering requirements dependencies; the literature shows that use cases indicate dependencies between functional requirements [37]. In ontology engineering, use-case-driven ontologies are prioritized to evaluate respective commitments during the test of a workflow [53]. Use cases are used to derive use case function points (UFP), which are employed for effort estimation [54].

Researchers have reported the significance of use-case-based prioritization not only in the early stages of software development but also in consequent later stages, resulting in the development of the use-case-centric software development process, which reflects the influence of the use case concept on requirements engineering, development, testing, and change management [38]. Use case prioritization implicitly prioritizes the most critical

defects [6,18,38], which requires a finite set of test cases [55]. Prioritizing use cases is the cornerstone of the process of identifying critical classes that are most frequently used within use cases to address critical non-functional requirements such as reliability [55]. In software product line engineering, the development strategy relies on the development of reusable functionalities and test assets to support the development of a new product in the same line with minimum effort and cost [38]. In this context, use-case-driven development improves the testing process in terms of time, cost, and effort [54], as reported with respect to the classification of use-case-driven test cases that are obsolete (i.e., new tests), retestable (i.e., currently exist but require a rerun to address the impact of change), and reusable (no retests needed) [38]. Use-case-driven test case classification can also take the form of RP.

According to Kundu and Samanta [5], prioritization of use cases is necessary to address quality goals and business goals. The former are related to non-behavioral aspects indicating which scenarios of a use case are more critical than others in terms of soft characteristics. Such goals should be addressed during the software analysis phase, highlighting areas of attention for consequent design quality, implementation, and testing activities. Business goals are related to behavioral aspects indicating which scenarios of a use case and which use cases should be released for the next increment to address customer business needs for a particular value or benefit. Neither type of goal can be defined without employing prioritized use cases and their embedded scenarios.

Use case prioritization has been conducted using well-known techniques such as analytic hierarchy processing (AHP) to consider stakeholder preferences and plays a central role in mobile development [44,55,56]. A mobile application must meet frequently changing market needs based on customer preferences [56]. Therefore, this paves the way for product optimization and mobile application development with reduced costs, resources, and effort [56].

Garcia et al. have analyzed use cases that are critically required to identify the most important functionalities of any system [57]. They analyzed not only applications in the software engineering field but also other fields, such as the oil and automotive industries. In the heavy vehicle industry, use case scenario prioritization was used to provide information about the most frequent accidents with associated injury and cost data [27].

Use case prioritization is employed in data analytics to generate use-case-driven data that are used in a prioritization matrix, assisting in the scaling of the respective manufacturing processes and accelerating the knowledge innovation level to maintain the competitiveness of an organization with maximum possible benefit [28–30]. Use case prioritization can also be used for early implementation and diagnostics with a minimum cost and failure rate [28,30]. Use case classification has also contributed to the identification of critical business metadata in system requirements to maintain the quality of data for the users' benefit [58]. In manufacturing, use case prioritization in data analytics systems contributed to the selection of optimal use cases for a production that was 28% more cost-effective than traditional approaches [29]. Identifying and prioritizing use cases in the big data context supports the improvement of organizations' business processes to address their goals [30].

Owing to the difficulty of selecting a project for digitalization with respect to constraints in terms of cost, resources, and know-how, use-case-based prioritization has paved the way for the appropriate selection of projects for digitalization purposes [39,58], contributing to further improvement of business processes and the selection of the appropriate experts and resources for a high-quality project implementation [39,58,59]. In the oil and gas industry, use case prioritization was utilized to identify the most and least important factors in implementing digital technology [60].

The role of use case identification and prioritization is evolving in the domain of the Internet of Things as part of customer experience priority understanding [35]. Use case prioritization can be employed to support decision makers in increasing sales, consumer-driven optimization, marketing, and brand awareness [35].

3.2. RQ2: Reported Research on Use-Case-Based Prioritization

This research question was answered by reviewing and analyzing the content of the 40 publications induced according to the selection strategy. Resources are ordered historically in Appendix A, which shows the resource, type of publication, prioritization criteria/approach applied, size of requirements, extent of formality (formal, semiformal, or informal) scenarios considered in the prioritization, respective contributions, and limitations.

3.3. RQ3: Stakeholders Involved in Use Case Prioritization and Their Viewpoints

Moisiadis considered stakeholder goals as the driving source in prioritization. Stakeholder goals may embody the values of a customer, cost preference, technical risk mitigation, or conformity with an organization's regulations. Prioritization of viewpoints is based on the degree to which a use case satisfies or fulfils (a) business goal(s) [40,41]. Some researchers consider the viewpoint of developers viable only within use case scenario-based prioritization, for which details matter [42,45]. In ontology engineering, some researchers have highlighted the need for domain knowledge experts to indicate and assess prioritization factors [53]. If ontology is employed, then ontology domain experts are involved in prioritization, particularly in identifying competency questions [53]. Domain experts are not only involved in prioritization in ontology engineering but also in big data, as a data scientist team and domain experts contribute to the classification of data-driven use cases based on strategic criteria [28,30,34]. Therefore, skilled domain experts who are willing to learn are essential stakeholders in prioritization [57–60].

Some prioritization techniques require the modeling of scenario-based diagrams using a system sequence diagram or UFP. In this regard, a business analyst or domain expert is responsible for prioritizing use case scenarios or function points by considering respective quantitative metrics [5,37,54,55]. During requirements analysis, Wah differentiated between user and customer stakeholders [49]. The former is the end user who directly uses the system, whereas the latter is the sponsor of the project. A sponsor can also be the end user. In this regard, users and customers are those in charge of prioritizing use cases as features of interest for the next release. However, a developer or development manager evaluates the feasibility of prioritization and negotiates whether changes are required [48,49]. Synthesis of the prioritization of users and developers is essential to minimize the expectation gap between users and sponsors [50]. In many cases, business analysts and other stakeholders implement prioritization techniques in concert to prioritize use cases [2,38,46].

In agile development, particularly in Scrum, use cases or features should be prioritized, which is the responsibility of the product owner, who lists the use cases in backlog for each sprint iteration [43]. One recommendation is to establish a feature team that is in charge of understanding, analyzing, prioritizing, and validating the new features for a use case. Feature team members are responsible for specifying features, business planning, designing, software implementation, testing, project management, platform implementation, and interworking. The team also consists of representatives of the requestor of a feature or simply the customer [43]. Others who implement Scrum in their projects include analysis teams, design teams, and management teams [44].

In the mobile development context, mobile applications must be frequently updated and aligned with market needs and changes to maintain or increase customer satisfaction [56]. Therefore, researchers have considered the involvement of customers or app users as essential evaluators in prioritization sessions [56]. Although the aforementioned research works classified stakeholders, some classified them as internal and external stakeholders [39], including any individual, group, or organization that can affect the project either positively or negatively or directly or indirectly [12].

The future of use case prioritization is represented by the Internet of Things, as stakeholders such as marketers, consumers, innovators, brand strategists, executives, and technologists are still involved [35].

3.4. RQ4: Gaps and Opening Insights to Enhance the Existing Use-Case-Based Prioritization Approaches

This review highlights gaps in terms of strategically driven prioritization of use cases that stem from the absence of strategic views represented in the form of goal-oriented models. Among the forty publications included, only 7.5%, represented by three research works, considered the goals of use-case-based prioritization [31,41,46]. One study implicitly mentioned the importance of considering goals without involving them in prioritization [31], and only one employed goals which were considered to be stakeholder-driven in a scenario-based context [41]. However, goals were not methodologically elicited, analyzed, designed, and operationalized [41], indicating that current use-case-based prioritization approaches are applied without considering the rationale of prioritizing requirements or use cases for system implementation. Overcoming this gap is anticipated to contribute to decision-making support for use cases prioritization, resulting in a highly complete, correct, feasible, testable, and consistent set of use cases. Consideration of goals informs the stakeholders of prioritization implications with respect to business objectives that are aligned with systems [10,11]. A thorough understanding of strategically driven and business-process-based prioritization of use cases is a high priority prerequisite to develop a decision support system [61] because decisions cannot be prioritized independent of respective strategic goals or domain processes, both of which consider actors a source of goals and use cases [62,63]. For example, the *i** framework is a goal-oriented and actor-centric modeling approach that considers goal prioritization as a means of criticality in its strategic dependency model. Moreover, the strategic rational model in the *i** framework supports strategic scenarios to address a goal in the form of social modeling [64], providing insight to engage strategic views and pave the way to influence the respective prioritization of not only the use cases but also their associated scenarios. Further insight is provided, supporting the proposition of a 2D traceability prioritization matrix that links goals or business objectives to respective use cases. An initial view of the 2D matrix is shown in Table 3. The prioritization ranking level is the intersection of a use case with a goal. For example, Use case 2 may be highly important for Goal a, whereas Use case 1 may be moderately important for the fulfilment of Goal b. The ranking level is indicated based on the selected prioritization approach [3].

Table 3. Suggested prioritization matrix of use cases with respect to goals.

Prioritized Goal/Use Case	Use Case 1	Use Case 2	Use Case n	Use Case m
Goal 1	Ranking level				
Goal 2		Ranking level			
Goal a		Ranking level			
... ..					Ranking level
... ..		Ranking level			
Goal b					Ranking level

The matrix is anticipated to act as a validation tool upon strategically based use case prioritization. Although scenarios are minimally considered in use case prioritization in the literature, in 13 of the reviewed studies (32.5%) [5,12,14,19,27,37,38,40,41,50–52,55], as shown in Table 4, each scenario required resources, cost, effort, and a risk management plan; however the stimulation incorporates the scenario as a third dimension in the proposed matrix, highlighting another gap in the current literature, i.e., that few of the reviewed studies reported scenario-based use case prioritization approaches. Although the proposed matrix is anticipated to resolve the gap described above, another gap is still apparent in relation to the operational business processes relevant to goals.

