Requirements Eng (2017) 22:491–526
DOI 10.1007/s00766-016-0253-7

ORIGINAL ARTICLE

Understanding requirement prioritization artifacts: a systematic mapping study

Rahul Thakurta[1]

Received: 2 January 2015 / Accepted: 2 May 2016 / Published online: 10 May 2016

Abstract The importance of prioritizing requirements stems from the fact that not all requirements can usually be met with available time and resource constraints. Efficient and trustworthy methods for prioritizing requirements are therefore in high demand. In this article, we present results of a systematic mapping study in order to appreciate the different considerations that have influenced prioritization of software requirements, identify the various types of artifacts proposed toward prioritizing software requirements, and examine certain characterizations of these artifacts. The results emphasize the heightened attention the domain of requirement prioritization has received in recent years. On the basis of this study, we are able to provide the following inferences regarding possible future research trajectories in software requirement prioritization artifacts: (1) focus on frameworks and tools; (2) emphasis on specialization; and (3) proposition of theory-based artifacts. Additional research possibilities are also pointed out at the end and are expected to stimulate further research on the topic.

Keywords Requirement prioritization · Content analysis · Systematic mapping study · Software engineering

✉ Rahul Thakurta
rahul@ximb.ac.in

[1] Information Systems Department, Xavier Institute of Management (XIMB), Xavier Square, Bhubaneswar 751 013, India

1 Introduction

The importance of requirement prioritization in software engineering has been well acknowledged [1]. Requirements are generally described as what the system is required to do along with the environment it is intended to operate in. Requirements provide description of the system, its behavior, application domain information, system constraints, specifications, and attributes [2]. The IEEE recommended practice for software requirements specifications [3] has defined and classified the requirements as external interface requirements, functional requirements, software system attributes [also known as quality requirements or non-functional requirements (NFRs)], database requirements, and an extra category comprising of derived requirements, design requirements, etc. The importance of prioritizing these requirements stems from the fact that not all requirements can usually be met with available time and resource constraints. Efficient and trustworthy methods for prioritizing requirements are therefore strongly demanded by practitioners [4].

The increased attention on prioritization given the fact that software products that are based on prioritized requirements are expected to have a lower probability of being rejected [5] has led the academic community to explore mechanisms by which prioritization of requirements can be achieved. The studies on requirement prioritization have generally focused on presenting techniques for prioritizing requirements. For example, Kukreja et al. [6] discuss a number of techniques that have been used in the industry in order to prioritize project requirements. There has been little emphasis to demonstrate the state-of-the-art artifacts on requirement prioritization. An artifact refers to a construction that "applies information technology to organizational tasks" (March and Smith [7], p. 52).

In the present research context, the artifact refers to constructions (e.g., methods, models, frameworks, systems, and tools) presented in a conceptual format (e.g., specification of the stratified analytic hierarchy method) and/or as a physical instantiation (e.g., implementation of the aforementioned method in a tool) in order to achieve prioritization of candidate requirements. In his master's thesis, Khan [8] carries out a systematic review of requirement prioritization techniques. The findings of the study describe the different techniques used for prioritizing requirements and indicate the variation in quality of the results across the studies. Herrmann and Daneva [1] present a systematic review of requirement prioritization with the focus on benefits and costs associated with project requirements. The authors propose a classification framework to analyze such studies and to chart the trajectory of progress. The results describe some of the characteristics of the requirement prioritization techniques covered in the systematic review. The authors also discuss the general limitations of these studies. Babar et al. [9] address the challenges and scopes related to requirement prioritization. The authors here stress on the need for automation of the existing techniques so that these are beneficial to the intended stakeholders. Pergher and Rossi [10] classify existing requirement prioritization contributions in terms of the type of contributions and the type of articles. The authors focus on empirical articles on requirement prioritization, which are categorized in the following eight categories: attractiveness, understandability, scalability, fault tolerance, ease of use, ease of learning, accuracy,[1] and time. The finding identifies accuracy as the key focus with respect to prioritization of requirements carried out predominately by applying suitable techniques. The authors also provide recommendations with respect to these findings. Pitangueira et al. [12] investigate the usage of the search-based software engineering (SBSE) approaches in selection and prioritization of requirements. The results indicate predominant usage of meta-heuristics in search of solutions for requirement selection and prioritization problems. The authors also discuss the gaps and trends in SBSE concerning requirement selection and prioritization. Achimugu et al. [5] perform a review of the requirement prioritization literature. The results based on selected journal, conference, and workshop papers indicate some of the limitations of the existing requirement prioritization approaches like lack of scalability, requirement dependency issues, etc. Riegel and Doerr [13] investigate the various criteria governing the requirement prioritization techniques covered in their systematic review. The authors identify a list of 280 prioritization criteria which they classify in the following six major categories, viz. (1) benefits, (2) costs, (3) risks, (4) penalties and penalty avoidance, (5) business context, and (6) technical context and requirements characteristics. Sher et al. [14] focus on analyzing how the existing prioritization techniques support the technical, business, and client aspects requiring to be addressed while prioritizing requirements. The various requirements prioritization aspects are selected to determine the current trends in the software requirements prioritization process. The initial results discussed in the article highlight lack of scalability and absence of support for business/client aspects as the primary limitations of the prioritization approaches.

The studies discussed above mostly focus on the investigation of the areas addressed in requirement prioritization literature and associated challenges, limitations, and contributions. Several observations can be put forward at this juncture. First, these existing reviews only partially cover the studies that exist today. The recent review articles that have covered a number of requirement prioritization studies given the nature of the search strategies employed in each include Achimugu et al. [5], Riegel and Doerr [13], and Sher et al. [14]. Achimugu et al. [5] include in its coverage studies describing requirement prioritization techniques till 2013 only. Riegel and Doerr [13] cover studies till 2014 with the focus on identifying requirement prioritization criteria. Sher et al. [14] also cover articles till 2014 and examine how the prevalent requirement prioritization techniques support some of the identified technical, business, and client attributes characterizing requirement prioritization. Second, the review papers do not mention the factors that have influenced the design of requirement prioritization artifacts proposed till date. These factors relate to the elements considered in the artifact designs toward satisfying the prioritization objectives. Riegel and Doerr [13] in their study discuss about the criteria that should be satisfied during prioritization of requirements, but they do not address the factors influencing prioritization. Third, excluding the article by Pergher and Rossi [10], there has been no other attempt to understand the granularity of a requirement prioritization contribution. The type of the requirement prioritization artifact can provide an explanation of how the proposed contribution addresses the research objective discussed in the concerned studies. For example, requirement prioritization constructs may indicate a new vocabulary to model a specific prioritization process. Further, the evolution of the artifacts is an indication of the trajectory of research initiatives in the domain and the maturity of the research domain. Fourth, the previous reviews have not focused on the solution characteristics of a requirement prioritization contribution, as in this

[1] Accuracy can be a measure of how much the ranking computed while using a given prioritization approach is close to the ideal target ranking (conceived as the ranking the decision maker has in mind as a result of some implicit considerations or based on negotiations) [11].

Table 1 Comparison between our study with the previous reviews

Article	Topic/purpose	Review objective	Time span	No. of included articles
Khan [8]	Systematic literature review	Describing state-of-the-art requirement prioritization techniques	Relevant studies up to 2005	8
Herrmann and Daneva [1]	Systematic literature review	Investigating requirement prioritization approaches based on benefit and cost estimation	Relevant studies up to 2007	240
Pergher and Rossi [10]	Systematic mapping study	Evolution of requirement prioritization research area, with specific focus on empirical studies	Relevant studies up to 2012	65
Pitangueira et al. [12]	Systematic literature review	Investigating search-based software engineering (SBSE) approaches for addressing requirement selection and prioritization problems	January 2001–December 2012	30
Achimugu et al. [5]	Systematic literature review	Identifying limitations, taxonomies, and processes of existing prioritization techniques	January 1996–December 2013	73
Riegel and Doerr [13]	Systematic literature review	Presenting a collection of prioritization criteria structured in six major categories and 31 subcategories	Relevant studies up to 2014	83
Sher et al. [14]	Systematic literature review	Investigating the support of the existing prioritization techniques for the technical, business, and client aspects characterizing requirement prioritization	January 1990–December 2014	115
This research	Systematic mapping study	Understanding requirement prioritization artifacts	Relevant studies up to December 2015	135

study. This is only the second systematic mapping study on requirement prioritization. The first systematic mapping study (i.e., Pergher and Rossi [10]) focused on requirement prioritization techniques as discussed above. In Table 1, we present a comparison between these studies with prior reviews undertaken on the subject to highlight the differences.

The objective of our study is to address the following four research questions:

- RQ1: What are the objectives of requirement prioritization?
- RQ2: What are the different software requirement prioritization artifacts that have been published till date?
- RQ3: What are the theoretical foundations and design characteristics of these requirement prioritization artifacts?
- RQ4: What are the factors that influence the overall requirement prioritization process?

In the first research question (RQ1), we investigate the various concerns that have guided prioritization of requirements. Some of these concerns are ordering of the identified requirements based on their importance in the project, selection of requirements for implementation from a specified master list, grouping requirements in specified categories, etc. In the second research question (RQ2), we investigate the granularity of the requirement prioritization contribution. Specifically, we identify the various types of requirement prioritization artifacts that have been proposed in the literature. Examples of such artifacts are prioritization methods, prioritization models, prioritization methodologies, etc. In the third research question (RQ3), we investigate the design foundation and the design characteristics of the prioritization process which the concerned artifact implements. With respect to the design foundation, we examine the rationale behind the prioritization process, i.e., whether the prioritization process is justified argumentatively or is based on an established theoretical core. With respect to the design characteristics, we examine the prioritization process to understand the requirement type and dependencies, and the nature of the computational mechanism (i.e., iterative vis-a-vis non-iterative process) implemented in the artifact. Finally, in the fourth research question (RQ4), we enumerate the various factors that have influenced prioritization of requirements. These factors have been classified into internal (i.e., project related) and external (i.e., prioritization environment and stakeholder related) and may influence the selection of the prioritization artifact, the prioritization process implemented in the artifact, or the results of prioritization. Examples of some of the internal factors that can influence requirement prioritization are considerations of requirement risks (i.e., an assessment of the risks associated with implementing a requirement), project development cost, project implementation dependencies (i.e., if there are assumed dependencies impacting project's requirement selection), etc. Similarly, some of the external factors that can influence requirement prioritization are stakeholder preferences (i.e., the candidate requirements for prioritization may be

specified by a stakeholder group), business goals (i.e., the prioritization of the requirements may be linked to the overall business goals), etc.

Our research contributes to theory by identification and characterization of the different artifact types and analyzing the trend to facilitate future prioritization attempts. Its contribution to practice is the delineation of the considerations influencing requirement prioritization attempts which requirement engineers might find useful. The remainder of the article is structured as follows: Sect. 2 describes the research process in detail. In Sect. 3, we present the findings from our systematic mapping study, which is subsequently discussed in Sect. 4. Finally, Sect. 5 summarizes our contribution, discusses limitations of our work, and outlines future work. We have used the term "requirements" to imply "software requirements" in the remaining sections of this article.

2 Research method

A systematic mapping study is used to map the current status of a research area to discover possible research avenues [15] and has the main goal to provide an overview of a research area, by identifying the quantity and type of research and results available within it [16]. In our mapping study, we resort to a non-experimental approach using content analysis in order to identify and analyze articles on requirement prioritization. Content analysis provides a systematic approach to describe and classify text material [17]. A review of identified articles was carried out following the guidelines specified by Higgins and Green [18], and Kitchenham et al. [19]. To start with, we developed a review protocol involving two research assistants[2] who carried out searches based on the protocol. After several refinements of the protocol, the final review protocol was specified. The review protocol addressed the following aspects: the research question(s), research strategy, search string(s), data sources, selection process, data extraction, and data synthesis. Excluding the research questions that we have already mentioned above, the other relevant processes of the protocol are described below.

2.1 Research strategy

The research strategy describes a set of constraints or criteria for the selection of articles in order to address the research questions. We employed the following criteria:

- The article addresses an issue related to requirement prioritization.
- The article has been written in English.
- The article has been published on or before 2015.
- The article describes issues that are meaningful and intuitive to follow.
- The artifact description is accessible. In case the source of the article does not describe the artifact clearly enough and in sufficient detail, we cite a more comprehensive description. In case the article is difficult to access, we rely on available descriptions of the artifact in other cited sources.
- The article has been cited by others unless it is a very recent article. We assume that a work cited and used by others is a hint of its usefulness.

2.2 Search string(s)

We used the following search strings: (1) (requirement OR requirements), (2) (prioritization OR prioritize OR prioritizing OR selection OR dependency OR management OR negotiation OR conflict), and these were concatenated using the Boolean AND operator in the search query. We had to proceed like this because no standardized, consistent terminology is used with respect to requirements prioritization. We also carried out a checking of the reference list within articles which presented some kind of discussion on prevailing requirement prioritization artifacts to ensure we did not miss out on potential articles matching the research objective. In addition, we reviewed works on requirement engineering as these may include prioritization aspects without mentioning the same in the search fields. We applied the search query on the metadata fields: title, abstract, and keywords as per the search specifications allowed by the data sources listed below.

2.3 Data sources

This refers to the channels from where the possible relevant studies were retrieved. Our data sources include the following:

1. Bibliographic databases, namely ACM Digital Library, IEEE Xplore, ScienceDirect, and Springer, which covered a number of journals, conference, and symposium proceedings in the domain of software engineering and requirements engineering.
2. Proceedings of the following conferences: Requirements Engineering: Foundation for Software Quality (REFSQ) and Systems Engineering, Test and Evaluation (SETE). The former is also provided by Springer albeit in a book format, while the latter is not listed in the databases mentioned above.