Table 4. Classification of publications in the literature in terms of the size of the project for scenario and non-scenario use case prioritization.

Seq ID No.	Ref No. in Historical Order	Size of Use Case					
		Small		Medium		Large	
		Scenario-Based	Not Scenario-Based	Scenario-Based	Not Scenario-Based	Scenario-Based	Not Scenario-Based
1	[40]	1					
2	[41]			1			
3	[15]				1		
4	[60]						1
5	[45]						1
6	[57]				1		
7	[53]						1
8	[18]		1				
9	[5]	1					
10	[47]		1				
11	[42]		1				
12	[49]		1				
13	[51]	1					
14	[50]			1			
15	[19]	1					
16	[55]	1					
17	[58]				1		
18	[27]					1	
19	[43]		1				
20	[52]			1			
21	[65]		1				
22	[62]		1				
23	[48]		1				
24	[31]						1
25	[46]				1		
26	[35]			Not mentioned			
27	[37]					1	
28	[54]				1		
29	[34]				1		
30	[14]			1			
31	[39]						1
32	[66]				1		
33	[28]						1
34	[44]	1					
35	[56]		1				
36	[29]						1
37	[60]						1

Table 4. Cont.

Seq ID No.	Ref No. in Historical Order	Size of Use Case					
		Small		Medium		Large	
		Scenario-Based	Not Scenario-Based	Scenario-Based	Not Scenario-Based	Scenario-Based	Not Scenario-Based
38	[30]						1
39	[38]					1	
40	[59]		1				
Sum (Percentage)		6 (15%)	10 (25%)	4 (10%)	7 (17.5%)	3 (7.5%)	9 (23%)
Overall		16		11		12	
Percentage		40%		28%		30%	

In addition to the absence of a strategic view, the absence of a business view is apparent and is represented in the form of business process models, representing another gap, providing eye-opening insight into current use-case-based prioritization. Business processes are inevitably considered when prioritizing use cases, as use cases are implicitly derived from business domain processes [7,63]. In research, use cases can bridge the gap between businesses and systems through business processes [63]. A business process is designed to fulfil at least one strategic goal and to derive at least one use case [62]. Therefore, we anticipate that prioritization of goals cascades further to respective business processes, use cases, scenarios, software design, implementation, and testing. In this situation, we suggest that use-case-based prioritization be initiated by strategic goals and associated business processes, which is anticipated to enrich the description of prioritized use cases and associated scenarios to become strategically driven and process-based and to increase the prioritization consistency of use cases. Because a use case specification is rich in required information that involves identifying data entities, strategic prioritization of a use case may also influence prioritization [63], highlighting the anticipated essential role of use cases in prioritizing data entities. This process also provides insights into the necessity of obtaining an architectural blueprint of the business processes that operate in the domain, namely, business process architecture. The architecture is anticipated to aid in assessing the impact of changes in reprioritization with respect to other business-process-driven use cases [63].

We anticipate that an elaboration of business goals can support the fulfilment of business processes and associated business rules, as described by the following proposed chain relationship:

Business goals → business processes → prioritization rules → prioritized use cases → prioritized scenarios.

Business process models are rich in rules that are represented, for example, in the form of conditional or event-based gateways that can be translated into prioritization rules that are formed with respect to process-based customized criteria, which are missing in the reported literature. Each business domain has customized goals that are operationalized in terms of requirements or use cases [67], motivating specification of the above chain within a framework that structurally derives customized prioritization rules and respective prioritized use cases from goals and business process models. This process also provides insight that can support the employment of conceptual modeling techniques to represent the framework.

Regarding identified criteria for prioritization, the literature lacks a structured and comprehensive classification of prioritization criteria that direct stakeholders for appropriate decision making. The majority of the reported approaches are informally conducted, accounting for 37.5% of reviewed publications, as shown in Table 5. Regarding the size of projects applied in case studies, small projects accounted for the majority (16, corresponding to 40% of the reported literature), whereas 65.5% of studies reported in the literature were

not scenario-based, as shown in Table 4. Therefore, there is a need to propose a structured and comprehensive classification method for prioritization criteria that are customized by the derivation of goals and scenarios to pave the way for formalization considering project sizes, given the lack of formal scenario-based approaches, as reported in Table 6 (30.8%).

Table 5. Classification of studies regarding formalization.

Seq ID No.	Ref No. in Historical Order	Formalization		
		Informal	Semiformal	Formal
1	[40]		1	
2	[41]		1	
3	[15]	1		
4	[60]			1
5	[45]			1
6	[57]		1	
7	[53]	1		
8	[18]		1	
9	[5]			1
10	[47]		1	
11	[42]			1
12	[49]	1		
13	[51]		1	
14	[50]			1
15	[19]	1		
16	[55]		1	
17	[58]	1		
18	[27]			1
19	[43]	1		
20	[52]		1	
21	[65]			1
22	[62]	1		
23	[48]			1
24	[31]			1
25	[46]			1
26	[35]	1		
27	[37]		1	
28	[54]		1	
29	[34]	1		
30	[14]		1	
31	[39]	1		
32	[66]	1		
33	[28]			1
34	[44]	1		
35	[56]			1
36	[29]			1

Table 5. Cont.

Seq ID No.	Ref No. in Historical Order	Formalization		
		Informal	Semiformal	Formal
37	[60]	1		
38	[30]	1		
39	[38]			1
40	[59]	1		
Sum	40	15	11	14
Percentage	100%	37.5%	27.5%	35%

Table 6. Classification of studies regarding formalization for scenario-based approaches.

ID No.	Ref No. in Historical Order	Formalization in Scenario-Based Approaches		
		Informal	Semiformal	Formal
1	[40]		1	
2	[41]		1	
9	[5]			1
13	[51]		1	
14	[50]			1
15	[19]	1		
16	[55]		1	
18	[27]			1
20	[52]		1	
27	[37]		1	
30	[14]		1	
34	[44]	1		
39	[38]			1
Sum	13	2	7	4
Percentage	100%	15.4%	53.8%	30.8%

4. Discussion

Despite extensive coverage of requirements prioritization in the literature [3,24,25], only 40 systematic mapping studies have applied use-case-based prioritization: 18 conference proceedings, 11 journal articles, 1 webpage, 4 dissertations, 3 technical reports, 1 white paper, and 2 books, highlighting the need for continued research on the importance of use-case-based prioritization. The discussion presented below is relevant to the four research questions.

To answer the first RQ, it appears that use case prioritization is necessary for the planning of project iterations [37,38,44], determining further innovation and digitalization [34,39], minimizing risks [34,39], increasing team osmotic communication [45], reducing testing and maintenance costs through an early detection of bugs [31,50,51], appropriate selection of projects [36,57,58], estimating efforts using function points [54], and addressing organizations' processes and goals [30]. Use case prioritization supports scaling of the respective manufacturing processes and acceleration of knowledge innovation levels to maintain the competitiveness of an organization with maximum possible benefit [28–30], resulting in improved quality and reduced development cost, time, and effort [40–43], as well as further increases in customer satisfaction [45]. Use case modeling not only supports the presentation of linked functional and non-functional requirements but extends

further to show interdependencies that are necessary to indicate the impact on prioritization [68]. Use case prioritization can be employed to support decision makers in increasing sales, consumer-driven optimization, marketing, and brand awareness in mobile development [35]. Therefore, answering the first question highlights the extent to which use case modeling is necessary to support project-development-related information.

Answers to the second RQ are shown in Appendix A. Considering scenarios is essential for prioritization when each use case is elaborated into one or more scenarios [10–12]. Therefore, we analyzed 40 reviewed publications and found that only 32.5% of them were scenario-based [5,12,14,19,27,37,38,40,41,50–52,55], as shown in Table 4, indicating a lack of scenario-based prioritization approaches, with 38.5% of studies considering use case as an abstract concept, whereas use case specifications or relations were not employed in [19,27,37,52]. Paying attention to scenarios during prioritization is necessary, as each scenario contains embedded information essential to understanding and specifying system functionalities, imposing constraints, and determining objectives to fulfil what is formulated as system behavior [68]. Hence, missing a scenario in use-case-based prioritization contributes to increased incompleteness and inconsistency of system functionalities [68].

We also quantitatively analyzed the included studies in terms of the extent of formality. The results show that use-case-based prioritization approaches were either informally applied (37.5%), formally applied (35%), or semi formally applied (27.5%), respectively, as shown in Table 5, motivating researchers to develop new approaches that are either formal or semiformal for automation in order to increase prioritization efficiency in terms of reducing efforts, costs, and errors. The majority of the 13 scenario-based prioritization approaches are classified as semiformal (53.8%), whereas the rest are classified as formal (30.8%) and informal (15.4%), indicating that the majority fits with the semiformal approaches represented in the form of models or diagrams. Therefore, any further improvement in a scenario-based approach is recommended to be represented using a semiformal approach.

Regarding the size of use cases, 16 approaches have a small size, 6 of which are scenario-based; 11 are medium-sized, 4 of which are scenario-based; and 12 are large, 3 of which are scenario-based, as shown in Table 4, indicating that research efforts were directed within the small size scope of use cases. Most of the scenario-based approaches are applied for the small size use cases, indicating that research efforts in the early stage of proposing formal or informal scenario-based approaches for medium-to-large use cases.

As stakeholders' goals were among the reported prioritization criteria and although very few were identified among the reviewed studies (3 of 40; 7.5%) [41,46,63], goals require special attention as they are the rationale for indicating the importance of use cases that meet stakeholder needs [41], representing a drawback in the literature regarding the consideration of goals in prioritization (only one of the three reported studies considered goals in a scenario-based approach (33.3%) [41], whereas the remaining two were not scenario-based (66.7%)) [46,63]. Because goal fulfilment requires elaboration of alternatives, which can be mapped to scenarios [64], semiformal scenario-based prioritization approaches can be enhanced using goal-oriented approaches [64,67]. This approach can be similarly applied to the concepts of business process modeling, whereby a goal can be achieved in a business process model in different flows of activities formulated as scenarios [69], bridging the prioritization-related gap between businesses and systems [63].

To answer the third RQ regarding the involvement of stakeholders in use-case-based prioritization, some researchers reported generic stakeholders [41], and some specified their roles. For example, domain knowledge experts conduct prioritization with respect to domain competency factors [53], and data scientists are responsible for prioritizing use cases according to strategic criteria [28,30,34]. Although researchers have attempted to specify stakeholders, the business analyst and direct users appear to be the primary stakeholders in all situations, collaboratively implementing prioritization, whereas managers and developers may contribute to the evaluation of prioritization feasibility to reduce the expectation gap [48–50]. In agile development, the product owner, along with users,

prioritizes use cases in each iteration [43]. In order to meet market needs, marketers, consumers, innovators, brand strategists, executives, and technologists contribute to use case prioritization [35] in order to increase customer satisfaction.

The answer to the fourth research question is summarized by an analysis of the gap in the reported studies in relation to the consideration of goal-based and actor-oriented approaches to prioritization [64]. Such approaches entail the absence of associated business processes that embody use case scenarios, as business process models are designed to visualize main workflows considering alternative paths [9,62]. Prior to obtaining business process models, the process architecture must be presented in order to assess the impact of anticipated prioritization on interrelated processes that embody use case scenarios. This process architecture must also consider an organization's policies, procedures, and protocol governance. In this study, we report a clear absence of customized prioritization criteria that are described as business rules derived from business processes, which are sources of use cases [63]. Another recent work proposed a hybrid model to customize priorities according to project factors [69,70].

Another recent study proposed the automatic generation of use case models [71], although the prioritization perspective remains absent from the business perspective. The origin of prioritization should be a source with a higher-level view of the business process models, namely, the business process architecture. Prioritization shows the organization and relationship among business processes before enacting software systems. Owing to the importance of prioritization in domain understanding, we recommend initiating prioritization in the early stage.

Even if goals and process models are considered, several challenges can rise during use case prioritization, such as the following:

- Stakeholder agreement: Different stakeholders may have conflicting priorities, interests, and multicriteria factors, making it challenging to arrive at a consensus agreement on prioritization. To reduce this challenge, in each iteration, a class of representative stakeholders is recommended to participate in prioritization. Agile development processes support a reduction in this challenge [72]. Approaches applied to IoT technology can be used to resolve the challenge of multicriteria factors in prioritization [73].
- Limited resources: Budget, time, and manpower are limited resources for any organization, making it challenging to determine the optimal use of these resources with competing use cases. Algorithms can be developed to determine the optimal use of resources in this regard. Such an algorithm could be used to identify desirable features for project management tools.
- Unclear business value: Quantifying and evaluating the business value of each use case is a difficult task. Prioritization becomes more difficult when there is a lack of clarity in terms of potential benefits, return on investment, and the impact on goals. However, this challenge can be reduced when the research community proposes quantitative evaluation methods to determine levels of business values for prioritized use cases.
- Changing priorities: As use case prioritization is an iterative process and owing to changes in market needs and the involvement of different stakeholders, requirements and priorities change over time. New use cases may emerge, whereas existing ones may become less relevant. Managing the change in these priorities is challenging. One proposed solution for this challenge is to tailor a change management process that aligns with the notion of the project.

5. Conclusions

In this paper, we reported on research undertaken to inform the current state of use-case-based prioritization approaches using a methodological research design based on the systematic mapping method. Only 40 research articles that addressed the orchestrated SMS research process were identified with a set of search keywords, inclusion and exclusion

criteria, and a set of four research questions to summarize attempts articulated in use-case-based prioritization applied in different fields and contexts. The analysis presented in this study classifies use case prioritization approaches into scenario-based and non-scenario-based approaches. Only 32.5% of the reviewed studies involved scenario-based prioritization approaches, about 50% of which were applied to small projects, whereas 7.5% of reviewed studies considered goals as a source for the prioritization of stakeholder use cases, indicating that the research community has made few efforts to develop such perspectives on use cases, although goals should be the contributing driver specifying prioritization criteria in order to achieve business and development objectives. Our analysis also shows that the majority of use-case-based prioritization studies involve informal specified approaches (37.5%), highlighting the limited number of efforts to develop semiformal and formal approaches.

This review highlights the need to engage goal-oriented modeling (i.e., strategic view) and business process modeling (i.e., business view), as both are anticipated to improve the communication of prioritization criteria between stakeholders. Both contribute to the creation of a common understanding without delving into the technology of software systems. The integration of both is anticipated to contribute to an increase in use case prioritization consistency, as they are both goal-driven and utilized to educate stakeholders. The present study also highlights the lack of a classification of prioritization criteria. For example, criteria can be classified into use-case-related (behavioral or non-behavioral), development-plan-related (e.g., resource availability, time, cost, etc.), or externally related factors (e.g., policies, regulations, and standard-related). This classification paves the way for the specification of comprehensive prioritization criteria.

This study highlights another drawback of failing to consider organizational policies and procedures with implicitly or explicitly embedded stakeholder needs and prioritization criteria. Risks were identified as necessary missing criteria. Using data science and analytics, risks can be identified by data-driven approaches, drawing attention to the necessity of including them in use case prioritization. One limitation of the present study is related to reporting the traceability impact of use case prioritization relative to dependent detailed system models, such as class models, sequence diagrams, and state diagrams.

Author Contributions: Y.O.: conceptualization, methodology, formal analysis, investigation, resources, data curation, writing—original draft, visualization, supervision, project administration, and funding acquisition. N.A.-S.: conceptualization, validation, investigation, resources, and writing—review and editing. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data presented in this study are available in the article.

Acknowledgments: The authors are grateful to Philadelphia University, Amman, Jordan, and Uruk University, Baghdad, Iraq, for research support.

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

Appendix A

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/ Limitations/ Weaknesses
1	IEEE Xplore Research Gate (Conference)	[40]	<p>Criteria were applied at the use case scenario level by indicating the following:</p> <ul style="list-style-type: none"> Total number of entities used in the use case; Total number of actors used in the use case; Frequency of a particular entity in the use case; Frequency of a particular actor in the use case; Frequency of entity usage in a particular scenario in the use case; Frequency of actor usage in a particular scenario in the use case. <p>Stakeholders' goals are considered in prioritization using the 'analytic hierarchy process' (AHP) approach.</p>	Small	Semiformal Scenario-based	<p>Enhanced the elicitation of entities for UML modeling.</p> <p>Supports forward and backward stakeholder-centric traceability for derivation of the most important scenarios, supporting change management.</p>	<p>Time-consuming approach to prioritize each single scenario.</p> <p>Criteria are not comprehensive.</p>
2	Google Scholar IEEE Xplore Research Gate Semantic Scholar (Conference)	[41]	<p>Applies a semi-quantitative method based on total quality management (TQM) in a two-tier prioritization framework.</p> <p>Tier 1: Determine the rank of use cases in the use case model by aggregating the rating values of stakeholders, business goals, and usage metrics.</p> <p>Tier 2: Identify the most important scenarios of the important use cases that satisfy stakeholder and product needs.</p>	Medium	Semiformal Scenario-based	<p>Addresses business goals and reduces risks in software design and implementation.</p> <p>Supports rationalization of each use case with respect to stakeholders, business goals, and usage metrics.</p>	<p>Few software organizations are willing to adopt this formal tool, as it highly depends on human decision making and heuristics.</p> <p>The technique is not well-structured.</p>
3	Google Scholar IEEE Xplore ACM Semantic Scholar Academia (Journal)	[15]	<p>Use cases prioritization is conducted from a user perspective according to the steps outlined below:</p> <ol style="list-style-type: none"> Select the use cases with the highest priority based on simple statistical calculations; Track the use case scenarios that achieve the associated goals through documentation. 	Medium	Informal Not scenario-based	<p>Guides reviewers with respect to all kinds of software inspection artifacts by finding the most important problems from a user's perspective.</p> <p>Facilitates resource-scheduling activities, as this guide can be considered to identify the most important problems in requirements engineering, design, implementation, and inspection.</p>	<p>The relationships between use cases and actors are not considered.</p>
4	Google Scholar Research Gate Semantic Scholar (Technical Report)	[60]	<p>Developers apply user-centric prioritization criteria to use cases based on:</p> <ul style="list-style-type: none"> The frequently used use cases; Time needed to gather the respective data in the selected use cases; The significance of results after implementation; Significance of data sources in each use case; Associated risks identified in relation to components in the software architecture, where each component is related to a use case. 	Medium	Formal Not scenario-based	<p>The CICE tool enables effective prioritization coordination between developers and users and facilitates early risk management during project planning.</p> <p>Enriched ontology implementation of an informational model representation by considering prioritization as constraints in the ontology.</p>	<p>A complex approach. Prioritization is highly dependent on developers' and end users' subjective personal capabilities.</p>