[2] Both are PhD students (one male and one female) with five or more years of work experience in information technology and with research interests in software engineering and project management.

The relevant studies were then selected based on applying the selection criteria to the candidate articles returned by running the search string queries mentioned above. In certain cases, where the origin of the concerned requirement prioritization artifact was traced to some other source like books, thesis, and websites, we also referred to those works. In Table 19 of the "Appendix," we include names of all the journals, conferences, and symposiums that have been covered in this review.

2.4 Selection process

We carried out the search individually in the identified sources between February and July 2014, and also in January 2016. We excluded editorials, prefaces, summaries of articles and tutorials, workshops, panels and poster sessions, and book chapters as these may not include enough details relevant to our research and may not be peer-reviewed. Use of the search queries resulted in over 1000 matches across the different channels. The articles returned by the search query were further screened for appropriateness. In the screening process, first the abstracts were read in order to identify whether the concerned article was addressing construction or evaluation of requirement prioritization artifact. In case the same was not clear from the abstract, we further analyzed the content of the article in search of evidence. As an exclusion criterion, we considered whether the article was outside of the software engineering domain. Based on the screening results, we were able to short-list 135 articles for full-text review. The short-listed articles address requirements prioritization and its various aspects and are based on both qualitative and quantitative research, from academicians as well as practitioners.

2.5 Data extraction

We used the Endnote software (www.endnote.com) to record reference details for each study. We synthesized relevant information from the short-listed articles based on a codebook developed by us comprising of categories and subcategories. The initial categories were derived based on the typical structure a research manuscript is expected to have, the research questions which have been identified above, and the classification basis used in Herrmann and Daneva [1] and Berander and Andrews [20]. This ensured that we were able to extract all the essential information from the short-listed studies to answer our research questions. Twelve broad categories were created and applied initially to the collected data, viz. article description (i.e., name of journal/conference/symposium, year, volume/proceeding no., no. of pages, publication date), article focus, research methodology adopted, project methodology, type of requirements prioritized, prioritization objective, number of prioritization objectives, prioritization artifact characteristic, theoretical basis of prioritization, contribution, underlying artifact on which the contribution is based, and type of instantiation.[3] Further subcategories were generated from the collected data for the above categories on review of the concerned articles. We provide in "Appendix," a brief overview of these categories which formed the basis of our data collection (Table 14).

2.6 Data synthesis

We synthesized the data by identifying patterns from the findings reported in the short-listed articles. As discussed above, twelve broad categories were identified initially in order to classify the articles under review. The two research assistants carried out the classification of the selected articles and coding independently. A spreadsheet template was created with individual columns assigned to each subcategory in order to facilitate the coding process. It was used by the research assistants to individually code the articles based on their respective judgments. We noted the number of times each subcategory was encountered in the articles that we reviewed and assigned each occurrence the same weight. The level of agreement between the two coders signifies the measure of shared, rather than individual, understanding of the content, and this is referred in the literature as inter-coder reliability [21]. This was assessed at the end of the coding process by evaluating Cohen's k coefficient. At this stage, the cases of disagreements were further discussed in the presence of a third research assistant (i.e., the author itself) who got involved in the data synthesis activity at this juncture. This ensured that either an agreement was achieved with regards the coding of the data, or a new subcategory was developed that satisfied all and met the objectives of the research. A couple of iterations involving revision of the codes were required until the final value of Cohen's k (0.82) was found to be in the acceptable range. The final step involving derivation of patterns from the coding results was carried out jointly in a workshop format by the three research assistants. Suggestions provided by Miles and Huberman [22] were referred to, in order to derive the results, which we report in the next section.

[3] Instantiation is the creation of a real instance or realization of an abstraction or template. This has been broadly classified into two types, viz. physical and conceptual (explained in the "Appendix").

Table 2 Breakup of articles retrieved from the chosen data sources

Data sources	Count	Percent
ACM Digital Library	22	16
IEEE Xplore	51	38
ScienceDirect	9	7
Springer	15	11
Requirements Engineering: Foundation for Software Quality (REFSQ)	6	4
Systems Engineering, Test and Evaluation (SETE)	1	1
Reference list of selected articles	31	23

3 Findings

3.1 Background information

3.1.1 Data source

The review includes 135 articles from the selected databases and conferences that were part of our search. The list excludes all duplications of articles reported in multiple channels. Multiple occurrences of articles were handled during the screening process by assigning such article to the first source that was found to be listing the concerned article. Any subsequent occurrence of the article in any other channel was subsequently ignored from the counting process. Table 2 gives a breakdown of where these 135 articles are published.

3.1.2 Article type

Twenty out of the total 135 articles described requirement prioritization artifacts whose original source was traced to books or book chapters (12), thesis (3), and websites (5) as discussed earlier. Out of the remaining 115 articles, 42 articles are journal articles, 67 articles are available as conference proceedings, and 6 articles are available as symposium proceedings. In Table 15 of the "Appendix," we include a list of all the articles that are part of the review.

3.1.3 Article focus

In terms of the focus of these articles, 95 articles (83 %) out of the 115 articles published in journals, conferences, or in symposium proceedings focus on issues directly related to requirement prioritization. These articles try to understand requirement prioritization in terms of existing status and challenges, present concepts used in the design of a prioritization artifact, present an artifact to arrive at the priority of requirements, and compare different prioritization artifacts. These also focus on the accuracy of the prioritization results, evaluation of the requirement prioritization artifacts, or deal with the utility and usefulness of the concerned artifact to stakeholders involved with the prioritization process or the outcome. The remaining twenty articles (17 %) focus on systems engineering, more specifically requirements engineering issues. In these articles, requirement prioritization is introduced in connection to the broader systems or requirements engineering goal(s) these articles address. Four out of these twenty articles focus specifically on systems engineering with the emphasis on managing or improving the concerned processes. The remaining sixteen articles discuss requirement prioritization in the broader context of the requirement engineering issues addressed in these articles. For example, article S11 provides a value-based approach in requirements engineering when creating product value through requirements selection for a software release, based on case studies in three different companies. While detailing on the case studies, the authors discuss the requirement prioritization process in the context of requirement selection in these three companies. Article S83 discusses handling requirement inconsistencies in distributed software requirements specifications. The authors describe a prioritized merging-based framework that allows construction of a globally prioritized requirements specification from the original requirements collections. This requirement specification is then mapped to a prioritized merging operator to identify all the consistent subsets of the requirement specification. Article S99 assesses the suitability of multicriteria decision making (MCDM) methods to support software engineers' decisions. As a part of this analysis, the authors propose a hybrid assessment method (HAM) which computes a priority measure of a set of requirements characterizing a software architectural alternative. In Table 16 of the "Appendix," we briefly indicate the focus of these twenty articles.

3.1.4 Research methodology

As in the previous case, here we also concentrate on the 115 articles published in journals, in conferences, or in symposium proceedings. In terms of the research methodology adopted, 96 out of the 115 articles are found to base their approach solely on quantitative research

methodology. Seventeen articles are found to resort to qualitative research methodology. Two articles have used both qualitative and quantitative methodology in their research. To cite some examples, Azar et al. (article S7) use quantitative research methodology to propose a value-oriented prioritization (VOP) framework for prioritizing project requirements. Racheva et al. (article S95) use qualitative research methodology to arrive at a conceptual model for understanding the inter-iteration prioritization process in agile software development. Li et al. (article S71) combine qualitative and quantitative methodologies to arrive at final priority ratings of customer requirements in product planning scenarios. A breakdown of all these articles according to research methodology is provided in Table 3.

3.1.5 Temporal view of publications

In order to understand the trend of publication, we divided the period in the following brackets: (A) articles published prior to 2000 representing the publications on requirement prioritization made in the last century and (B) articles published from the year 2000 to 2015 representing the publications made in the present century. The range 2000–2015 was further subdivided in the following way: (1) articles published between 2000 and 2005 (both years inclusive), (2) articles published between 2006 and 2010 (both years inclusive), and lastly (3) articles published from 2011 to 2015 (both years inclusive). We divided the articles published in and after 2000 in three equal durations so that we are able to comment more specifically on the trend of publications. The results shown in Fig. 1 indicate that 96 out of the 135 articles (72 %) have been published since 2006. Twenty-three articles (17 %) were published between 2000 and 2005, inclusive. Sixteen articles (12 %) were published before 2000. The figures clearly indicate that the emphasis on requirement prioritization has been more in the last ten years as compared to the initial attempts.

3.2 Understanding requirement prioritization artifacts

Here we report on how the existing literature contributes to our understanding of requirement prioritization artifacts. Figure 2 gives an overview of how the research questions integrate together to give a comprehensive view of our topic. The figure indicates that the prioritization process is guided or driven by its objectives (prioritization concerns). The end result of the design manifests as an artifact toward prioritization of the candidate requirements. The contextual factors can influence the prioritization concerns, process, or the outcome.

Prioritization objectives relate to the concerns the prioritization is expected to fulfill (RQ1). Explicating these concerns in the study is important, since the usefulness of a prioritization artifact is dependent on the purpose for what it is needed. Prioritization artifact design relates to the investigation of the design foundation and design characteristics of the prioritization mechanism (RQ3). Under prioritization artifact, we explore the different types of artifact that has been proposed toward prioritization of the requirements (RQ2). The contextual factors are the project related or external factors that have an influence on the prioritization of requirements (RQ4). We now address these research questions in more detail.

RQ1: What are the objectives of requirement prioritization?

We address the first research question (RQ1) based on 88 articles published in journals, conferences, or symposium proceedings that discuss construction or evaluation of requirement prioritization artifacts. Results from these studies indicate the various concerns that have guided prioritization of requirements. These concerns were identified based on understanding of the research objective and/or the research questions in these articles. The different concerns can be broadly classified in the following dimensions: demonstrating requirement priority, requirement selection,

Table 3 Requirement methodology

Research methodology	References[a]	# studies
Quantitative	S1, S5, S6, S7, S9, S10, S11, S12, S120, S122, S124, S14, S15, S16, S17, S18, S19, S20, S22, S23, S24, S25, S26, S28, S29, S32, S33, S34, S39, S40, S42, S43, S46, S49, S50, S51, S52, S53, S54, S55, S56, S59, S60, S61, S62, S63, S64, S70, S72, S73, S74, S75, S76, S77, S78, S79, S80, S81, S82, S83, S85, S86, S87, S88, S90, S91, S92, S96, S97, S99, S100, S102, S103, S106, S107, S108, S109, S110, S113, S114, S115, S116, S117, S118, S121, S123, S125, S126, S127, S128, S129, S130, S132, S133, S134, S135	96
Qualitative	S3, S8, S30, S31, S35, S36, S37, S44, S67, S68, S69, S89, S93, S94, S95, S98, S111	17
Mixed	S47, S71	2

[a] Please refer to Table 15 for the article Id's mentioned in this column

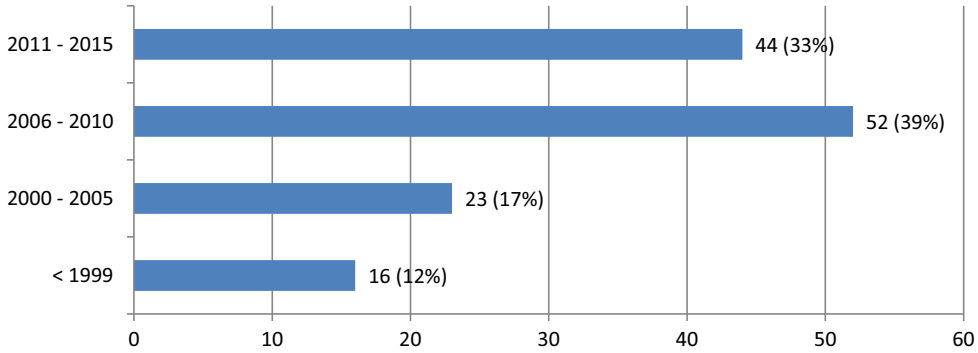

Fig. 1 Year-wise distribution of articles

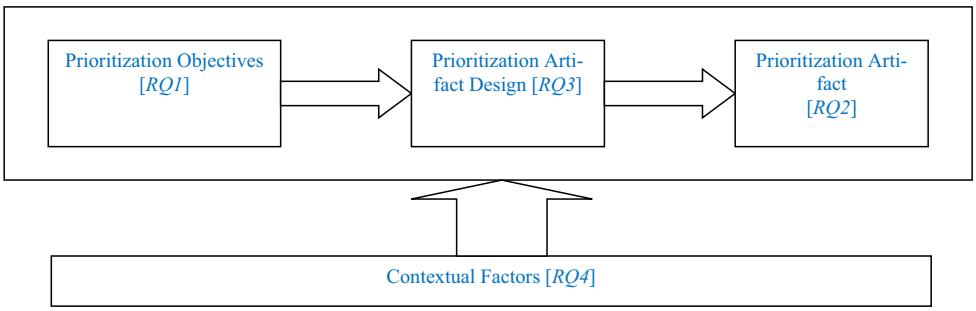

Fig. 2 Relationship between our research questions

categorizing requirements, requirement value assessment, and others. The last dimension has been introduced to address prioritization concerns, which cannot be classified under the first four dimensions. A description of these concerns is provided in Table 14 of the "Appendix," While categorizing articles in these dimensions, we have limit ourselves to what the authors have actually intended to achieve in the concerned articles. In Table 4, we present the review results for the objectives.