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/Limitations/Weaknesses
5	Google Scholar Academia.edu Research Gate (Technical Report)	[45]	<p>Use cases are prioritized based on the most important user-oriented interactions indicated by:</p> <ul style="list-style-type: none"> • The frequency of interaction; • The duration of data gathering; • The form of the results. <p>The approach is applied in domain ontology prior to interface development.</p>	Large	Formal Not scenario-based	<p>Permits early indication of use case quantities that should be implemented, along with associated risks before interface design and development, contributing to improved risk management.</p> <p>No prior domain knowledge is required to prioritize clients' needs in the form of use cases.</p>	<p>Prioritization is highly dependent on developers' perspectives and the influence of data and its sources.</p> <p>Criteria are not comprehensive.</p> <p>The relationships between use cases and actors are not considered.</p>
6	Google Scholar Springer Research Gate Semantic Scholar (Conference)	[57]	<p>Prioritization is based on analysis of a structured use case model. Criteria values result from computation of criticality factors including but not limited to:</p> <ul style="list-style-type: none"> • Safety; • Security; • Infrastructure; • Frequency; • Revenues; • Availability; • Reliability. 	Medium	Semiformal Not scenario-based	<p>The prioritized analyzed use cases provide valuable early information related to design, implementation, and testing.</p> <p>The use case relationships (included and extended) are considered in prioritization.</p>	<p>Prioritization of critical factors is highly dependent on knowledge experts.</p> <p>Many factors are subjective.</p> <p>Some are hard to predict during the early stage of system development.</p>
7	Google Scholar Research Gate Academia (Conference)	[53]	<p>A new context of use cases is introduced. Use cases are described as "paths of knowledge" to specify the respective knowledge that should be encoded in the ontology.</p>	Large	Informal Not scenario-based	<p>An ontology-oriented use-case-driven model bridges the gap between the users (i.e., who requires the ontology) and the ontology developers.</p> <p>Facilitates user-centric ontology building.</p>	<p>Highly dependent on the complexity of the ontology being built, as well as the skills and experience of involved stakeholders.</p> <p>Understanding use case concept in ontology development context is challenging and not well-structured.</p> <p>No clear prioritization technique/criteria are presented.</p> <p>The relationships between use cases and actors are not considered.</p> <p>No use case model is used for demonstration.</p>
8	Google Scholar ACM IEEE Xplore Elsevier Research Gate Academia (Conference)	[18]	<p>Expert-based use case prioritization performed by domain application experts according to their importance the user viewpoint. Then, test cases are generated accordingly.</p>	Small	Semiformal Not scenario-based	<p>Reduced defects in the software product.</p> <p>Employed in validation and verification to control the number of generated test cases.</p>	<p>No use case model is used for demonstration.</p> <p>Importance criterion needs further elaboration. No clear prioritization criteria are presented.</p> <p>Limited involvement of stakeholders, with domain experts only consulted during prioritization.</p>

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/ Limitations/ Weaknesses
9	Google Scholar IEEE Xplore Academia.edu Research Gate Semantic Scholar (Conference)	[5]	<p>The path density and criticality of use case scenarios are prioritized based on:</p> <ul style="list-style-type: none"> • Subpath length; • Commonality with other scenarios paths; • Number of repeated subpaths in other paths of a scenario. <p>This technique consists of three steps:</p> <ol style="list-style-type: none"> 1. Each use case description is designed according to a template that shows all possible use case scenarios. One use case contains at least one scenario; 2. Each use case scenario is converted into a system sequence diagram (SSD) that shows the message flow between actors and the system; 3. All SSDs are converted into one scenario graph (SG); 4. Scenarios are prioritized on the SG: <ol style="list-style-type: none"> a. Convert the SG into a tree; b. Calculate the weight of each edge; c. Calculate weight of scenarios to generate a prioritization metric. <p>High scenario cost (SC) values indicate the most important/critical paths. The calculation of SC is independent of the software analyst.</p>	Small	Formal Scenario-based	<p>Improves the SDLC quality by paying more attention to the scenarios that are involved in design, implementation, and testing phases early in the process.</p> <p>Reduces disagreement between involved stakeholders in prioritizing the use cases, as the approach is described as a structured formal analytical technique that is free of human heuristics. Therefore, the need for domain analyst experts during prioritization is low.</p> <p>Determined early during the analysis phase by the cost of each scenario and associated effort in each use case, paving the way for automation.</p>	<p>The design of the SSDs of all considered use cases for prioritization is time- and effort-consuming.</p> <p>Prioritization weight is absent in use case relations (extended and included).</p> <p>Prioritization weight is absent among use case actors (primary and secondary) and stakeholders.</p> <p>Criteria are not comprehensive.</p>
10	Google Scholar Research Gate IEEE Xplore (Conference)	[47]	<ul style="list-style-type: none"> • Categorization of a fixed set of use cases based on master–transaction, transaction–transaction, master–reports, and transaction–reports; • Design use cases precedence diagrams (UCPDs) from the developer’s perspective; • Requirements are prioritized based on their ease of construction. 	Small	Semiformal Not scenario-based	<p>Prompts the implementation phase due to the consideration of the developer’s viewpoint.</p> <p>The use case relationships (included and extended) are considered in prioritization.</p>	<p>Other stakeholders’ viewpoints are not considered.</p> <p>Prioritization is limited to a fixed set of use cases.</p> <p>Prioritization criteria are not comprehensive.</p>
11	Google Scholar Springer Research Gate (Journal)	[42]	<p>The approach has two main phases:</p> <ol style="list-style-type: none"> 1. Prioritize use cases using a precedence diagram and extract rules proposed to count unadjusted function points (UFPs) using UML class diagrams; 2. Estimate effort using UFP and COCOMO’s effort adjustment factor (EAF). 	Small	Formal Not scenario-based	<p>Contributes to early prediction and estimation of efforts in incremental software development processes.</p>	<p>Actors and use case relations are not considered in prioritization.</p> <p>No clear prioritization criteria are presented.</p>

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/Limitations/Weaknesses
12	Google Scholar Springer Academia.edu Research Gate ACM (Journal)	[49]	<p>Prioritization criteria are:</p> <ul style="list-style-type: none"> • Importance to customer; • The extent to which feasibility is addressed for organization. 	Small	Informal Not scenario-based	Supports traceability requirements and their dependencies.	Criteria are not comprehensive and not well-specified. Actors and use case relations are not considered in prioritization.
13	Google Scholar IEEE Xplore ACM Springer Research Gate (Conference)	[51]	<ul style="list-style-type: none"> • Scenarios in a use case are represented using UML activity diagrams. Each path in the activity diagram represents one scenario of a use case; • Prioritization is applied using simple mathematical formulas; • Each activity diagram is converted into a tree using depth-first search traversal steps: <ol style="list-style-type: none"> 1. Convert the activity diagram into a tree structure; 2. Prioritize scenarios in the tree based on: <ol style="list-style-type: none"> a. Weights of nodes: each node is assigned a weight between one and three based on the complexity and possibility of occurrence of defects; b. Weights of edges: each edge is assigned a weight based on the number of incoming dependencies and the number of outgoing dependencies of its node; c. Each path (scenario) is assigned a weight that equals the sum of the weights of its nodes and edges; d. The highest-priority scenario is that with the lowest-weight path. <p>The criteria are based on the lowest calculated weight path in a scenario in a single use case.</p>	Small	Semiformal Scenario-based	Finds critical scenarios to assisted in highly accurate estimation of the effort required for planning in software design, implementation, and testing. Contributes to improved software quality.	<p>Overall prioritization of use cases is not considered, as the approach targets use case scenario prioritization only.</p> <p>Use case relations ('include' and 'extend') and actors are not considered in prioritization.</p> <p>Calculating each path weight value is time-consuming.</p> <p>The validity and reliability of assigned weights require further investigation.</p> <p>Criteria are not comprehensive.</p>

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/Limitations/Weaknesses
14	Google Scholar IEEE Xplore Research Gate ACM Semantic Scholar (Conference)	[50]	<p>The approach involves the steps listed below:</p> <ol style="list-style-type: none"> 1. Store use case diagrams as .xml files; 2. Capture data from the primitive use case diagrams; 3. Generate activity diagrams and store them as xml files; 4. Generate scenarios from the activity diagrams; 5. Assign weight to nodes depending on the complexity of the activity diagram and the possibility of defects; 6. Assign weights to edges; 7. Calculate the weight of each path based on: <ol style="list-style-type: none"> a. Actor priority; b. Customer prioritization of use cases based on importance; c. Cost; d. Effort; e. Time; <p>Weights are assigned by both customers (based on their importance) and managers (based on their technical knowledge).</p>	Medium	Formal Scenario-based	<p>Contributes to early derivation of scenario tests that aid in validating requirements and reducing failures.</p> <p>Relations between use cases (included and extended) are considered in prioritization.</p>	<p>Highly dependent on the designer's effort in capturing scenarios; any missing scenarios lead to missing requirements and undetected faults.</p> <p>The impact of assigning weights has not been investigated or evaluated to date.</p> <p>Criteria do not indicate whether they are representative or comprehensive.</p>
15	Google Scholar Research Gate (Master's Thesis)	[19]	<ol style="list-style-type: none"> 1. Generate a requirement traceability matrix with respect to use cases. Use case importance is defined by the usage frequency of a functionality in the form of high–medium–low; 2. Generate scenarios for each use case; <ol style="list-style-type: none"> 2.1 Prioritize scenarios according to importance of functionality; 2.2 Generate at least three test cases for important scenarios. 	Small	Informal Scenario-based	<p>Facilitates the testing process, as the work is applied in different testing approaches.</p>	<p>The technique cannot be performed for functional testing.</p> <p>A use case model is not presented.</p> <p>No clear identification of stakeholders.</p> <p>No clear prioritization criteria are presented. The criteria "importance" and "frequency" are not comprehensive and require further elaboration.</p>
16	Google Scholar ACM Research Gate (Conference)	[55]	<p>Proposed genetic-based algorithm technique that prioritizes test cases based on:</p> <ul style="list-style-type: none"> • Frequent usage of use cases by an average user; • Frequency event execution integrated in a sequence diagram graph (SDG); • The influence of direct/indirect prioritization on classes; • Prioritized test cases. 	Small	Semiformal Scenario-based	<p>Assists in selecting test cases that have the highest impact regarding the perceived reliability of the system. Similar advantages to those listed in No. 9.</p>	<p>The algorithm is highly time- and effort-consuming, particularly in designing the SDG.</p> <p>Prioritization weight is absent in use case relations (extend and include) and actors.</p>

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/Limitations/Weaknesses
17	Google Scholar Research Gate ACM (Journal)	[58]	<p>Use cases are prioritized from 1 to 5, where 5 represents the highest priority. Most important use cases are prioritized based on two criteria:</p> <ul style="list-style-type: none"> Stakeholder personal point of view regarding the potential business value of use cases; Implementation cost per use case. 	Medium	Informal Not scenario-based	Contributes to the derivation of structured business metadata and associated requirements that are traceable to business needs.	<p>The evaluation is not representative enough and requires further analytical study.</p> <p>The stakeholders' criteria are not clearly identified, as they are highly dependent on their personal perspectives. Business value and implementation cost are not comprehensive criteria.</p>
18	Google Scholar (Master's Thesis)	[27]	<p>Use case prioritization is conducted based on simulation of subsystems using complex mathematical computations:</p> <ol style="list-style-type: none"> Find accident statistics regarding frequent accident scenarios; Identify use cases using complex subsystems, including braking, steering, and integrated braking and steering for collision avoidance maneuvers; Generate optimal control results; Find path stability simulation results. 	Large	Formal Scenario-based	Use case prioritization and alternative flows in a sequence diagram cover all possible solutions for large accident scenarios involving trucks that were not considered by previous approaches.	<p>No use case model was used for demonstration, as this work falls in the heavy vehicle industry.</p> <p>Use case relations ('include' and 'extend') and actors are not considered in prioritization.</p> <p>Domain-dependent approach.</p>
19	Google Scholar BSC Thesis	[43]	<p>Prioritization of use cases is considered for each iteration. The prioritization criteria are:</p> <ul style="list-style-type: none"> Customer-related: <ul style="list-style-type: none"> The most relevant use cases in terms of content and timing from the customer perspective; Technically related: <ul style="list-style-type: none"> The most essential early required available use cases; The riskiest use cases that should be verified first. Relevance of implementation and verification capabilities: <ul style="list-style-type: none"> The most important use cases that should be implemented and verified first with respect to the current resources. <p>Prioritization criteria are applied by the feature team during the use case workshop:</p> <ul style="list-style-type: none"> Business priority (scale of 1–3, where 1 represents the highest priority) proposed by the person responsible for the business; Technical difficulty: a three-level scale (low/medium/high) proposed by the implementation experts; Verification impact: a three-level scale (low/medium/high) proposed by the testers; Complexity for customer: a three-level scale (low/medium/high) proposed by a representative of the customer team. 	Small	Informal Not scenario-based	<p>Improved quality of an increment in a current iteration.</p> <p>Improved quality of requirements such as accuracy.</p> <p>Prioritization is enriched with inputs from different categories of stakeholders in the use case workshop. The feature team adds a special value. The generated list of prioritized use cases provides a common understanding between different team member roles.</p>	No use case diagram shown for demonstration.

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/Limitations/Weaknesses
20	Google Scholar IEEE Xplore Research Gate Semantic Scholar (Conference)	[52]	<p>Prioritization is based on the centrality of interactions in a component-based use case scenario. Prioritization is based on two criteria derived from the component interaction graph (CIG):</p> <ul style="list-style-type: none"> • Edge degree centrality (EDC): the number of nodes connected to a given node; • Centrality measure (CM) for a scenario in a use case. The most important scenarios are the frequent interactions that occur in other scenarios. 	Medium	Semiformal Scenario-based	The approach contributes to understanding of component interaction and the planning of integration, testing, and maintenance. The approach is simple and addresses the feature level.	Not automated and fully depends on human effort. Customer role, actors, and use case relations are missing in the prioritization. Features in a use case are not clearly demonstrated. Criteria are not comprehensive.
21	Academia Research Gate (Journal)	[65]	<p>Prioritize use cases based on the requirement structure and importance values through two steps:</p> <p>In phase 1:</p> <ol style="list-style-type: none"> 1. Store UML use case diagrams in XML (extensible mark-up language) files; 2. Identify the actors and use cases and map them to a directed topological network model; 3. Find the priority list. The most important use case node is ranked based on a metric that measures the mutual influence of nodes and sorts them in ascending order. <p>In phase 2: A requirement analyzer is applied to the nodes with the same importance value using the 'analytic hierarchy process'.</p>	Small	Formal Not scenario-based	Contributes to a reduction in effort and time spent on prioritization. Enriches use-case-related documents.	Relationships between requirements and use cases are not considered. Scenarios of use cases are not considered. Criteria are not comprehensive.
22	Google Scholar Research Gate Semantic Scholar (Journal)	[6]	<p>Classify use cases into three categories based on fault importance. The fault-based prioritization categories are as follows:</p> <ul style="list-style-type: none"> • Faults in system function that are crucial and important to end users; • Faults that affect important functions that are either important and rarely used or not as important but often used; • Faults in functions that are not important and rarely used. 	Small	Informal Not scenario-based	Contributes to increased quality of a system by prioritizing its functions according to fault-prevention importance.	Use case model and its actors, as well as use case relationships, are not used for demonstration. Criteria are limited to fault categories.
23	Google Scholar Research Gate (Conference)	[48]	<p>Prioritize use cases based on their consistency and stability according to the following steps:</p> <ol style="list-style-type: none"> 1. Identify the functionalities that are the most significant and stable with the help of the development manager. Create a use case template; 2. Prioritize use cases based on the manpower relevance level (primary actor, stakeholder, etc.). 	Small	Formal Not scenario-based	Contributes to the generation of a high-level documentation guide that supports the maintenance phase. Supports consideration of resources in prioritization for improved stakeholder and resource management. Requires a complete use case model to generate a solution.	Developers' involvement in prioritization is minimal. Prioritization criteria significance and stability are not comprehensive.

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/ Limitations/ Weaknesses
24	Google Scholar IEEE Xplore Research Gate Semantic Scholar (Conference)	[31]	Prioritize use cases using neuro fuzzy systems with the support of real and artificial datasets.	Large	Formal Not scenario-based	Higher customer satisfaction according to the performance on the real dataset.	No use case model is used for demonstration. Use case relations are not considered in the prioritization. No clear prioritization criteria are identified.
25	Google Scholar Research Gate (Journal)	[46]	<p>Prioritize use cases based on the analytic hierarchy process (AHP) mechanism through the five following steps:</p> <ol style="list-style-type: none"> 1. Extract the actor and use case weight data from UC specification; then, calculate use case points (UCPs); 2. Find UCP prioritization values and prioritize requirements using a use-case-oriented requirement process; 3. Create an n*n matrix using a decision process based on the hierarchical order of use case attributes: goal, criteria, subcriteria, and alternatives; 4. Classify use cases (presented as columns) with respect to common properties (presented as rows) and perform pairwise comparison depending on customer scores on questionnaires (1–9). 5. Apply consistency verification and decide whether it is acceptable. 6. Prioritize use cases using the n*n matrix and AHP. 	Medium	Formal Not scenario-based	<p>Contributes to business goals by considering them as prioritization factors.</p> <p>Verification of use case consistency results in fewer faults than in later activities.</p>	<p>The approach is described as complex and time-consuming.</p> <p>The n*n matrix requires rework, as it depends on use case properties without considering use case relations and actors.</p> <p>The matrix data are highly dependent on customer 'questionnaires' in which other stakeholders are not considered.</p> <p>The use of a pair comparison approach is limited to small and medium systems.</p>

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/ Limitations/ Weaknesses
26	Google Scholar (Technical Report)	[35]	<p>Customer-experience-based prioritization criteria:</p> <ul style="list-style-type: none"> • Reward; • Innovation; • Service; • Facilitation; • Information and decision making. • Use case prioritization criteria: • Customer experience future state vision: <ul style="list-style-type: none"> ○ Existing pain points; ○ Behavioral preferences; ○ Empowering customer opportunities. • Empathy: <ul style="list-style-type: none"> ○ Worthiness; ○ Friction level; ○ Privacy concern. • Replacing behavior: <ul style="list-style-type: none"> ○ Pleasure; ○ Usability; ○ Interaction and learnability. • Platform: <ul style="list-style-type: none"> ○ Types of sensors required; ○ Leveraged or proprietary sensors; ○ Time; ○ Location; ○ Psychographics: <ul style="list-style-type: none"> ○ >Behavioral preferences. • Business: <ul style="list-style-type: none"> ○ Business objectives; ○ Gap analysis and new investment. 	Not mentioned	Informal Not scenario-based	The approach contributes to a rethinking of the evolution of the use concept using five digital-marketing-driven characteristics.	No use case design model used for demonstration, only providing a description of how to enhance user experience. All defined criteria are theoretical where actors and use case relations are not considered in prioritization. The approach has not yet been evaluated.
27	Google Scholar Research Gate ACM Springer (Conference)	[37]	<p>Systematic prioritization is applied to use case scenarios using: Four set-based similarity measure functions:</p> <ul style="list-style-type: none"> • Counting function (CNT); • Jaccard index (JAC); • Gower–Legendre (GOW); • Sokal–Sneath (SOK). <p>Scenario-based similarity measure functions:</p> <ul style="list-style-type: none"> • Normalized Longest Common Subsequence (NLCS); • Levenshtein distance (LEV); • Global alignment and Needleman–Wunsch (NW); • Local alignment and Smith–Waterman (SW). 	Large	Semiformal Scenario-based	Cost-effective in testing. Solves the optimization problem.	No use case model is used for demonstration. The set- and scenario-based prioritization criteria are difficult to understand.