In three instances (i.e., S9, S72, and S124), the specification of the requirement prioritization artifact is found to be driven by multiple objectives. For example in article S72, the specification of the requirement prioritization model is aimed at minimizing the project lead time and maximizing project revenue. Table 4 indicates that the objective has been predominantly demonstration of requirement priority where the focus has been on ranking of requirements, deriving prioritized values of requirements or requirement attributes, performing feature and use case prioritization, etc. Requirement selection for allocation to a group or a phase is the next most cited objective. A description of the objectives of these 88 articles is included in Table 17 of "Appendix."

RQ2: What are the different software requirement prioritization artifacts that have been published till date?

Ninety-eight out of the 135 articles (73 %) referred to various kinds of artifacts on requirement prioritization. On certain occasions, we found the same artifact to be published in different channels by some author(s), for example an artifact first introduced in a conference or a symposium paper and then again referred in an extended journal version, etc. Ignoring such multiple contributions of the same artifact, the articles introduced 123 artifacts which again can be classified as follows (figures within braces indicate the total number of occurrences of these artifacts in all the articles we reviewed): Constructs {1}, Model {5}, Method {77}, Methodology {6}, Framework {13}, System {2}, and Tool {19} (Table 7). An explanation of these different types of artifacts has been included in "Appendix" (within Table 14). The considerations we adopted in order to classify the articles based on the type of the requirement prioritization artifact identified above are listed in Table 5. Similar arguments have also been used in the design science research categorization of artifacts [23, 24].

The first six types of artifacts (i.e., constructs, model, method, methodology, framework, and system) are representations of artifacts at a conceptual level, implying that the concerned artifact has been presented in an abstract manner. The last type of artifact (i.e., tool) indicates a physical instantiation of the artifact in concern. Hence, a tool may represent physical instantiation of a model, method, methodology, framework, or a system as explained in Table 5. The physical instantiation may be achieved by implementing the tool as an application in a suitable platform. In Table 6, we present examples of each of these artifact types from existing studies.

A point to note here with respect to some apparent contradictions in the figures pertaining to the count of the total

Table 4 Review results on requirement prioritization objectives

Objectives	References	# studies
Demonstrating requirement priority	S1, S5, S6, S7, S9, S12, S14, S17, S19, S20, S21, S23, S28, S29, S34, S39, S41, S43, S46, S47, S49, S50, S51, S59, S63, S71, S75, S76, S77, S80, S81, S82, S85, S86, S87, S88, S91, S96, S97, S99, S100, S107, S109, S110, S113, S114, S115, S117, S120, S122, S123, S127, S128, S130	54
Requirement selection	S24, S35, S36, S42, S61, S72, S73, S89, S102, S103, S124, S134, S135	13
Categorizing requirements	S26, S38, S40, S55, S119, S125	6
Requirement value assessment	S27, S33, S106, S118, S126	5
Others	S15, S16, S22, S56, S60, S74, S78, S83, S108, S133	10

Table 5 Basis of categorization of the artifacts

Artifact type	Description
Construct	The article describes vocabularies or symbols in order to reason on aspects of requirement prioritization
	The article may or may not describe other artifacts which are listed below, but these are not related to prioritization (these may address other software engineering or requirement engineering issues, etc.)[b]
Model	The article presents model which specifies some sort of relationship among established or identified constructs in its depiction of a requirement prioritization scenario
	The article may present descriptions of constructs characterizing the model which can be attributed to other sources or are not highlighted as contributions[a] of the article by the author(s)
	The article may or may not describe other artifacts which are listed below, but these are not related to prioritization (these may address other software engineering or requirement engineering issues, etc.)[b]
Method	The article describes requirement prioritization technique, activity, approach, algorithms, or practices
	The article may discuss the associated constructs or models related to the method which can be attributed to other sources or are not highlighted as contributions[a] of the article by the author(s)
	The article may or may not describe other artifacts which are listed below, but these are not related to prioritization (these may address other software engineering or requirement engineering issues, etc.)[b]
Methodology	The article discusses a methodology related to the requirement prioritization process
	The article may address descriptions of constructs, models, or methods characterizing the methodology in concern which can be attributed to other sources or are not highlighted as contributions[a] of the article by the author(s)
	The article may or may not describe other artifacts which are listed below, but these are not contributions in the requirement prioritization domain[b]
Framework	The article describes a framework associated with requirement prioritization
	The article may address descriptions of the artifacts presented above; however, these artifacts may be attributed to other sources or are not highlighted as the contributions[a] of the article in concern
	The article may or may not describe a system (listed below), but the concerned system is not a contribution in the requirement prioritization domain[b]
System	The article describes a requirement prioritization system
	The article may address descriptions of the artifacts presented above; however, these artifacts may be attributed to other sources or are not highlighted as the contributions[a] of the article in concern

[a] These if presented in a stand-alone manner may be very abstract or may not include sufficient details to be noteworthy

[b] However, if these other artifacts are also noteworthy contributions on requirement prioritization (possible when an article introduces multiple requirement prioritization artifacts which may or may not be related), these multiple artifact types are explicitly indicated alongside the article entry (please refer to Table 18 in "Appendix")

number of artifacts (123), the total number of articles under review (135), and the claim made above that the reported artifacts constitutes 73 % of the total number of articles. The source of this contradiction rests on the fact that in several articles we observed reference to more than one artifact focusing on requirement prioritization. For example, Benestad and Hannay (article S16) mention four types of requirement prioritization techniques, viz. simple dropdown, drag into bins, sortable table, and pairwise comparisons and ranking. Similarly, Gottesdiener (article S41) provides description of the following three agile requirement prioritization methods, namely weighted criteria analysis, dot

Table 6 Examples of artifact type occurrences in existing studies

Article id	Artifact type	Description
Examples of the artifact types		
S74	Construct	The article describes an extension of the traditional goal modeling notation for establishing reasoning for preferential requirements during requirements engineering
S47	Model	The article describes a new model for requirement prioritization rooted on the analytical hierarchical process (AHP)
S49	Method	The article describes the numerical assignment method toward classifying requirements into pre-defined categories based on priority ratings ascribed to each requirement
S106	Methodology	The article describes fuzzy quality function deployment-based methodology for prioritizing non-functional requirements for selection and integration with the functional requirements
S77	Framework	The article presents a framework to guide prioritization of software requirements gathered from multiple stakeholders. The representation of the framework uses a relationship matrix in order to integrate priorities from multiple perspectives and combines these in a final priority value based on computations in a number of steps
S88	System	The article describes a system called the Distributed Collaboration Priorities Tool (DCPT) in order to perform collaborative prioritization of requirements, goals, and stakeholder win conditions in projects
S88	Tool	The physical instantiation of the system discussed above is achieved through construction of the tool named Distributed Collaboration Priorities Tool (DCPT)
Examples of articles introducing multiple types of artifacts		
S16, S41	Method(s)	Each of these two articles provides independent examples of multiple requirement prioritization artifacts. Both have been briefly discussed in the paragraph above
S9	Method, Tool	The article describes the stratified analytic hierarchy method toward selection and ranking of relevant software features in a staged configuration process. The description of the method is followed by a discussion on the tool support for the method as a plug-in which can be implemented in a spreadsheet software
S33	Methodology, Tool	The article develops a systematic methodology called AMUSE (Appraisal and Measurement of User Satisfaction) that specifies feature prioritization in one of its steps. Tool-supported rating systems having graphical capabilities are then used in the demonstration of the AMUSE methodology in a case study
S82	Framework, Tool	The article describes a 6-step framework to guide prioritization of requirements with the aim of addressing the apparent weaknesses in current approaches. The specification of the framework makes use of stakeholder profile model, requirement dependencies, and requirement usage metrics which guides the derivation of the subjective and objective ratings for each requirement. The demonstration of the framework is carried out based on a requirement prioritization tool for the purpose of evaluation
S88	System, Tool	*<Discussed above>*
S100	Model, Framework	The article presents a prioritization framework to facilitate prioritization of requirements in business process-driven software development. The description of the framework introduces three different models (i.e., value model, issue model, and stakeholder/role model) that represents the different issues that needs to be considered, the criteria to be used for prioritization, and the various roles that are relevant for prioritizing the requirements

voting (use of sticky dots for allocation to requirements), and pairwise analysis. Excluding article S5, all the other articles which reported construction of a tool also elaborated on the design details of the concerned artifact.

A year-wise classification of the artifacts in Table 7 indicates that the initial efforts have been mostly on developing methods of requirement prioritization. The current effort, however, indicates a partial shift from proposing methods to developing requirement prioritization frameworks. Some recent contributions also aim at arriving at requirement prioritization models or proposing notations to reason on requirement prioritization issues. There is also an increasing interest in developing tools on requirement prioritization. In Table 18 of "Appendix," we indicate these artifacts on requirement prioritization that we covered in our review. We request the readers to consult the respective sources for details on these artifacts.

RQ3: What are the theoretical foundations and design characteristics of these requirement prioritization artifacts?

We answer RQ3 based on the 98 articles which deals with construction and/or evaluation of the artifacts introduced earlier. Table 8 provides a classification of these articles based on the theoretical foundations described below. Additionally, in Table 9, we present a year-wise breakdown of these theoretical foundations. With respect to the theoretical foundation, we focus on the justifications governing the design of the concerned artifacts. In 57 articles (Table 8), the foundation of the concerned requirement

Table 7 Year-wise classification of requirement prioritization artifacts

	Constructs	Model	Method	Methodology	Framework	System	Tool
<1999			15			1	3
2000–2005			13		3		4
2006–2010		3	19	6	5		4
2011–2015	1	2	30		5	1	8
Overall	1	5	77	6	13	2	19

Table 8 Theoretical foundation of design

Objectives	References	# studies[a]
Uses/extends previous contribution	S1, S6, S9, S12, S14, S17, S20, S22, S23, S46, S47, S48, S50, S51, S55, S71, S99, S103, S106, S108, S110, S113, S115, S117, S118, S127, S134	27
Justified by theory	S13, S14, S15, S33, S34, S42, S57, S60, S61, S71, S72, S73, S74, S87, S99, S114, S118, S130	18
Argumentative	S4, S5, S7, S16, S19, S21, S24, S26, S27, S28, S35, S36, S38, S39, S40, S41, S43, S49, S56, S58, S59, S63, S65, S75, S76, S77, S78, S80, S81, S82, S83, S84, S85, S86, S88, S89, S91, S96, S97, S100, S101, S102, S104, S105, S107, S109, S112, S119, S120, S122, S123, S124, S125, S126, S128, S133, S135	57

[a] Count is 102 as 4 articles (S14, S71, S99, S118) occur twice in the table (reasons mentioned above)

prioritization artifact has been explained argumentatively, i.e., by providing justifications based on logic. In these cases, the design of the concerned artifact has been proposed based on arguments presented by the authors, and the design is validated based on case studies, etc. There is no theoretical underpinning of the design, i.e., the design is not derived from established theoretical propositions. For example, the numerical assignment method (article S49) specifies a simple heuristic to categorize requirements based on the ratings ascribed to these requirements, and the validity of the method has been demonstrated based on case studies. There is no reference to any theoretical basis in the design of its method. In the remaining cases, either the concerned artifact uses or extends previous contribution(s) by demonstrating improvements in the process or the outcome, or the concerned artifact is justified based on established theoretical propositions (explained in Table 14 of "Appendix"). Some examples of instances where the concerned requirement prioritization artifact extends existing artifacts include the hierarchy AHP approach (article S51) which extends AHP technique in a level-wise fashion on requirements arranged in a hierarchy. Similarly, the hierarchical cumulative voting (HCV) (article S17) uses both the AHP technique and the cumulative voting (CV) technique in order to address weaknesses of these individual approaches[4] and achieve better prioritization of candidate requirements. Similarly, examples where the concerned requirement prioritization artifact is based on theoretical propositions include the minimum spanning tree matrix approach (article S60) which rests on the graph theory principles [25] to carry out prioritization of requirements. Likewise, the justification in multi-attribute utility theory (MAUT) (article S57) is based on the principles of multi-criteria decision making [26]. In four instances (e.g., articles S14, S71, S99, and S118), the proposed artifact is justified by theoretical propositions and also uses or extends previous contributions. For example, article S118 presents an algorithm for prioritizing requirements which uses AHP to determine stakeholder contributions and utility theory [27] to compute ranks.

With respect to the design characteristics of the requirement prioritization artifacts, we investigate the type of requirements considered for prioritization, the nature of dependencies among requirements, and the nature of computation used in the prioritization of artifact. Considering the type of requirements that are candidate for prioritization, 67 out of the 98 articles deal with a requirement prioritization artifact where there is no explicit mention of the requirement type. These articles describe aspects of requirement prioritization without any explicit mention of the type of requirements considered for prioritization. Of the rest, seven articles explicitly mention of prioritizing project functional requirements, features, or performance requirements. For example, article S33 discusses the AMUSE (appraisal and measurement of user satisfaction) approach that uses a prioritization methodology for selecting the most promising features in the requirements engineering phase. Artifacts dealing with prioritizing NFRs are the key concern of eight articles covered in the review. For example, article S106 describes a fuzzy quality

[4] Weakness of AHP is the number of pairwise comparisons necessary with increasing requirement size, and CV does not support hierarchical requirement structures.

Table 9 Year-wise breakdown of theoretical foundation

Objectives	<1999	2000–2005	2006–2010	2011–2015
Justified by theory (A)	2	2	5	5
Uses/extends previous contribution (B)	4	1	12	6
Uses/justified by both (A) and (B)[a]	0	0	0	4
Argumentative	9	13	16	19

[a] Note that in four instances (described above), the prioritization artifact is justified by multiple considerations

function deployment approach for determining the NFRs which are important to an organization's software selection decisions. Here the focus is exclusively on selection of NFRs, and integrating the same with project functional requirements. Article S127 discusses prioritization of both the functional requirements and the NFRs simultaneously by presenting an approach which extends the hybrid assessment method (HAM). Finally, 18 articles mentioned of prioritization artifacts with focus on requirements catering to projects implementing the agile project methodology. These articles, with the exception of article S73, do not explicitly indicate the type of the requirements considered for prioritization, but rather emphasize on the requirement prioritization mechanism considering an agile methodology of project implementation. Given this explicit indication of the project methodology for which the requirement prioritization artifact has been proposed, we chose to represent the same separately. For example, article S105 discusses the ping-pong ball approach for prioritizing requirements in projects implementing the agile methodology. The ping-pong balls represent units of one dimension for prioritization such as value, risk, or cost, and these are prioritized by the intended group of stakeholders who discuss how to allocate these balls to the items till a consensus is reached. Article S73 mentions above focus on identifying effective techniques to prioritize in order to facilitate selection of NFRs during the early stages of agile software development and assess the impacts that NFRs have on the software development process. A breakdown of these requirement types is provided in Table 10.

In order to analyze the nature of dependencies among requirements, we use an established classification taxonomy proposed by Herrmann and Daneva [1] and treat the dependencies as follows: fixed priority of requirement, requirement grouping, relative instead of absolute value, pairwise comparison, and discrete instead of continuous scale.

- Under fixed priority of requirement, each requirement's priority value is assumed to be fixed, thereby disregarding all possible dependencies among requirements. In this respect excluding one article (i.e., article S112), in all other cases the priority value assigned to the requirement is assumed to be fixed (i.e., as single value). Article S112 specifies the multi-voting prioritization approach where a single stakeholder can provide multiple ratings for the candidate requirements during the prioritization process. Examples of artifacts using fixed priority value of requirements include the numerical assignment (grouping) method (article S49) where the requirements can assume a single fixed priority value which is subsequently used to classify these requirements into pre-defined categories. In Wiegers' method (article S120), ratings ascribed on several input parameters are combined to arrive at a single priority figure for each of the candidate requirements.
- With respect to requirement grouping, we investigated whether the requirement prioritization process specified grouping of requirements into bundles during the computation process. In eleven instances, grouping of requirements could be identified toward achieving the prioritization objective. For example, in the hierarchical cumulative voting (HCV) approach (article S17), the candidate requirements are grouped at different levels of a hierarchy and prioritization is performed within the identified groups.
- Relative versus absolute value deals with the nature of representation of the data pertaining to the requirements. Absolute value implies that actual values of parameters characterizing the requirements are used (e.g., monetary values of requirements) in the prioritization process. In 18 instances, actual values of parameters characterizing the requirements are found to be used in the prioritization process. For example, in cost–benefit analysis (article S84), a trade-off analysis of actual cost of requirements versus its value is carried out in order to arrive at the results. The remaining articles refer to requirement prioritization artifacts where relative measures are used to characterize project requirements (i.e., the parameters used to describe the candidate requirements are ascribed values on a chosen scale), or the computation process is independent of the nature of representation of these parameters. For example, in the Wieger's method (article S120) mentioned above, the input parameters associated with project requirements assume relative values based on

Table 10 Requirement types

Prioritization requirements	References	# studies[a]
Features[b]	S9, S16, S33, S126	4
Functional	S122, S127	2
Performance	S81	1
Non-functional (NFRs)	S46, S73, S98, S99, S106, S113, S117, S127	8
Agile	S4[c], S13, S21, S27, S38, S41, S72[c], S73, S78, S85, S89, S97[c], S104[c], S105, S112, S119, S120[c], S133[c]	18
<No mention>	<Remaining articles>	67

[a] Count is 100 as two articles (S73, S127) occur twice in the above table (each has been discussed above)

[b] Features provide a description of the characteristics (e.g., physical attributes) in existing/desired product or service, for example, the 1-Click Ordering available in Amazon. This is different from a function which specifies what the product or the service does or is intended for

[c] The concerned artifact proposed in a requirement neutral fashion and later on referred as applicable also in agile project contexts

which the final priority value is calculated. The numerical assignment (grouping) method (article S49) can use either absolute or relative measure in its requirement classification procedure.

- Pairwise comparison relate to arriving at the relative value of requirements through pairwise comparison of relevant attributes. In 37 instances, the concerned requirement prioritization artifact resorted to pairwise comparison of requirements in order to arrive at a relative value of the requirement(s). Article S113 provides an example where the participating NFRs are compared in a pairwise manner using a matrix in order to ascertain their mutual impact.
- In case of discrete scale, the parameters characterizing the requirements are measured using a set of categories, e.g., an ordinal scale which ranks the requirements by their order of importance or a nominal scale like the values low/medium/high, or mandatory/desirable/inessential [1]. The sole use of discrete scale representation of these parameters can be noted in 32 instances. For example, in the Wieger's method (article S120), the input parameters are measured independently using a discrete scale ranging from 1 (minimum) to 9 (maximum). In the remaining cases, either the parameters assumed continuous values in the respective prioritization artifacts, or no restrictions are imposed on the scale of measurement of the concerned parameters. In the numerical assignment (grouping) method (article S49), the requirements assume values on a continuous scale toward arriving at a categorization of these requirements.

Finally, the process of prioritization used in the requirement prioritization artifacts to arrive at the results is observed mostly to be a non-iterative computation. However, in 20 instances, we observed the process to specify iteration over the computation steps in order to arrive at the final computed value. For example, the quantitative win–win approach (article S102) uses the AHP technique iteratively with an aim to balance the stakeholders' preferences related to different classes of requirements. Gupta et al. (article S43) also specify an iterative procedure based on requirement regression counts in order to finalize the requirements for selection in project increments. Table 11 provides a breakdown of the articles based on these characteristics.

RQ4: What are the factors that influence the overall requirement prioritization process?

The results of the mapping study indicate the different factors that have influenced prioritization of requirements. These factors can be internal (e.g., those pertaining to the project, process, product or service utilizing the requirements considered for prioritization) or external (e.g., prioritization environment and stakeholder-related considerations influencing the prioritization process). The contextual factors that emerged from the studies were classified in the following dimensions: requirement attributes, project attributes, process attributes, product–service attributes, subject attributes, and prioritization environment. Requirement attributes relate to the specific requirement characteristics that might influence the prioritization process. Project attributes relate to the description of the project context influencing the prioritization process. Process attributes are specific process-related characteristics having an influence on the prioritization process. Product–service attributes are attributes related to the product or the service under consideration and to be rendered by the project implementing the concerned requirements. Subject attributes relate to the stakeholder characteristics influencing the prioritization process. Finally, prioritization environment relate to the organization and market factors influencing requirement

Table 11 Artifact design characteristics

Objectives	References	# studies
Fixed requirement priority	<All excluding S112>	97
Requirement grouping	S17, S23, S51, S61, S71, S80, S82, S88, S102, S103, S125	11
Absolute value	S12, S16, S21, S22, S28, S36, S41, S7, S72, S75, S80, S81, S82, S84, S87, S106, S113, S135	18
Pairwise comparison	S1, S4, S6, S9, S12, S14, S15, S16, S21, S23, S27, S35, S41, S42, S43, S47, S48, S50, S51, S55, S56, S60, S61, S71, S77, S87, S88, S99, S101, S102, S104, S113, S115, S117, S127, S130, S133	37
Discrete scale	S1, S7, S9, S13, S16, S23, S27, S33, S35, S38, S42, S43, S46, S47, S48, S50, S51, S55, S60, S75, S82, S88, S99, S102, S104, S106, S115, S120, S128, S130, S133, S134	32
Iterative computation	S4, S6, S9, S13, S14, S15, S21, S26, S36, S41, S42, S43, S59, S87, S89, S102, S107, S115, S117, S130	20

prioritization. In Table 12, we list the factors constituting these dimensions. In the table, beside each factor we indicate the article sources from which the factor was derived. Brief description of these factors is included in Table 19 of "Appendix." Additionally, in Table 13 below, we discuss how some of these factors influence the prioritization of requirements in selected articles.

4 Discussions

Here, we discuss the findings of our systematic mapping study on requirement prioritization artifacts. As evident from Table 3, research on requirement prioritization has mostly adopted a quantitative approach, as the ultimate objective in most of the articles has been to arrive at a mechanism to compute the prioritized values of requirements. Research adopting qualitative research methodology has mostly investigated requirement prioritization issues from different perspectives like the study by Daneva et al. (article S30). The authors use interviews to understand the concepts that practitioners in a large software organization use in the prioritization process, and the practitioner's perspective of practices that are deemed good. In two instances, the use of both qualitative and quantitative research approaches can be noted as discussed earlier.

From the year-wise distribution of articles (Fig. 1), it is evident that more than half of the articles focusing on requirement prioritization have been published since 2006. The fact highlights the level of attention the domain has been receiving lately from researchers worldwide. The reason behind increased interest on the domain of requirement prioritization may be attributed to the growing complexities of systems, the nature of the projects (distributed, large-scale, etc.), and requirements possessing the attributes of being changed due to innovation, technological advancement or business growth [5].

We observed that among the articles which have focused on conceptual or physical instantiation of a requirement prioritization artifact, the predominance of methods is the largest (Table 7). This is quite intuitive and can be justified by the fact that the initial attempts sought to develop different requirement prioritization methods that satisfied different priorities and concerns. A growing interest on requirement prioritization frameworks and tools can be noted from Table 7 results. Again this seems justified in the present business environment as these frameworks can integrate multiple methods in non-repetitive or iterative process, and in the process achieve prioritization of the intended requirements. For example, in article S1, the authors present a framework to aid software engineers to perform prioritization process by combining existing techniques and approaches. The framework proposed by the authors combines existing prioritization methods (hundred-dollar, AHP, B-tree) that can be used to prioritize project requirements. Article S6 presents a Case-Based Ranking (CBRank) Framework that exploits machine learning techniques to address scalability issues in requirement prioritization. Article S7 presents a value-oriented prioritization (VOP) framework that uses the prioritization matrix proposed by Wiegers (article S120). Article S100 specifies the Business Process Requirement Engineering (BPRE) framework that integrates value model, issue model, stakeholder/role model, and a set of prioritization methods for prioritizing requirements in business process-driven software development. Article S113 integrates Wiegers' prioritization matrix in the prioritization process in the specification of a NFR prioritization framework. By integrating different methods, these frameworks can leverage on the strengths of these approaches as some of the existing studies have noted [28,

Table 12 List of factors influencing software requirement prioritization

Grouping	Subgrouping	List of factors
Internal	Requirement attributes	Requirement Type (S34, S122), Requirement Risk (S7, S30, S46, S62, S83, S113, S120, S134), Requirement Complexity (S10, S11), Requirement Importance/Priority/Preference (S12, S17, S19, S23, S38, S51, S75, S80, S82, S99, S106, S111, S113, S114, S115, S119, S123, S130), Requirement Volatility/Stability (S11, S30, S118, S122), Requirement Performance (S35), Requirement Value (S24, S50, S54, S55, S96, S113, S120, S126), Requirement Cost (S36, S43, S50, S55, S83, S113, S120, S135), Requirement Benefit (S36, S67, S111), Requirement Hierarchy (S17, S103, S130), Requirement Aspects (S134), Quality of Requirements (S71, S80, S111), Importance of Views (S17, S38, S51)
	Project attributes	Development Cost (S11, S31, S67), Development Benefit (S11, S31), Project Duration (Delivery Date) (S11, S26, S135), Effort (S30, S41, S103, S113), Project Size (S31, S78), Resources (Competencies) (S11, S24, S67, S78), Project Goal (S128), Project Costs (S87, S111, S124), Project Expectations (S61), Project Influence (S75), Project Constraints (S32, S78, S87, S102, S135), Project Contexts (S10), Application Domain (S66, S67)
	Process attributes	Requirement Dependencies (S10, S24, S30, S38, S41, S50, S58, S59, S63, S72, S82, S113, S114, S115), Requirement Trade-offs (S59, S113), Implementation Dependencies (S37), Artifact Traceability (S5), Legal Complexity/Mandate (S37), Response Time (S81), Regression Count (S43), Software Architecture & Relevant Attributes (S39, S59), Requirement Base-lining (S38)
	Product–service attributes	Technical debt (S30), System Impact (S11), Requirement Usability (S10, S11, S23, S82), Frequency of Use (S37, S122), Quality Attributes/Characteristics (S117), Penalty (S120), Harm Avoidance (S120)
External	Subject attributes	Requirement's Issuer (S11), Requirement Hits (S81), Number of Unique Users (S81), Multiple Groups (S77, S85, S110, S113, S125), Stakeholder Roles (S122), Stakeholder's Relative Importance (S43, S67, S87, S124), Stakeholder Preferences (S6, S9, S11, S18, S34, S35, S39, S40, S41, S51, S58, S63, S74, S82, S83, S102, S113, S114, S115, S123, S124, S125, S126, S128, S130, S135), Stakeholder Perspectives (S11, S18, S22, S35, S77, S87, S97, S113, S124)
	Prioritization environment	Business Values (S7, S28, S61, S78, S83, S92, S113, S126, S135), Business Goals (S9, S34, S35, S82, S110), Organization Size (S7, S94), Time to Market (S102), Competitive Benchmarking (S11, S67)

29]. The interest on requirement prioritization tools could be stemmed from the fact that even though the literature has proposed different approaches, frameworks, and models on requirement prioritization, these are not usable as the presentation of these artifacts is still more at the conceptual level. Similar concern has also been echoed in Svensson et al. [30] who observed that the prioritization of requirements in organizations is still very ad hoc and intuitive. Further, the growing interest in requirement prioritization frameworks and tools also suggests that the research area has attained more maturity as compared to the initial years [10].