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/ Limitations/ Weaknesses
28	IEEE Research Gate (Conference)	[54]	<ol style="list-style-type: none"> 1. Estimate the actor weight in each use case. Based on the use case points (UCPs), the estimated weights can be: <ul style="list-style-type: none"> • 1 for simple; • 2 for average; • 3 for complex. 2. Estimate a use case weight based on the number of transactions within the use case. <ul style="list-style-type: none"> • If the number of transactions <3, then the UC weight is simple; • If the number of transactions is 4–7, then the UC weight is average; • If the number of transactions >8, then the UC weight is complex. 3. Calculate the ‘unadjusted use case points’ (UCP) using an actor and use case weights; 4. Determine the technical complexity factor of the entire system that ranges between no effects (0) and large effects (5) on the entire system; 5. Apply an environmental factor weight based on the UCP classification; 6. Compare and rank the extracted UCPs for UCP-based prioritization. <p>The following criteria are used in prioritization: actor weight, use case weight, unadjusted use case point, technical complexity factor, and environmental factor.</p>	Medium	Semiformal Not scenario-based	Improved estimation of UCP by prioritized use cases for more accurate results. Use case model and relationships are considered in prioritization.	The UCP does not consider use case specification. Criteria are still not comprehensive.
29	Google (Website—Blog)	[34]	<ul style="list-style-type: none"> • A data-driven approach to prioritize use cases based on high business value, documenting them in a use case grid that includes data sources, functionalities, technology, and other organizational aspects; 	Medium	Informal Not scenario-based	Sets up “systems of innovation” and supports organizations for explorative data analysis.	Time-consuming. No use case model used for demonstration. Actors and relations between use cases are not considered in the prioritization grid. Criteria are not clearly identified.

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/Limitations/Weaknesses
30	Google Research Gate Springer (Book)	[14]	<ol style="list-style-type: none"> 1. Find the basic and alternative paths in scenarios; 2. Design the activity diagram and extract the control flow graph (CFG) and the independent paths; 3. Find the complexity of each path: <ul style="list-style-type: none"> • Use the information complexity of each node to weight each path; • Determine the path complexity; • Identify node coverage by each path; • Indicate the decision node coverage of each path; • Each path is weighted based on the weights assigned to the edges traversed by the path, considering the type of coupling between nodes connecting the edges.; • Generate the total weight of each path. 	Medium	Semiformal Scenario-based	Contributes to a reduction in the time and cost of test-suite execution. Facilitates planning and management in use-case-centric development.	Applying this approach is time-consuming, especially when accommodating other UML models. Stakeholders' preferences are not involved in the weight calculations. Criteria are not comprehensive.
31	Google Scholar Research Gate (Journal)	[39]	<ul style="list-style-type: none"> • Extract a company's business model for small to medium-sized enterprises; • Specify business ideas at different levels of detail; • Translate and specify business ideas into concrete scenarios or use cases in UML; • Cluster each use case activity that helps in finding the right experts for a focused implementation based on experts' validation decisions; • Identify, prioritize, and evaluate use cases with respect to implementation benefits and costs. 	Large scale	Informal Not scenario-based	Informs implementation readiness for business-driven projects.	Applying this approach is time-consuming and constrained by experts' points of view. The use case model is not used for demonstration. Actors and use case relationships are not considered in the approach. Implementation benefits and costs criteria are not comprehensive.
32	Google Scholar Springer Research Gate (Book)	[66]	<ul style="list-style-type: none"> • User stories are prioritized from a to-be process model based on an as-is interaction diagram; • Centric roles in the process are held by stakeholders identified for prioritization. 	Medium	Informal Not scenario-based	Bridges the gap between business goals, processes, and systems, as user stories are linked to goals and used as part of other related prioritization project activities.	No use case model is designed for demonstration. Relations between user stories are not considered in the prioritization. Prioritization criteria are not well-defined.

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/Limitations/Weaknesses
33	Google Scholar Research Gate Elsevier (Conference)	[28]	<p>Use cases are prioritized with respect to database criteria:</p> <ul style="list-style-type: none"> • Timing requirement; • Response time; • Data storage; • Benefit. <p>Use cases are prioritized with respect to the strategic criteria applied using a clustering approach based on: Benefit:</p> <ul style="list-style-type: none"> ○ Reduction in workload; ○ Financial benefits. ○ Cost/effort; ○ Initial cost; ○ Running cost. 	Large	Formal Not scenario-based	<p>Provides information about data diagnostics, scaling processes, and knowledge generation acceleration during the early phases of prototype development.</p> <p>Assists in determining the cost to implement each use case and their associated benefits.</p> <p>The output of the approach aids in planning the construction of IT infrastructure.</p>	<p>Complex approach.</p> <p>A large number of prioritized use cases might increase the overhead cost of computing power nodes.</p> <p>Definitions of the requirements and the benefits of each use case are limited to knowledge domain experts.</p> <p>A use case model is not used for demonstration.</p> <p>Actors and use case relations are not considered in the prioritization.</p>
34	Google Scholar Springer Academia.edu Research Gate Semantic Scholar (Journal)	[44]	<p>The approach is as follows:</p> <ol style="list-style-type: none"> 1. Elicit user preferences in a textual description using personas; 2. Specify scenarios; 3. Identify use cases; 4. Develop a use case diagram; 5. Prioritize use cases. 	Small	Informal Scenario-based	<p>The work demonstrates the role of a use case model and scenarios in mobile application development.</p> <p>The interaction model of use case scenarios contributes to the understanding mobile development.</p> <p>The approach requires input from different classes of stakeholders, as Scrum involves heavy communication between stakeholders.</p>	<p>Lack of clear criteria for prioritization of use cases.</p> <p>Prioritization highly depends on the Scrum stakeholder domain experts.</p>
35	Google Scholar Research Gate Springer (Journal)	[56]	<p>Analytic hierarchy process (AHP) applied to evaluate the use cases for the development of mobile apps that meet users' needs.</p> <ol style="list-style-type: none"> 1. Compute the relative importance weights of the use cases and rank them into nine ranks based on stakeholder preferences obtained through a questionnaire, where a score of nine corresponds to the highest level of importance; 2. Aggregate the relative importance weight values by calculating the mean of individual scores of evaluators. The value should range from 0 to 100. 	Small	Formal Not scenario-based	<p>Simple approach.</p> <p>Contributes to the development a tool to determine the importance of the requirements in mobile apps considering user preferences.</p> <p>The approach is stakeholder-centric and contributes to the optimization of the value of the product, development time, and project costs.</p>	<p>The use case relations are not mentioned explicitly in the approach.</p> <p>The model has only been applied to to-do list apps.</p> <p>The criteria are indicated by an importance scale that is not well-defined.</p> <p>The approach is limited and not scalable.</p> <p>Needs further evaluation with a higher number of use cases.</p> <p>A large number of evaluators with small use cases may lead to disagreement and complex communication.</p>
36	Google Scholar Research Gate Elsevier (Conference)	[29]	<p>Prioritization is mathematically performed to identify the optimal combination for each use case by calculating:</p> <ul style="list-style-type: none"> • Expected cost; • Expected benefit. 	Large	Formal Not scenario-based	<p>Supports optimal prioritization of data analytics-based use cases for large-scale manufacturing during the planning and development phases.</p> <p>The approach is 28% more cost-effective than other existing approaches.</p>	<p>The approach is complex.</p> <p>Use case actors and relations are not considered in demonstrating the prioritization.</p> <p>Cost and benefit criteria are not sufficiently comprehensive.</p>

Seq	Digital Library (Publication Type)	Use-Case-Based Prioritization Reference	Criteria/Approach Used in Prioritization	Requirements Size	Formal/Informal Scenario-Based?	Contributions/Strengths	Challenges/Limitations/Weaknesses
37	Google Scholar (Conference)	[60]	<p>This prioritization approach is based on the following factors:</p> <ul style="list-style-type: none"> Financial; Failure; Implementation of digital technology; Adaptability; Culture. 	Large	Informal Not scenario-based	The use case concept helped the oil and gas industry to investigate technologies that can be applied to oil and gas challenges.	No use case model used for demonstration. Use case actors and relations are not considered in the prioritization. Prioritization criteria are not well-defined.
38	Google Scholar White paper	[30]	<p>This technique combines two approaches:</p> <ul style="list-style-type: none"> Ideation of use cases (top-down): identify and formulate potentially beneficial use cases; Identification of data-driven use cases (bottom-up): <ol style="list-style-type: none"> Understand the data used in the strategy and what use cases can be derived from them; Prioritize use cases using a scoring model based on potentials and limitations. Prioritized use cases are employed in an analytics strategy. 	Large	Informal Not scenario-based	<p>Contributes to identifying hidden patterns and correlations that are necessary for organizations to create new opportunities and improve their existing business processes.</p> <p>Contributes to reductions in costs and the failure rate.</p>	<p>Prioritization criteria are subjective and not well-defined.</p> <p>Prioritization highly depends on the analyst's experience and skills.</p>
39	Google Scholar Research Gate Semantic Scholar (Journal)	[38]	The prioritization approach is applied by classifying use cases as essential or optional according to their variability relation.	Large	Formal Scenario-based	Supports test cases prioritization in the incremental testing of product lines. Use case model, actors, and use cases are all considered in prioritization.	<p>A complex mathematical approach is applied.</p> <p>None of the stakeholders are involved in prioritization, which may lead to missing information, as they are the source of requirements and use cases.</p> <p>Prioritization criteria are essential or optional based on variability, which is not a sufficiently representative criterion.</p>
40	Google Scholar (Master's Dissertation)	[59]	<p>A use-case-based AI road map that involves two criteria (complexity of implementation and business benefits) to identify the four prioritization categories:</p> <ul style="list-style-type: none"> Must-do use cases: required for a business base; Can-do use cases: a use case is addressed as an opportunity; Case by case: a use case can be addressed, although better results are achieved with AI. Need-to-do use cases: a use case is required to address competitiveness. <p>Stakeholders involved in prioritization include:</p> <ul style="list-style-type: none"> Business stakeholders; IT service consultants; Principal consultants of global IT service companies. 	Small	Informal Not scenario-based	<p>Various stakeholders are involved in the proposed framework that enriches prioritization of the use case. The approach contributes to:</p> <ul style="list-style-type: none"> Identification of use cases with a quick time to market. Succession in transforming small business to moderate business through AI criteria. Identifying process standards during evolution; Increasing technical flexibility. 	<p>No use case diagram is used for demonstration. Actors and use case relations are not considered in the prioritization.</p> <p>Lack of measurable AI criteria for prioritization of use cases.</p>