Table 12 results indicate that internal factors have mostly influenced prioritization of requirements. This also confirms expectations as prioritizing requirements could be the responsibility of the project organization, and hence, project-specific issues might have assumed more weightage in arriving at the outcome of prioritization. The results of Table 10 further indicate that the requirement prioritization artifacts have mostly not tried to make any distinction regarding the type of requirements to be prioritized. However, some recent articles (e.g., articles S81 and S127) specifically mention of the type of requirements (i.e., performance requirements, functional requirements, NFRs) considered for prioritization. This indicates an emphasis on specialization with regard to specification of requirement prioritization artifacts. The earlier attempts were mostly neutral to these concerns, while the recent initiatives have tried to incorporate these considerations in the artifact designs. In Table 10, we have also provided references to prioritization artifacts catering to projects implementing the agile methodology. This can serve as pointers to practitioners seeking a suitable prioritization artifact for projects using the agile methodology.

In terms of characteristic of the requirement prioritization artifacts, we noticed the use of both discrete and continuous scale of measurements for parameters associated with the requirements in the respective prioritization artifacts (Table 11). These parameters assumed either absolute values or relative values in the prioritization process. In almost all the cases, the priority associated with the requirements assumed a fixed value (Table 11). This characterizes mostly a deterministic scenario with the level of priority assumed to be independent of any possible factors that might change randomly during the lifespan of the requirement in the assumed context.

We make the following inferences regarding the possibilities with respect to the requirement prioritization artifacts:

Table 13 Occurrence of factors in selected requirement prioritization articles

Description of factor occurrence in articles
Article S11 presents case studies which discuss requirement selection issues in three companies. The research uses interviews which uncovered the following factors influencing requirement prioritization decisions in organizations: The choice of requirements influenced by requirements issuer, preferences of stakeholder's as well as competitive benchmarking requirements. These are external factors depicting subject attributes (first two) and the project environment The design of the prioritization mechanism can be influenced by available resources and other project attributes like development cost, development benefit, project duration, etc The prioritization mechanism may involve assessment of requirement and product–service attributes like requirement complexity, requirement usability, system impact, etc
Article S38 addresses categorizing requirements based on prioritization by adopting a TQM-style process. The process involves specifying requirement preferences (requirement attribute). The categorization process also involves assessment of requirement dependencies and performing requirement baselining which are the process-related factors influencing the specified process
Article S67 describe the prevalent requirement prioritization practices in two case companies and the associated considerations. The study identified a number of factors influencing requirement priority which related to customers (subject attributes), business (prioritization environment), and implementation (process attributes) issues
Article S78 specifies a methodology for selection of requirement in different releases in agile projects. The specified methodology also accounts for uncertainty in requirements planning which are influenced by factors such as project size, project constraints, resources (project attributes), and business values (prioritization environment)
Article S82 describes a prioritization framework and tool which incorporates various aspects of prioritizing requirements. The framework makes use of the following toward guiding prioritization of requirements The stakeholder preferences of requirements and the business goals to be satisfied by the requirements, both of which are the external factors Assessment of requirement importance, requirement usability, and requirement dependencies which represents the internal factors toward derivation of the priority rating of requirements
Article S83 describes a prioritized-based merging framework toward handling inconsistencies during requirement specification. The specification of the framework involves balancing of the business values (prioritization environment) of the requirements against its cost and risk (requirement attributes). It also utilizes a prioritization viewpoint for representing preferences (subject attributes) of the associated stakeholders
Article S113 describes a 6-step framework toward guiding prioritization of NFRs. The selection of the NFRs is based on the preferences of different stakeholder groups comprising of both project organization and user representatives. These represent the subject attributes (external). The derivation of NFR priority is based on assessment of the following: Project effort, requirement dependencies, and requirement trade-offs. These are internal factors with the former representing project attribute and others are process-related attributes Requirement risk, requirement cost, requirement value, and requirement preferences which are internal factors representing relevant requirement attributes Project business values which are representations of the prioritization environment
Article S120 describes a method for assessment of requirement priority. The method uses estimates of requirement risk, requirement value, and requirement cost which are requirement attributes and penalty and harm avoidance which are product service attributes toward deriving the priority value of candidate requirements
Article S135 discusses requirement prioritization modeling for selection of requirements in different releases. The mathematical model aims to specify selection of requirements toward maximizing the business value (prioritization environment) under various constraints such as requirement cost, project duration, and other functionality-related constraints
Article S124 discusses methods for selection of competing requirements from stakeholders. The procedure takes into consideration the subject attributes like the relative importance of the stakeholders, and their preferences of requirements for selection. An additional factor considered in the selection process of requirements is the adherence to the overall project costs (project attribute)

- *Nature of artifact*: Even though the emphasis has been on designing requirement prioritization methods, the trends also indicate contributions on frameworks and tools aimed at prioritizing the candidate requirements. Table 7 indicates that other than methods, contributions of the type frameworks and tools are also prominent in the latest time horizon (i.e., 2011–2015) as compared to the initial year ranges. The same is also represented in Fig. 3 above. We expect the trend to continue wherein new frameworks are conceptualized with specified objectives that provide the intended benefits. The emphasis on the development of tools on requirement prioritization is also expected to continue. Research on these tools can focus more on the usability or the integration aspects, which has been found to be lacking in the existing attempts as observed by Pergher and Rossi [10].

- *Concentration on specialization*: Most of the contributions are neutral with regard to specifying the type of requirements to be prioritized by the concerned artifact. From Table 10 results, we can notice prioritization artifacts focusing on specific requirement types, for example functional requirements, NFRS, and performance requirements. It may also be possible to design artifacts focusing on requirements of other types like external interface requirement, database requirement, etc. In this way, it may be possible to develop prioritization artifacts which cater to various requirement types and in various contexts as hypothesized in the concerned research. It remains to be seen whether results based on artifacts designed for requirements of a specific type are more relevant as compared to results from using artifacts that do not distinguish among the requirements.
- *Theory-based artifact*: Even though it is expected that the requirement prioritization artifacts in future extends the existing contributions, we also expect to see artifacts with the prioritization rationale grounded in theory rather than being derived argumentatively. The basis of this assumption is the existing evidences of year-wise occurrence of theory justified artifacts in our review (Table 9), and the growing interest in design science research [31]. A plot of the proportion of theory-based artifacts based on Table 9 data depicts an increasing trend over the year ranges (Fig. 4), laying support to our claim. Design science research strives to create innovative artifacts with the purpose of solving practical problems and then tracing the contribution to kernel theories (i.e., theories from natural or social science related to the concerned artifact) that justifies the proposed design. Some of the advantages of a theory-based design are realization of artifacts, which are both effective and feasible, making design decisions more traceable, and avoiding unjustified selection of random features [32]. Given these benefits, more research endeavors might target realization of theory-based requirement prioritization artifacts. These artifacts are expected to address not also the specified design objectives, but also provide a theory-based rationale that justifies the design by relating the same to existing social or natural science theories.

5 Conclusions

We had set out to review the current state of requirement prioritization based on evidences disseminated in various scholastic channels. We proposed a set of four research questions in order to appreciate the developments surrounding prioritization of software requirements. We resorted to a content analysis following a review protocol in order to address our research questions and provide recommendations. The findings indicate requirement prioritization to be a lively research area with increasing number of contributions in recent years. The results describe the different factors that have influenced prioritization of software requirements and reveal the nature and characterization of the artifacts toward prioritization of project requirements. Based on the results, we are able to chart out possible areas of progress which we identify as focus on frameworks and tools, emphasis on specialization, and proposition of theory-based artifacts.

Our research offers scope of extending previous contributions in requirement prioritization research. We expect that the focus of future contributions in this domain to be aligned with the possibilities which we have elaborated in the previous section. In addition, future work might delve into the possibility of increased representation of external factors in requirement prioritization artifacts. This could be driven by the fact that there has been an increased emphasis on prioritization frameworks that are able to incorporate more structural complexities. The incorporation of the external factors can be instrumental in understanding the synergy between requirement prioritization during product or service development and its business considerations as

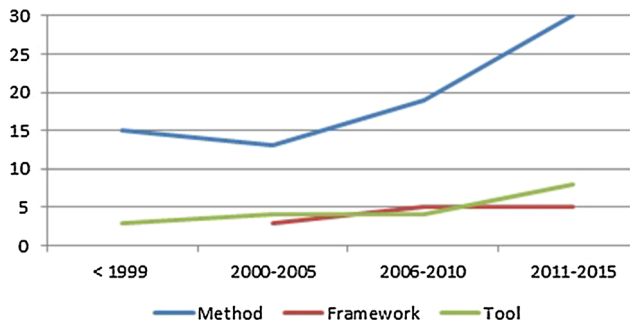

Fig. 3 Year-wise trends for selected artifacts types

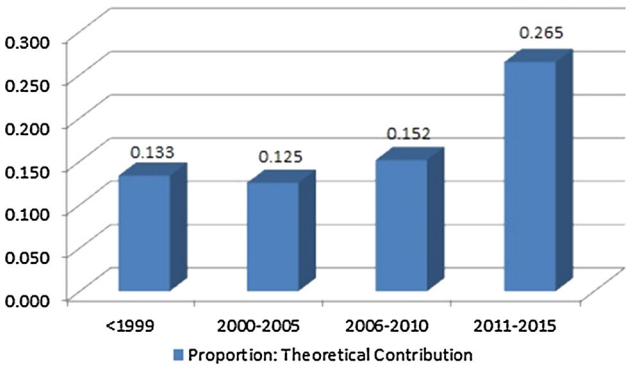

Fig. 4 Year-wise proportion of theory-based artifacts

observed by Davis [33]. We further expect that future research strives on designing requirement prioritization artifacts which adhere to multiple objectives, thereby combining or extending capabilities that characterize existing artifacts. Research possibilities also exist with respect to conceptualization of stochastic parameters in order to represent requirement priority values. This assumes that the priority parameter is able to assume a range of values depending upon the prioritization context. This provides opportunities where a single artifact can be designed to cater to specialization possibilities discussed above. Focusing on parameters, discussion on the relevance of these parameters in the design of requirement prioritization artifacts is also missing, which again can be a potential area of research. This also provides scope to investigate into the benefit accrued by using the prioritization artifact given the objective, or the accuracy rather than arriving at a hypothetical ranking of requirements. Future research may also try to correlate results of our second research question (i.e., different types of artifacts) with the other questions in order to investigate the existence of these artifacts. It may also be possible to arrive at a decision analysis framework to guide decision makers in selection of prioritization artifacts based on the objectives, process design requirement, and the related contextual factors. Finally, research opportunities exist in relation to the evaluation of requirement prioritization artifacts. This aspect has received relatively less attention as compared to research which conceptualizes or instantiates requirement prioritization artifacts. There is a need of not only evaluating the artifact in terms of function but also in terms of form where the relevance of the structural attributes of the artifact is demonstrated.

Our work is not without limitations. First, the relevance of the articles included in the review can be questioned. The selection of the databases and the conference proceedings is obviously incomplete, and hence, chances exist that we have missed out articles that are relevant to the objectives of this research. Given that journals, conferences, and symposiums have specific aims and objectives, this selection is expected to influence the results and the interpretations we make out of the analysis. Second, the sample of articles that govern our analysis is the result of using the search strings on specified identifiers in the channels that were short-listed. Even though we ensured a lot of care in framing the search query by including synonyms and related terms, it does not guarantee that the search results return all the articles associated with requirement prioritization. For example, an article using the words “feature identification” can still include prioritization as a necessary step, but such an article might not have been identified in our search if the terms used in our search query are not part of the identifiers being scanned during article retrieval process. We still expect that the chance of missing out on relevant articles is small because of inclusion of multiple identifiers in our search process. Finally, our research is not free from methodological limitations related to the sole use of content analysis of articles on requirement prioritization.

Despite these limitations, we believe that in this essay we are able to capture the present status and the possible future concerning research on requirement prioritization artifacts. We hope that this research note is able to motivate scholars in pursuing research on the topic and in the process contributing to enriching the software engineering domain.

Appendix

Tables 14, 15, 16, 17, 18, 19 and 20.

Table 14 Description of the categories

Category	Subcategory	Description
Article description	Journal/conference/ symposium	The specific journal, conference, or the symposium proceedings of which the article is part of
	Year	The year of publication of the article
	Volume/proceeding no.	The volume number in case the article has been published in a journal, or the conference/symposium proceeding indicator
	No. of pages	The length of the article
	Publication date	The specific date of publication of the article or the month in which the article was published (in case additional details are not available)
Article focus	Requirement priority	The objective of the article is specifically to arrive at a measure of priority in suitable format of the requirements set under consideration
	Requirements engineering	The objective of the article is to emphasize the use of systematic and repeatable techniques that ensure the completeness, consistency, and relevance of the system requirements
	Systems engineering	The objective of the article is to focus on means in order to enable realization of successful systems. Systems engineering integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation
Research methodology adopted	Qualitative	The concerned articles investigate issues concerning requirement prioritization from different perspectives to gain an understanding of the problem under investigation
	Quantitative	The concerned articles emphasize objective measurements and the statistical, mathematical, or numerical analysis of data collected through polls, questionnaires, and surveys, or by manipulating preexisting statistical data using computational techniques to address the problem under investigation
	Mixed	The concerned articles combine both qualitative and quantitative methodologies in their investigation of the requirement prioritization issue under consideration
Project methodology	General	The article does not mention of a specific methodology that characterizes the project of which the requirements under consideration are part of
	Agile methodology	The requirements considered for prioritization are specifically part of projects that implements the agile methodology. Agile methodology includes methods based on iterative and incremental development, where requirements and solutions evolve through collaboration between self-organizing, cross-functional teams
Type of requirements prioritized	General	The article does not explicitly mention the type of requirements considered for prioritization in the artifact under description
	Functional requirements	Functional requirements relate to the statement of services the system is expected to provide, how the system is expected to react to particular inputs, and how the system is expected to behave in particular situations
	NFRs	Software system attributes also known as quality requirements or non-functional requirements (NFRs) represent the services or functions offered by the system. These apply to the system as a whole and describe the capabilities of the system. NFRs include timing constraints, and constraints on the development process and standards
Prioritization objective (concerns)	Demonstrating requirement priority	The objective of the prioritization process embedded in the artifact is to arrive at an indicator of priority of the candidate requirements in appropriate format. The priority measure is an indicator of the ranking of the requirements on chosen attributes, for example performance-based ranking of requirements
	Requirement selection	The objective of the prioritization process embedded in the artifact is to specify selection of requirements for allocation to activities or entities. The articles that are classified under this dimension discuss the requirement prioritization process so as to facilitate selection or allocation of the requirements as per the research objectives
	Categorizing requirements	The objective of the prioritization process embedded in the artifact is to merge and group the candidate requirements in pre-defined categories. The prioritization process discussed in the articles classified under this dimension categorizes requirements as per the research objectives, for example grouping requirements based on functionalities and classifying requirements on a matrix
	Requirement value assessment	The objective of the prioritization process embedded in the artifact is to arrive at a suitable representation of requirement value. Requirement value may represent the extent of satisfaction that is achievable from implementation of a set of requirements, a measure of some completeness value of requirements based on certain attributes, a measure of the importance of the requirement based on certain criteria, etc

Table 14 continued

Category	Subcategory	Description
	Others	The objective of the prioritization process embedded in the artifact is concerns other than those stated above, like for example, requirement engineering concerns (e.g., requirement specification issues), stakeholder concerns (e.g., stakeholder choices and preferences), etc
Number of prioritization objectives	Single	The prioritization process embedded in the artifact is guided by a single objective
	Multiple	The prioritization process embedded in the artifact is guided by more than one objective
Prioritization artifact characteristic	Fixed importance	Here each requirement's priority value is assumed to be fixed
	Requirement grouping	Requirements are grouped into bundles with due consideration of the important dependencies such that each group is relatively independent of the others
	Relative versus absolute	Absolute value implies that actual values of parameters characterizing the requirements are used in the prioritization process. Relative value implies assignment of values relative to a scale specifically chosen for the purpose
	Pairwise comparison	This refers to mutual (i.e., one-to-one) comparison of the relevant attributes of the entities (i.e., requirements) under consideration
	Discrete versus continuous scale	Discrete scale implies that the parameters characterizing the requirements are measured using a set of categories, e.g., an ordinal scale. Continuous scale implies that the parameters characterizing the requirements are measured using an interval or a ratio scale
	Iterative versus non-iterative	Iterative implies that the prioritization process specifies repetition over several steps specified in the computation procedure. Non-iterative is a sequential computation procedure without any repetition of the steps
Theoretical basis of prioritization	Theory driven	The prioritization process embedded in the requirement prioritization artifact is supported by a clearly delineable theoretical core. The design of the prioritization process in this case is based on existing social, natural, or design science theories
	Argumentative	There is no particular theoretical underpinning of the prioritization process. The design of the prioritization process is motivated by argumentation and references to literature. There is no explicit specification of theories underlying the design
Contribution	Construct	The contribution of the article is a set of vocabulary or symbols that informs the design of a requirement prioritization artifact
	Model	The contribution of the article is a specification (abstraction/representation) of some sort of relationships among established or identified constructs that informs the design of a requirement prioritization artifact
	Method	The contribution of the article is a specification of algorithms, practices, approaches, or techniques toward designing the requirement prioritization artifact
	Methodology	The contribution of the article is a specification of a set of methods, principles and rules toward designing the requirement prioritization artifact
	Framework	The contribution of the article is a specification of the structure (real or conceptual) toward supporting or guiding the design of the requirement prioritization artifact
	System[a]	The contribution of the article is a specification of an integrated set of components that describes a requirement prioritization artifact
	Tool	The contribution of the article is a suitable representation of the physical instant of a requirement prioritization artifact
Underlying artifact on which the contribution is based	Yes/no	Yes: If the requirement prioritization artifact under discussion extends or uses a prior established artifact
		No: if otherwise
Type of instantiation	Conceptual instantiation	The representation of the requirement prioritization artifact is abstract
	Physical instantiation	The representation of the requirement prioritization artifact is concrete, i.e., the concerned artifact has been implemented in a tool for demonstration or usage purpose

[a] The difference between a system and a framework is that a system is a depiction of an integrated set of components, whereas a framework may serve as a guide toward realization of such a system

Table 15 Studies included in the review

ID	Article details
[S1]	Aasem, M., Ramzan, M., and Jaffar, A. "Analysis and optimization of software requirements prioritization techniques," International Conference on Information and Emerging Technologies (ICIET), IEEE, 2010, pp. 1–6
[S2]	Ahl, V. "An experimental comparison of five prioritization methods," Master's Thesis, School of Engineering, Blekinge Institute of Technology, Ronneby, Sweden) 2005
[S3]	Ahmad, A., Shahzad, A., Padmanabhuni, K., Mansoor, A., Joseph, S., and Arshad, Z. "Requirements prioritization with respect to Geographically Distributed Stakeholders," IEEE International Conference on. Computer Science and Automation Engineering (CSAE), 2011, pp. 290–294
[S4]	Aho, A.V., Hopcroft, J.E., and Ullman, J.D. "Data structures and algorithms. 1983," Addison-Wesley
[S5]	Asghar, M.W., Marchetto, A., Susi, A., and Scanniello, G. "Maintainability-based requirements prioritization by using artifacts traceability and code metrics," 17th European Conference on Software Maintenance and Reengineering (CSMR) IEEE, 2013, pp. 417–420
[S6]	Avesani, P., Bazzanella, C., Perini, A., and Susi, A. "Facing scalability issues in requirements prioritization with machine learning techniques," 13th IEEE International Conference on Requirements Engineering, 2005., IEEE, 2005, pp. 297–305
[S7]	Azar, J., Smith, R.K., and Cordes, D. "Value-oriented requirements prioritization in a small development organization," Software, IEEE (24:1) 2007, pp 32–37
[S8]	Babar, M.I., Ramzan, M., and Ghayyur, S.A.K. "Challenges and future trends in software requirements prioritization," International Conference on Computer Networks and Information Technology (ICCNIT), IEEE, 2011, pp. 319–324
[S9]	Bagheri, E., Asadi, M., Gasevic, D., and Soltani, S. "Stratified analytic hierarchy process: Prioritization and selection of software features," in: Software Product Lines: Going Beyond, Springer, 2010, pp. 300–315
[S10]	Bakalova, Z., Daneva, M., Herrmann, A., and Wieringa, R. "Agile requirements prioritization: What happens in practice and what is described in literature," in: Requirements engineering: Foundation for software quality, Springer, 2011, pp. 181–195
[S11]	Barney, S., Aurum, A., and Wohlin, C. "A product management challenge: Creating software product value through requirements selection," Journal of Systems Architecture (54:6) 2008, pp 576–593
[S12]	Bebensee, T., van de Weerd, I., and Brinkkemper, S. "Binary priority list for prioritizing software requirements," in: Requirements Engineering: Foundation for Software Quality, Springer, 2010, pp. 67–78
[S13]	Beck, K., and Andres, C. Extreme programming explained: embrace change Addison-Wesley Professional, 2004
[S14]	Babar, M.I., Ghazali, M., Jawawi, D.N., Shamsuddin, S.M., and Ibrahim, N. "PHandler: An expert system for a scalable software requirements prioritization process," Knowledge-Based Systems (84) 2015, pp 179–202
[S15]	Beg, M.R., Verma, R.P., and Joshi, A. "Reduction in number of comparisons for requirement prioritization using B-Tree," IEEE International Advance Computing Conference, IACC 2009., IEEE, 2009, pp. 340–344
[S16]	Benestad, H.C., and Hannay, J.E. "Does the prioritization technique affect stakeholders' selection of essential software product features?," Proceedings of the ACM-IEEE international symposium on Empirical software engineering and measurement, ACM, 2012, pp. 261–270
[S17]	Berander, P., and Johansson, P. "Hierarchical cumulative voting (hcv) prioritization of requirements in hierarchies," International Journal of Software Engineering and Knowledge Engineering (16:06) 2006, pp 819–849
[S18]	Berander, P. "Using students as subjects in requirements prioritization," International Symposium on Empirical Software Engineering, IEEE, 2004, pp. 167–176
[S19]	Berander, P., and Andrews, A. "Requirements prioritization," in: Engineering and managing software requirements, Springer, 2005, pp. 69–94
[S20]	Berander, P., and Svahnberg, M. "Evaluating two ways of calculating priorities in requirements hierarchies—An experiment on hierarchical cumulative voting," Journal of Systems and Software (82:5) 2009, pp 836–850
[S21]	Berteig, M. "Methods of Prioritization, March 20, 2006 in Agile Advice online practitioners forum," 2006
[S22]	Brodie, L., and Woodman, M. "Prioritization of Stakeholder Value Using Metrics," in: Evaluation of Novel Approaches to Software Engineering, Springer, 2011, pp. 74–88
[S23]	Büyüközkan, G. "Determining the mobile commerce user requirements using an analytic approach," Computer Standards & Interfaces (31:1) 2009, pp 144–152
[S24]	Carlshamre, P. "Release planning in market-driven software product development: Provoking an understanding," Requirements Engineering (7:3) 2002, pp 139–151
[S25]	Chatzipetrou, P., Angelis, L., Rovegard, P., and Wohlin, C. "Prioritization of issues and requirements by cumulative voting: a compositional data analysis framework," 6th EUROMICRO Conference on Software Engineering and Advanced Applications (SEAA), IEEE, 2010, pp. 361–370
[S26]	Cleland-Huang, J., and Denne, M. "Financially informed requirements prioritization," 27th International Conference on ICSE 2005., IEEE, 2005, pp. 710–711

Table 15 continued

ID	Article details
[S27]	Crow, K. "Customer-focused development with QFD," Annual Quality Congress Proceedings, American Society for Quality Control, 1994, pp. 839–839
[S28]	Danesh, A.S., Mortazavi, S.M., and Danesh, S.Y.S. "Requirements prioritization in on-line banking systems: using value-oriented framework," International Conference on Computer Technology and Development, IEEE, 2009, pp. 158–161
[S29]	Danesh, A.S., and Ahmad, R. "Study of prioritization techniques using students as subjects," International Conference on Information Management and Engineering, IEEE, 2009, pp. 390–394
[S30]	Daneva, M., Van Der Veen, E., Amrit, C., Ghaisas, S., Sikkel, K., Kumar, R., Ajmeri, N., Ramteerthkar, U., and Wieringa, R. "Agile requirements prioritization in large-scale outsourced system projects: An empirical study," Journal of Systems and Software (86:5) 2013, pp 1333–1353
[S31]	Daneva, M., and Herrmann, A. "Requirements prioritization based on benefit and cost prediction: A method classification framework," Euromicro Conference Software Engineering and Advanced Applications, IEEE, 2008, pp. 240–247
[S32]	Davis, A.M. "The art of requirements triage," Computer (36:3) 2003, pp 42–49
[S33]	Doerr, J., Hartkopf, S., Kerkow, D., Landmann, D., and Amthor, P. "Built-in user satisfaction-Feature appraisal and prioritization with AMUSE," 15th IEEE International Requirements Engineering Conference, RE'07., IEEE, 2007, pp. 101–110
[S34]	Duan, C., Laurent, P., Cleland-Huang, J., and Kwiatkowski, C. "Towards automated requirements prioritization and triage," Requirements engineering (14:2) 2009, pp 73–89
[S35]	Elahi, G., and Yu, E. "Comparing alternatives for analyzing requirements trade-offs-In the absence of numerical data," Information and Software Technology (54:6) 2012, pp 517–530
[S36]	Feather, M.S., and Menzies, T. "Converging on the optimal attainment of requirements," IEEE Joint International Conference on Requirements Engineering, IEEE, 2002, pp. 263–270
[S37]	Firesmith, D. "Prioritizing Requirements," Journal of Object Technology (3:8) 2004, pp 35–48
[S38]	Fraser, J. (2002) Setting Priorities [online]. http://www.adaptivepath.net/ideas/e000018/
[S39]	Galster, M., and Eberlein, A. "Facilitating software architecting by ranking requirements based on their impact on the architecture process," IEEE International Conference and Workshops on Engineering of Computer Based Systems (ECBS), IEEE, 2011, pp. 232–240
[S40]	Gaur, V., Soni, A., and Bedi, P. "An application of multi-person decision-making model for negotiating and prioritizing requirements in agent-oriented paradigm," International Conference on Data Storage and Data Engineering (DSDE), IEEE, 2010, pp. 164–168
[S41]	Gottesdiener, E. "At a Glance: Other Prioritization Methods," EBG Consulting, 2007. http://www.ebgconsulting.com/Pubs/Articles/At%20a%20Glance-Other%20Prioritization%20Methods-supplement-EBG%20Consulting.pdf
[S42]	Greer, D., and Ruhe, G. "Software release planning: an evolutionary and iterative approach," Information and Software Technology (46:4) 2004, pp 243–253
[S43]	Gupta, V., Chauhan, D.S., and Dutta, K. "Incremental development & revolutions of E-learning software systems in education sector: a case study approach," Human-centric Computing and Information Sciences (3:1) 2013, pp 1–14
[S44]	Hatton, S. "Choosing the right prioritisation method," 19th Australian Conference on Software Engineering, ASWEC 2008., IEEE, 2008, pp. 517–526
[S45]	Herrmann, A., and Daneva, M. "Requirements prioritization based on benefit and cost prediction: An agenda for future research," 16th IEEE International Requirements Engineering, RE'08., IEEE, 2008, pp. 125–134
[S46]	Herrmann, A., and Paech, B. "Practical challenges of requirements prioritization based on risk estimation," Empirical Software Engineering (14:6), 2009, pp. 644–684
[S47]	Iqbal, M.A., Zaidi, A.M., and Murtaza, S. "A New Requirement Prioritization Model for Market Driven Products Using Analytical Hierarchical Process," International Conference on Data Storage and Data Engineering (DSDE), IEEE, 2010, pp. 142–149
[S48]	Karlsson, J. "Towards a Strategy for Software Requirements Selection. Licentiate," in: Department of Computer and Information Science, Linköping University, 1995
[S49]	Karlsson, J. "Software requirements prioritizing," Proceedings of the Second International Conference on Requirements Engineering, IEEE, 1996, pp. 110–116
[S50]	Karlsson, J., Olsson, S., and Ryan, K. "Improved practical support for large-scale requirements prioritising," Requirements Engineering (2:1) 1997, pp 51–60
[S51]	Karlsson, J., Wohlin, C., and Regnell, B. "An evaluation of methods for prioritizing software requirements," Information and Software Technology (39:14) 1998, pp 939–947
[S52]	Karlsson, L., Berander, P., Regnell, B., and Wohlin, C. "Requirements prioritisation: an experiment on exhaustive pair-wise comparisons versus planning game partitioning," Empirical Assessment in Software Engineering (EASE 2004)) 2004, pp 145–154
[S53]	Karlsson, L., Höst, M., and Regnell, B. "Evaluating the practical use of different measurement scales in requirements prioritisation," ACM, 2006, pp. 326–335