References

1. Perini, A.; Susi, A.; Avesani, P. A Machine Learning Approach to Software Requirements Prioritization. *IEEE Trans. Softw. Eng.* **2013**, *39*, 445–461. [CrossRef]
2. Ali, S.; Hafeez, Y.; Hussain, S.; Yang, S.; Jamal, M. Requirement Prioritization Framework Using Case-Based Reasoning: A Mining-Based Approach. *Expert Syst.* **2021**, *38*, e12770. [CrossRef]
3. Hujainah, F.; Abu Bakar, R.B.; Abdulgaber, M.A.; Zamli, K. Software Requirements Prioritisation: A Systematic Literature Review on Significance, Stakeholders, Techniques and Challenges. *IEEE Access* **2018**, *6*, 71497–71523. [CrossRef]
4. Da Silva, R.; Savić, D. Linguistic Patterns and Linguistic Styles for Requirements Specification: Focus on Data Entities. *Appl. Sci.* **2021**, *11*, 4119. [CrossRef]
5. Kundu, D.; Samanta, D. A Novel Approach of Prioritizing Use Case Scenarios. In Proceedings of the 14th Asia-Pacific Software Engineering Conference (APSEC'07), Nagoya, Japan, 4–7 December 2007; pp. 542–549. [CrossRef]
6. Medeiros, P.; Santos, D.; Travassos, G. On the representation and aggregation of evidence in software engineering: A theory and belief-based perspective. *Electron. Notes Theor. Comput. Sci.* **2013**, *292*, 95–118. [CrossRef]
7. Shishkov, B. *Business Modeling and Software Design*, 1st ed.; Springer: Berlin, Germany, 2020.
8. Kamalrudin, M.; Grundy, J.; Hosking, J. Tool Support for Essential Use Cases to Better Capture Software Requirements. In Proceedings of the 25th of the ASE'10: IEEE/ACM International Conference on Automated Software Engineering, Antwerp, Belgium, 20–24 September 2010; pp. 255–264. [CrossRef]
9. Odeh, M.; Kamm, R. Bridging the Gap Between Business Models and System Models. *Inf. Softw. Technol.* **2003**, *45*, 1053–1060. [CrossRef]
10. Jacobson, I.; Booch, G.; Rumbaugh, J. *The Unified Software Development Process*, 1st ed.; Addison Wesley: London, UK, 1998.
11. Booch, G.; Rumbaugh, J.; Jacobson, I. *The Unified Modeling Language User Guide*, 2nd ed.; Addison Wesley: Boston, MA, USA, 2011.
12. Wiegers, K.; Beatty, J. *Software Requirements*, 3rd ed.; Microsoft: Redmond, WA, USA, 2013.
13. Chung, L.; Supakkul, S. *Representing NFRs and FRs: A Goal-Oriented and Use Case Driven Approach*; Springer: Berlin, Germany, 2006; pp. 29–41.
14. Bhuyan, P.; Ray, A.; Das, M. Test Scenario Prioritization Using UML Use Case and Activity Diagram. In *Computational Intelligence in Data Mining Computational Intelligence in Data Mining*; Springer: Singapore, 2017. [CrossRef]
15. Thelin, T.; Runeson, P.; Wohlin, C. Prioritized Use Cases as A Vehicle for Software Inspections. *IEEE Access* **2003**, *20*, 30–33. [CrossRef]
16. Azzeh, M. Software Cost Estimation Based on Use Case Points for Global Software Development. In Proceedings of the 5th International Conference on Computer Science and Information Technology, Bristol, UK, 2–5 December 2013; IEEE: Amman, Jordan, 2013; pp. 214–218. [CrossRef]
17. Issa, A.; Odeh, M.; Coward, D. Software Cost Estimation using Use-Case Models: A Critical Evaluation. In Proceedings of the 2006 2nd International Conference on Information & Communication Technologies, Damascus, Syria, 24–28 April 2006; pp. 2766–2771. [CrossRef]
18. Winkler, D.; Riedl, B.; Biffl, S. Improvement of Design Specifications with Inspection and Testing. In Proceedings of the 31st EUROMICRO Conference on Software Engineering and Advanced Applications, Washington, DC, USA, 30 August–3 September 2005; IEEE: Porto, Portugal, 2005; pp. 222–230. [CrossRef]
19. Saeed, U.; Amjad, A. ISTQB: Black Box Testing Strategies Used in Financial Industry for Functional Testing. Master's Thesis, Blekinge Institute of Technology, Ronneby, Sweden, 2009. Available online: <http://bth.diva-portal.org/smash/record.jsf?pid=diva2%3A830538&dswid=-1751> (accessed on 18 August 2020).
20. Odeh, Y. Deriving a Goal-based Use Case Model from Business Goals and Process Model. In Proceedings of the ICIME 2018: 2018 10th International Conference on Information Management and Engineering, Salford, UK, 22–24 September 2018; ACM: New York, NY, USA, 2018; pp. 16–20. [CrossRef]
21. Majthoub, M.; Odeh, Y.; Hijawi, M. Non-Functional Requirements Classification for Aligning Business with Information Systems. In Proceedings of the ICBDM 2020: 2020 International Conference on Big Data in Management, Manchester, UK, 15–17 May 2020; ACM: Manchester, UK, 2020; pp. 84–89. [CrossRef]
22. Aasem, M.; Ramzan, M.; Jaffar, A. Analysis and Optimization of Software Requirements Prioritization Techniques. In Proceedings of the 2010 International Conference on Information and Emerging Technologies, Karachi, Pakistan, 14–16 June 2010; IEEE: Karachi, Pakistan, 2010; pp. 1–6. [CrossRef]
23. Daneva, M.; Van Der Veen, E.; Amrit, C.; Ghasis, S.; Sikkil, K.; Kumar, R.; Ajmeri, N.; Ramteerthkar, U.; Wierings, R. Agile requirements prioritization in large-scale outsourced system projects: An empirical study. *J. Syst. Softw.* **2013**, *86*, 1333–1353. [CrossRef]
24. Yaseen, M.; Mustapha, M.A.; Ur Rahma, A.; Khan, S.; Kamal, W. Importance of Requirements Prioritization in Parallel Developing Software Projects. *Int. J. Comput. Sci. Eng.* **2020**, *8*, 171–179.
25. Jawale, B.; Patnaik, G.; Bhole, A. Requirement Prioritization using Adaptive Fuzzy Hierarchical Cumulative Voting. In Proceedings of the 7th International Advance Computing Conference (IACC), Hyderabad, India, 5–7 January 2017; IEEE: Hyderabad, India, 2017; pp. 95–102. [CrossRef]