Table 15 continued

ID	Article details
[S54]	Karlsson, L., Thelin, T., Regnell, B., Berander, P., and Wohlin, C. "Pair-wise comparisons versus planning game partitioning—experiments on requirements prioritisation techniques," Empirical Software Engineering (12:1) 2007, pp 3–33
[S55]	Karlsson, J., and Ryan, K. "A cost-value approach for prioritizing requirements," Software, IEEE (14:5) 1997, pp 67–74
[S56]	Keeney, R.L. "Foundations for making smart decisions," IIE solutions (31:5) 1999, pp 24–30
[S57]	Keeney, R.L. Decisions with multiple objectives: preferences and value trade-offs Cambridge University Press, 1993
[S58]	Kerkow, D., Doerr, J., Paech, B., Olsson, T., and Koenig, T. "Elicitation and documentation of non-functional requirements for sociotechnical systems," Requirements Engineering for Sociotechnical Systems, Idea Group, Inc) 2004
[S59]	Koziolek, A. "Architecture-driven quality requirements prioritization," Requirements Engineering: Foundation for Software Quality, IEEE, 2012, pp. 15–19
[S60]	Kruskal, J.B. "On the shortest spanning subtree of a graph and the traveling salesman problem," Proceedings of the American Mathematical society (7:1) 1956, pp 48–50
[S61]	Kukreja, N. "Decision theoretic requirements prioritization: a two-step approach for sliding towards value realization," 35th International Conference on Software Engineering (ICSE), IEEE Press, 2013, pp. 1465–1467
[S62]	Kukreja, N., Payyavula, S.S., Boehm, B., and Padmanabhuni, S. "Selecting an Appropriate Framework for Value-Based Requirements Prioritization: A Case Study," 20th IEEE International Requirements Engineering Conference, Illinois, Chicago, USA, 2012
[S63]	Laurent, P., Cleland-Huang, J., and Duan, C. "Towards automated requirements triage," 15th IEEE International Requirements Engineering Conference, RE'07., IEEE, 2007, pp. 131–140
[S64]	Laurent, P., and Cleland-Huang, J. "Lessons learned from open source projects for facilitating online requirements processes," in: Requirements Engineering: Foundation for Software Quality, Springer, 2009, pp. 240–255
[S65]	Lauesen, S. Software requirements: styles and techniques Pearson Education, Upper Saddle River, NJ, 2002
[S66]	Lehtola, L. "Providing value by prioritizing requirements throughout product development: State of practice and suitability of prioritization methods," Licentiate Thesis, HUT/Department of Computer Science) 2006
[S67]	Lehtola, L., Kauppinen, M., and Kujala, S. "Requirements prioritization challenges in practice," in: Product focused software process improvement, Springer, 2004, pp. 497–508
[S68]	Lehtola, L., and Kauppinen, M. "Empirical evaluation of two requirements prioritization methods in product development projects," Software Process Improvement, Springer, 2004, pp. 161–170
[S69]	Lehtola, L., and Kauppinen, M. "Suitability of requirements prioritization methods for market-driven software product development," Software Process: Improvement and Practice (11:1) 2006, pp 7–19
[S70]	Kakar, A.K. "Investigating the penalty reward calculus of software users and its impact on requirements prioritization," Information and Software Technology (65) 2015, pp 56–68
[S71]	Li, Y.-L., Chin, K.-S., and Luo, X.-G. "Determining the final priority ratings of customer requirements in product planning by MDBM and BSC," Expert systems with Applications (39:1) 2012, pp 1243–1255
[S72]	Li, C., Van Den Akker, J.M., Brinkkemper, S., and Diepen, G. "Integrated requirement selection and scheduling for the release planning of a software product," in: Requirements Engineering: Foundation for Software Quality, Springer, 2007, pp. 93–108
[S73]	Maiti, R.R., and Mitropoulos, F.J. "Capturing, eliciting, predicting and prioritizing (CEPP) non-functional requirements metadata during the early stages of agile software development," in SoutheastCon 2015, pp. 1–8
[S74]	Liaskos, S., McIlraith, S.A., Sohrabi, S., and Mylopoulos, J. "Representing and reasoning about preferences in requirements engineering," Requirements Engineering (16:3) 2011, pp 227–249
[S75]	Lim, S.L., and Finkelstein, A. "StakeRare: using social networks and collaborative filtering for large-scale requirements elicitation," IEEE Transactions on Software Engineering (38:3) 2012, pp 707–735
[S76]	Lima, D.C., Freitas, F., Campos, G., and Souza, J. "A fuzzy approach to requirements prioritization," in: Search Based Software Engineering, Springer, 2011, pp. 64–69
[S77]	Liu, X., Veera, C.S., Sun, Y., Noguchi, K., and Kyoya, Y. "Priority assessment of software requirements from multiple perspectives," IEEE, 2004, pp. 410–415
[S78]	Logue, K., and McDaid, K. "Handling uncertainty in agile requirement prioritization and scheduling using statistical simulation," AGILE'08. Conference, IEEE, 2008, pp. 73–82
[S79]	Marjaie, S.A., and Kulkarni, V. "Recognition of hidden factors in requirements prioritization using factor analysis," IEEE, 2010, pp. 1–5
[S80]	Martinez, A.B.B., Arias, J.J.P., Vilas, A.F., Duque, J.G., Nores, M.L., Redondo, R.P.D., and Fernández, Y.B. "Composing requirements specifications from multiple prioritized sources," Requirements Engineering (13:3) 2008, pp 187–206
[S81]	Mendizabal, O.M., Spier, M., and Saad, R. "Log-based approach for performance requirements elicitation and prioritization," 20th IEEE International Requirements Engineering Conference (RE), IEEE, 2012, pp. 297–302
[S82]	Moisiadis, F. "The fundamentals of prioritising requirements," Proceedings of the systems engineering, test and evaluation conference (SETE'2002), Citeseer, 2002, pp. 108–119

Table 15 continued

ID	Article details
[S83]	Mu, K., Liu, W., Jin, Z., Lu, R., Yue, A., and Bell, D. "Handling inconsistency in distributed software requirements specifications based on prioritized merging," Fundamenta Informaticae (91:3) 2009, pp 631–670
[S84]	Nas, T.F. Cost-benefit analysis: Theory and application Sage, 1996
[S85]	Negräo, E.C., and Guerra, E.M. "A case study for prioritizing features in environments with multiple stakeholders," Proceedings of the ACM international conference companion on Object oriented programming systems languages and applications companion, ACM, 2011, pp. 131–140
[S86]	Otero, C.E., Dell, E., Qureshi, A., and Otero, L.D. "A quality-based requirement prioritization framework using binary inputs," Fourth Asia International Conference on Mathematical/Analytical Modelling and Computer Simulation (AMS), IEEE, 2010, pp. 187–192
[S87]	Palma, F., Susi, A., and Tonella, P. "Using an SMT solver for interactive requirements prioritization," Proceedings of the 19th ACM SIGSOFT symposium and the 13th European conference on Foundations of software engineering, ACM, 2011, pp. 48–58
[S88]	Park, J.-W., Port, D., and Boehm, B. "Supporting distributed collaborative prioritization," Sixth Asia Pacific Software Engineering Conference, (APSEC'99)., IEEE, 1999, pp. 560–563
[S89]	Patton, J. "Finding the forest in the trees," Companion to the 20th annual ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications, ACM, 2005, pp. 266–274
[S90]	Perini, A., Ricca, F., and Susi, A. "Tool-supported requirements prioritization: Comparing the AHP and CBRank methods," Information and Software Technology (51:6) 2009, pp 1021–1032
[S91]	Misra, J., Sengupta, S., and Das, S. "Latent semantic centrality based automated requirements prioritization," Proceedings of the 7th India Software Engineering Conference (ISEC), ACM, 2014
[S92]	Port, D., Olkov, A., and Menzies, T. "Using simulation to investigate requirements prioritization strategies," Proceedings of the 2008 23rd IEEE/ACM International Conference on Automated Software Engineering, IEEE Computer Society, 2008, pp. 268–277
[S93]	Port, D., Boehm, B., and Klappholz, D. "Nancy R. Mead: Making Requirements Prioritization a Priority," 21st IEEE Conference on Software Engineering Education and Training, CSEET'08., IEEE, 2008, pp. 250–261
[S94]	Racheva, Z., Daneva, M., Sikkel, K., Herrmann, A., and Wieringa, R. "Do we know enough about requirements prioritization in agile projects: insights from a case study," 18th IEEE International Requirements Engineering Conference (RE), IEEE, 2010, pp. 147–156
[S95]	Racheva, Z., Daneva, M., Herrmann, A., and Wieringa, R.J. "A conceptual model and process for client-driven agile requirements prioritization," Fourth International Conference on Research Challenges in Information Science (RCIS), IEEE, 2010, pp. 287–298
[S96]	Ramzan, M., Jaffar, M.A., Iqbal, M.A., Anwar, S., and Shahid, A.A. "Value based fuzzy requirement prioritization and its evaluation framework," Fourth International Conference on Innovative Computing, Information and Control (ICICIC), IEEE, 2009, pp. 1464–1468
[S97]	Regnell, B., Höst, M., och Dag, J.N., Beremark, P., and Hjelm, T. "An industrial case study on distributed prioritisation in market-driven requirements engineering for packaged software," Requirements Engineering (6:1) 2001, pp 51–62
[S98]	Regnell, B., Höst, M., and Svensson, R.B. "A quality performance model for cost-benefit analysis of non-functional requirements applied to the mobile handset domain," in: Requirements Engineering: Foundation for Software Quality, Springer, 2007, pp. 277–291
[S99]	Ribeiro, R.A., Moreira, A.M., Van den Broek, P., and Pimentel, A. "Hybrid assessment method for software engineering decisions," Decision Support Systems (51:1) 2011, pp 208–219
[S100]	Riegel, N. "Model-based prioritization in business-process-driven software development," 20th IEEE International Requirements Engineering Conference (RE), IEEE, 2012, pp. 349–352
[S101]	Roy, B. Multicriteria methodology for decision aiding Springer, 1996
[S102]	Ruhe, G., Eberlein, A., and Pfahl, D. "Trade-off analysis for requirements selection," International Journal of Software Engineering and Knowledge Engineering (13:4) 2003, pp 345–366
[S103]	Riegel, N., and Doerr, J. "An analysis of priority-based decision heuristics for optimizing elicitation efficiency," in: Requirements Engineering: Foundation for Software Quality, Springer, 2014, pp. 268–284
[S104]	Tl, S. "The analytic hierarchy process," McGraw-Hill, New York, 1980
[S105]	Schwaber, K.X. Agile project management with Scrum O'Reilly Media, Inc., 2004
[S106]	Sen, C.G., and Baraçlı, H. "Fuzzy quality function deployment based methodology for acquiring enterprise software selection requirements," Expert Systems with Applications (37:4) 2010, pp 3415–3426
[S107]	Shao, P. "Sample selection: an algorithm for requirements prioritization," 46th Annual Southeast Regional Conference on Software Engineering, ACM, 2008, pp. 525–526
[S108]	Smiley, K., He, Q., Kielczewski, E., and Dagnino, A. "Architectural requirements prioritization and analysis applied to software technology evaluation," 2009 ACM symposium on Applied Computing, ACM, 2009, pp. 397–398
[S109]	Sobczak, A., and Berry, D.M. "Distributed priority ranking of strategic preliminary requirements for management information systems in economic organizations," Information and Software technology (49:9) 2007, pp 960–984
[S110]	Sun, Y., and Liu, X.F. "Business-oriented software process improvement based on CMMI using QFD," Information and software technology (52:1) 2010, pp 79–91