26. Albuga, S.; Odeh, Y. Towards Prioritizing Software Business Requirements in Startups. In Proceedings of the 8th International Conference on Computer Science and Information Technology (CSIT), Amman, Jordan, 11–12 July 2018; IEEE: Amman, Jordan, 2018. [CrossRef]
27. Nozad, A. Heavy Vehicle Path Stability Control for Collision Avoidance Applications. Master's Thesis, Chalmers University of Technology, Gothenburg, Sweden, 2011.
28. Kampker, A.; Heimes, H.; Bühner, U.; Lienemann, C.; Krottil, S. Enabling Data Analytics in Large Scale Manufacturing. *Procedia Manuf.* **2018**, *24*, 120–127. [CrossRef]
29. Heimes, H.; Kampker, H.A.; Bühner, U.; Schroth, P.; Krottil, S. Prioritizing Digitalization Use Cases during Early Development Phases of Large Scale Manufacturing Systems. *Procedia Manuf.* **2019**, *28*, 37–42. [CrossRef]
30. Paradkar, S. BP Trends. 2019. Available online: <https://www.bptrends.com/use-case-identification-framework/2020> (accessed on 20 November 2020).
31. Ahmed, R.; Musleh, D.; Ahmed, M.; El-Attar, M. Use Case Prioritization Using Fuzzy Logic System. In Proceedings of the IEEE 5th International Conference on Software Engineering and Service Science, Beijing, China, 27–29 June 2014; IEEE: Piscataway, NJ, USA, 2014; pp. 149–152. [CrossRef]
32. Kitchenham, B.; Charter, S. *Guidelines for Performing Systematic Literature Reviews in Software Engineering*; EBES: New Castle, DE, USA, 2007.
33. Achimugu, P.; Selmat, A.; Ibrahim, R.; Mahrin, M. A Systematic Literature Review of Software Requirements Prioritization Research. *Inf. Softw. Technol.* **2014**, *56*, 568–585. [CrossRef]
34. Bange, C. Big Data and Digitalization Use Cases: How to Identify and Prioritize. 2016. Available online: <https://www.thedigitaltransformationpeople.com/channels/analytics/how-to-prioritise-your-big-data-analytics-use-cases/> (accessed on 20 March 2020).
35. Groopman, J. Customer Experience in the Internet of Things. Available online: https://jessgroopman.com/wp-content/uploads/2022/02/1503-Customer-Experience-in-the-IoT_Altimeter-2015.pdf (accessed on 18 June 2023).
36. Al-Moslmi, T.; Omar, N.; Abdullah, S.; Albared, M. Approaches to Cross-Domain Sentiment Analysis: A Systematic Literature Review. *IEEE Access* **2017**, *5*, 16173–16192. [CrossRef]
37. Zhang, H.; Yue, T.; Ali, S.; Liu, C. Facilitating Requirements Inspection with Search-Based Selection of Diverse Use Case Scenarios. *EAI Endorsed Trans. Creat. Technol.* **2016**, *3*, e4. [CrossRef]
38. Hajri, I.; Goknil, A.; Pastore, F.; Briand, L. Automating System Test Case Classification and Prioritization for Use Case-Driven Testing in Product Lines. *Empir. Softw. Eng.* **2020**, *25*, 3711–3769. [CrossRef]
39. Heberle, A.; Löwe, W.; Gustafsson, A.; Vorrei, O. Digitalization Canvas—Towards Identifying Digitalization Use Cases and Projects. *Univers. Comput. Sci.* **2017**, *23*, 1070–1097. [CrossRef]
40. Moisiadis, F. Prioritising Scenario Evolution. In Proceedings of the Fourth International Conference on Requirements Engineering, ICRE 2000, Karlskrona, Sweden, 19–23 June 2000; IEEE: Karlskrona, Sweden, 2000. [CrossRef]
41. Moisiadis, F. Prioritising Use Cases and Scenarios. In Proceedings of the 37th International Conference on Technology of Object-Oriented Languages and Systems, TOOLS-Pacific 2000, Sydney, Australia, 20–23 November 2000; IEEE: Sydney, NSW, Australia, 2000; pp. 108–119. [CrossRef]
42. Pow-Sang, J.; Nakasone, A.; Imbert, R.; Moreno, A. An Approach to Determine Software Requirement Construction Sequences Based On Use Cases. In Proceedings of the 2008 Advanced Software Engineering and Its Applications, Hainan, China, 13–15 December 2008; IEEE: Hainan, China, 2008; pp. 17–22. [CrossRef]
43. Vainola, K. Agile and Iterative Requirement Definition-Efficient Requirement Definition in Agile and Iterative Software Development. Bachelor's Thesis, University of Applied Sciences, Vankankälde, Finland, 2012.
44. Lopes, A.; Valentim, N.; Moraes, B.; Zilse, R.; Conte, T. Applying User-Centered Techniques to Analyze and Design a Mobile Application. *J. Softw. Eng. Res. Dev.* **2018**, *6*, 5. [CrossRef]
45. Pohl, J.; Pohl, K.; Leighton, R.; Zang, M.; Gollery, S.; Porczak, M. The ICDM Development Toolkit: Purpose and Overview. Available online: <https://digitalcommons.calpoly.edu/cadrc/25/> (accessed on 16 August 2020).
46. Park, B.; Kim, R. Making A Decision About Importance Analysis and Prioritization of Use Cases Through Comparison the Analytic Hierarchy Process (AHP) With Use Case Point (UCP) Technique. *Int. J. Softw. Eng. Its Appl.* **2014**, *8*, 89–96. [CrossRef]
47. Pow-Sang, J.; Imbert, R. *Effort Estimation in Incremental Software Development Projects Using Function Points*. *Computer Applications for Software Engineering, Disaster Recovery, and Business Continuity*; Springer: Berlin, Germany, 2012; pp. 458–465. [CrossRef]
48. Costal, D.; Javier, F.; Nuria, R.; Luis, D.; Carme, J. Introducing Use Cases in a Small Organization: An Experience and Lessons Learned. In Proceedings of the CibSE—IberoAmerican Conference on Software Engineering Steering Committee, Montevideo, Uruguay, 8–10 April 2013; pp. 26–31.
49. Wah, B. Requirements Specification and Analysis. In *Wiley Encyclopedia of Computer Science and Engineering*; John Wiley & Sons: Hoboken, NJ, USA, 2008; pp. 1–8. [CrossRef]
50. Spana, P.G.; Mohanty, H. Prioritizing Use Cases to Aid Ordering of Scenarios. In Proceedings of the 3rd UKSim European Symposium on Computer Modeling and Simulation, Athens, Greece, 25–27 November 2009; IEEE: Athens, Greece, 2009; pp. 136–141. [CrossRef]

51. Spana, P.G.; Mohanty, H. Prioritization of Scenarios Based on UML Activity Diagrams. In Proceedings of the 1st International Conference on Computational Intelligence, Communication Systems and Networks, Bhopal, India, 26–28 November 2010; IEEE: Indore, India, 2009; pp. 271–276. [[CrossRef](#)]
52. Mahmood, S.; Khan, M. A Degree Centrality-Based Approach to Prioritize Interactions of Component-Based Systems. In Proceedings of the International Conference on Computer & Information Science (ICCIS), Kuala Lumpur, Malaysia, 12–14 June 2012; IEEE: Kuala Lumpur, Malaysia, 2012; pp. 863–867. [[CrossRef](#)]
53. Missikoff, M.; Navigli, R. Applying the Unified Process to Large-Scale Ontology Building. *IFAC Proc.* **2005**, *38*, 94–99. [[CrossRef](#)]
54. Park, B.K.; Moon, S.Y.; Kimm, R. Improving Use Case Point (UCP) Based on Function Point (FP) Mechanism. In Proceedings of the 2016 International Conference on Platform Technology and Service (PlatCon), Jeju, Republic of Korea, 15–17 February 2016; IEEE: Jeju, Republic of Korea, 2016; pp. 1–5. [[CrossRef](#)]
55. Ray, M.; Mohapatra, D. A Scheme to Prioritize Classes at the Early Stage for Improving Observable Reliability. In Proceedings of the ISEC '10: 3rd India Software Engineering Conference, Mysore, India, 25–27 February 2010; pp. 69–72. [[CrossRef](#)]
56. Yildirim, O.; Peker, S. Prioritizing Use Cases for Development of Mobile Apps Using AHP. *Mob. Web Intell. Inf. Syst.* **2019**, *11673*, 308–315. [[CrossRef](#)]
57. García, J.D.; Carretero, J.; Pérez, J.M.; García, F. A Model for Use Case Prioritization Using Criticality Analysis. In *Computational Science and Its Applications—ICCSA 2004, Proceedings of the ICCSA 2004, Lecture Notes in Computer Science, 3046, Assisi, Italy, 14–17 May 2004*; Laganá, A., Gavrilova, M.L., Kumar, V., Mun, Y., Tan, C.J.K., Gervasi, O., Eds.; Springer: Berlin, Germany, 2004. [[CrossRef](#)]
58. Stock, D.; Wortmann, F.; Mayer, J. Use Cases for Business Metadata—A Viewpoint-Based Approach to Structuring and Prioritizing Business Needs. In Proceedings of the BIS 2010 International Workshop, Berlin, Germany, 3–5 May 2010; pp. 226–237. [[CrossRef](#)]
59. Seetharam, S.B. 2020 Developing a Digital AI Roadmap for Retail, Metropolia University of Applied Sciences, Helsinki. Available online: <https://core.ac.uk/download/pdf/326044700.pdf> (accessed on 20 August 2022).
60. Wishnow, D.; Azar, H.R.; Rad, M.P. A Deep Dive into Disruptive Technologies in the Oil and Gas Industry. In Proceedings of the Offshore Technology Conference, Houston, TX, USA, 6–9 May 2019. [[CrossRef](#)]
61. Nute, D.; Rosenberg, G.; Nath, S.; Verma, B.; Rauscher, H.M.; Twery, M.J.; Grove, M. Goals and Goal Orientation in Decision Support Systems for Ecosystem Management. *Comput. Electron. Agric.* **2000**, *27*, 355–375. [[CrossRef](#)]
62. Kavakli, V.; Loucopoulos, P. Goal-Driven Business Process Analysis Application in Electricity Deregulation. *Inf. Syst.* **1999**, *24*, 187–207. [[CrossRef](#)]
63. Schneider, G.; Winters, J. *Applying Use Cases: A Practical Guide*; Addison-Wesley: Boston, MA, USA, 2001.
64. Giorgini, E.; Giorgini, P.; Maiden, N.; Mylopoulos, J. *Social Modeling for Requirements Engineering*; The MIT Press: London, UK, 2010.
65. Ma, Y.; Liu, Y.; Zhang, H.; Chen, G. A hybrid method for prioritizing software requirements in terms of use cases. *J. Conver. Inf. Technol.* **2012**, *17*, 17–27.
66. Neubauer, M.; Stary, C. *S-BPM in The Production Industry—A Stakeholder Approach*; Springer: Cham, Switzerland, 2017.
67. Chung, L.; Nixon, B.; Yu, E.; Mylopoulos, J. *Non-Functional Requirements in Software Engineering*; Kluwer Publisher: Baltimore, MD, USA, 2000.
68. Supakkul, S.; Chung, L. Integrating FRs and NFRs: A Use Case and Goal Driven Approach. *Framework* **2005**, *6*, 7.
69. Ali, H.; Ahmad, A.; Ahmed, S. A Hybrid Approach for Requirements Prioritization. *Int. J. Sci. Eng. Res.* **2018**, *9*, 100–105.
70. Arshad, H.; Shaheen, S.; Khan, J. A Novel Hybrid Requirements Prioritization Approach Based on Critical Software Project Factors. *Cogn. Technol. Work.* **2023**. [[CrossRef](#)]
71. Chen, X.; Deng, Y. An Evidential Software Risk Evaluation Model. *Mathematics* **2022**, *10*, 2325. [[CrossRef](#)]
72. Borhan, N.; Zulzalil, H.; Hassan, S.; Ali, N. Requirements Prioritization in Agile Projects: From Experts' Perspectives. *J. Theor. Appl. Technol.* **2022**, *100*, 5710–5723.
73. Khan, H.; Sohail, M.; Ali, F.; Nazir, S.; Ghadi, Y.; Ullah, I. Prioritizing the Multi-criterial Features Based on Comparative Approaches for Enhancing Security of IoT Devices. *Phys. Commun.* **2023**, *59*, 102084. [[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.