Table 15 continued

ID	Article details
[S111]	Svensson, R.B., Gorschek, T., Regnell, B., Torkar, R., Shahrokni, A., Feldt, R., and Aurum, A. "Prioritization of quality requirements: State of practice in eleven companies," 19th IEEE International Requirements Engineering Conference (RE), IEEE, 2011, pp. 69–78
[S112]	Tabaka, J.X. Collaboration explained: facilitation skills for software project leaders Pearson Education, 2006
[S113]	Thakurta, R. "A framework for prioritization of quality requirements for inclusion in a software project," Software Quality Journal) 2013, pp 573–597
[S114]	Tonella, P., Susi, A., and Palma, F. "Using interactive GA for requirements prioritization," Second International Symposium on Search Based Software Engineering (SSBSE), IEEE, 2010, pp. 57–66
[S115]	Tonella, P., Susi, A., and Palma, F. "Interactive requirements prioritization using a genetic algorithm," Information and Software Technology (55:1) 2013, pp 173–187
[S116]	Tourwé, T., Codenie, W., Boucart, N., and Blagojević, V. "Demystifying release definition: from requirements prioritization to collaborative value quantification," in: Requirements Engineering: Foundation for Software Quality, Springer, 2009, pp. 37–44
[S117]	Trienekens, J.J.M., Kusters, R.J., and Brussel, D.C. "Quality specification and metrication, results from a case-study in a mission-critical software domain," Software Quality Journal (18:4) 2010, pp 469–490
[S118]	Voola, P., and Babu, A.V. "Interval evidential reasoning algorithm for requirements prioritization," International Conference on Information Systems Design and Intelligent Applications, Springer, Visakhapatnam, India, 2012, pp. 915–922
[S119]	Waters, K. "Prioritization using MoSCoW," in: Agile Planning, 2009
[S120]	Wiegers, K. "First things first: prioritizing requirements," Software Development (7:9) 1999, pp 24–30
[S121]	Yeh, A.C. "Requirements engineering support technique (REQUEST): a market driven requirements management process," Proceedings of the Second Symposium on Assessment of Quality Software Development Tools, IEEE, 1992, pp. 211–223
[S122]	Yu, D., He, K., Wang, J., and Mao, H. "An Algorithm for Priority Ranking of Individualized Functional Requirements in Networked Software," Seventh International Conference on Grid and Cooperative Computing, GCC'08., IEEE, 2008, pp. 701–706
[S123]	Yi-zeng, C., and Qi, Y. "A fuzzy game approach to prioritize customer requirements in Quality Function Deployment," International Conference on Management Science & Engineering (ICMSE), 2014, pp. 207–212, IEEE
[S124]	Zhang, H., Harman, M., Finkelstein, A., and Mansouri, S.A. "Comparing the performance of metaheuristics for the analysis of multi-stakeholder tradeoffs in requirements optimisation," Information and Software Technology (53:7) 2011, pp 761–773
[S125]	Azmeh, Z., Mirbel, I., and El Jiani, L. "A tool to improve requirements review in collaborative software development platforms," Eighth International Conference on Research Challenges in Information Science (RCIS), 2014, pp. 1–6, IEEE
[S126]	Cleland-Huang, J. "Injecting Value-Thinking into Prioritization Decisions," IEEE Software (2) 2015, pp 14–18
[S127]	Dabbagh, M., Sai P. L., and Reza M. P. "Application of hybrid assessment method for priority assessment of functional and non-functional requirements," International Conference on Information Science and Applications (ICISA), 2014, IEEE
[S128]	Elsood, M., Hefny, H., and Nasr, E.S. "A goal-based technique for requirements prioritization," 9th International Conference on Informatics and Systems (INFOS), 2014, pp. SW-18, IEEE
[S129]	Fernandes, J.M., Prozil Rodrigues, S., and Costa, L.A. "Comparing AHP and ELECTRE I for prioritizing software requirements," 16th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD), 2015, pp. 1–8, IEEE
[S130]	Franceschini, F., Maisano, D., and Mastrogiacomo, L. "Customer requirement prioritization on QFD: a new proposal based on the generalized Yager's algorithm," Research in Engineering Design 26(2), 2015, pp. 171–187
[S131]	Modern Analyst LLC. "Requirements Prioritization," http://www.requirements.com/Glossary/RequirementsPrioritization/tabid/121/Default.aspx, 2014
[S132]	Mustafa, B., and Zainuddin, A. "An experimental design to compare software requirements prioritization techniques," International Conference on Computational Science and Technology (ICCST), 2014, pp. 1–5, IEEE
[S133]	Saxena, S.K., and Chakraborty, R. "Decisively: Application of Quantitative Analysis and Decision Science in Agile Requirements Engineering," 22nd International Requirements Engineering Conference (RE), 2014, pp. 323–324, IEEE
[S134]	Sher, F., Jawawi, D.N., Mohamad, R., and Babar, M.I. "Multi-aspects based requirements prioritization technique for value-based software developments," International Conference on Emerging Technologies (ICET), 2014, pp. 1–6, IEEE
[S135]	Sureka, A. "Requirements Prioritization and Next-Release Problem under Non-Additive Value Conditions," 23rd Australian Software Engineering Conference (ASWEC), 2014, pp. 120–123, IEEE

Table 16 Article focus areas

Article ID	Article focus
Requirements engineering	
S11	The article describes the requirement release planning processes used in the software industry to create software product value. Here requirement prioritization has been discussed as a part of the case studies in the analysis of the requirement release planning processes in three companies
S24	The article discusses the implementation and evaluation of a support tool for software requirement release planning. The support tool implements a prioritization algorithm for requirement selection
S42	The article presents an approach which offers decision support for software release planning in incremental software development. The approach includes iterative computation of requirement priorities in each increments based on a genetic algorithm
S46	The article explores practical challenges and needs of risk estimations during requirements engineering. As a part of the experimentations, the authors identified factors, which influence the quality of requirement prioritization, and examined a risk-based prioritization method
S64	The article explores and evaluates the requirement engineering processes adopted by vendor-based open-source software projects. The effectiveness of various requirement prioritization practices was evaluated toward making suitable recommendations
S72	The article examines the process of requirement selection and scheduling during requirement release planning. The authors propose two integer linear programming models and compare them with traditional prioritization models in order to make observations on the requirement selection and scheduling effectiveness
S78	The article focuses on managing uncertainty in the planning of what functionality to include in upcoming releases in agile projects. The article proposes a statistical methodology toward agile requirements prioritization that accounts for uncertainties within the business value of each story, the story size and the resources available
S83	The article addresses handling inconsistencies during requirement specification. A prioritized-based merging framework is proposed in an attempt to address the issue
S89	The article discusses requirement identification issues and challenges in agile software development. In the description of a release strategy, a prioritization method has been illustrated
S103	The article presents a set of priority-based decision heuristics in order to support requirements engineers in deciding which requirements should be elaborated next at a certain point during elicitation of requirements
S106	The article presents an approach for identifying the non-functional requirements that are important to a company's software selection decision based on and integrated with its functional requirements. The identification of the non-functional requirements is based on prioritizing these requirements using a fuzzy quality function deployment-based methodology so as to arrive at an importance degree of these requirements
S117	The article presents an approach for dealing with software quality in the process of mission-critical systems engineering. A case study is described which uses requirement prioritization while addressing specification, prioritization, and metrication of software product quality
S121	The article discusses the requirements engineering support technique (REQUEST) which introduces technical and managerial discipline into the requirements engineering process. The REQUEST technique uses prioritization in one of its stages as it transforms customer requirements in an understandable and measurable format
S124	The article analyzes internal tensioning among multi-stakeholder in requirements analysis. This introduces two multi-objective requirement prioritization methods for selecting stakeholders' competing requirements in the presence of increasing budgetary pressure
S125	The article discusses an approach of improving the prioritization process by automatically grouping similar requirements in collaborative software development. The authors present a tool that uses prioritization to break large set of requirements into meaningful categories
S133	The article presents an agile requirement engineering (RE) tool, Decisively, which brings a new perspective to automation in the RE process through application of Quantitative Analysis and Decision Science (QUADS) to address requirement discovery, analysis, estimation, and prioritization. The prioritization is achieved using the *analytic hierarchy process (AHP)* technique
Systems engineering	
S56	The article discusses how to develop a qualify foundation for engineering decisions. A prioritization approach is introduced for organizing objectives as a part of decision making
S99	The article assesses the suitability of multi-criteria decision making (MCDM) methods to support software engineers' decisions. The analysis uses a requirement prioritization method to arrive at a priority measure of a set of requirements characterizing a software architectural alternative
S108	The article focuses on evaluation of software technology alternatives that have important architectural impact. The research applied the Attribute Hierarchy-based Evaluation of Architectural Designs (AHEAD) method which involves prioritization of architectural requirements
S110	The article presents a software process improvement (SPI) framework in order to ensure highest satisfaction level of process requirements that is achievable. The framework integrates and prioritizes requirements from multiple perspectives so as to prioritize SPI actions based on process requirements

Table 17 Requirement prioritization objectives

ID	Brief objective of the article
Demonstrating requirement priority	
[S1]	The article describes a framework to perform the requirement prioritization process by combining existing techniques and approaches
[S5]	The article describes a tool that prioritizes (change) requirements by using artifacts traceability information, to locate the requirements implementation, and a set of code-based metrics, to measure several properties (e.g., coupling, size, and scattering) of the requirements implementation
[S6]	The article describes a framework called case-based framework for requirements prioritization which exploits machine learning techniques to overcome the scalability problem with growing number of requirements
[S7]	The article describes the value-oriented prioritization (VOP) framework for clarifying and quantifying requirement prioritization issues for selection
[S9]	The article presents a novel method, the Stratified Analytic Hierarchy process, which first helps to rank and select the most relevant high-level business objectives for the target stakeholders (e.g., security over implementation costs) and then helps to rank and select the most relevant features from the feature model to be used as the starting point in the staged configuration process
[S12]	The article introduces a binary search-based technique for prioritizing requirements and assesses its prioritization process quality by comparing it to another prioritization technique
[S14]	The article proposes an expert system, called the Priority Handler (PHandler), for prioritizing requirements which is based on the value-based intelligent requirement prioritization technique, neural network, and the analytical hierarchical process (AHP)
[S17]	The article proposes a new technique for prioritizing hierarchically structured requirements by combining strengths and weaknesses of two existing approaches, viz. AHP and Cumulative Voting (CV)
[S19]	The article presents an overview of techniques for prioritization of requirements for software products and illustrates how to combine different techniques based on overall consideration
[S20]	The article presents an approach for prioritizing requirements residing on several hierarchical levels and at different abstraction levels
[S21]	The article discusses several methods of collaboratively prioritizing work item lists of project's implementing agile methodologies
[S23]	The article proposes an analytic framework to provide practitioners an effective and efficient model for prioritizing m-commerce requirements
[S28]	The article presents a study designed for prioritizing of online banking system requirements through using value-oriented framework based on business value
[S29]	The article compares two different prioritization methods, viz. AHP and numerical assignment using students as subjects
[S34]	The article proposes a method of requirement prioritization which automates a significant part of the prioritization process by utilizing data mining and machine learning techniques
[S39]	The article presents a method for ranking requirements with a focus on the impact of requirements on the architecture process.
[S41]	The article discusses some of the practices for prioritizing product requirements
[S43]	The article proposes a new requirement prioritization method that selects those requirements for implementation that are essentially required by stakeholders and has a lower regression count associated with them, thereby reducing regression testing effort
[S46]	The article explores practical challenges and needs of risk estimations in general and of the MOQARE (misuse-oriented quality requirements engineering) method to support intuitive and systematic identification of quality requirements
[S47]	The article presents a new model for requirement prioritization using AHP (analytical hierarchical process) for market-driven software development
[S49]	The article presents a case study at Ericsson Radio Systems AB to introduce two techniques for software requirements prioritizing as a means for determining the importance of candidate requirements
[S50]	The article improves an existing cost-value approach of prioritizing requirements by reducing number of comparisons and managing requirement interdependencies
[S51]	The article describes and evaluates six different methods for prioritizing software requirements
[S59]	The article describes a tool-supported prioritization process for quality requirements based on quantitative quality evaluation of software architecture models
[S63]	The article proposes an approach for automating a significant part of the prioritization process by utilizing a probabilistic traceability model and a standard hierarchical clustering algorithm
[S71]	The article presents a systematic and operational method based on the integration of a minimal deviation-based method (MDBM), balanced scorecard (BSC), analytic hierarchy process (AHP) and scale method to determine the final priority ratings of customer requirements
[S75]	The article proposes StakeRare, a novel method that uses social networks and collaborative filtering to identify and prioritize requirements in large software projects
[S76]	The article presents a formal framework to aid the decision making in prioritizing requirements in a software development process, including ambiguous and vague data

Table 17 continued

ID	Brief objective of the article
[S77]	The article presents a framework that prioritizes software requirements gathered from multiple stakeholders by incorporating inter-perspective relationships
[S80]	The article describes a priority-based approach in order to prioritize requirements obtained from multiple sources toward composing requirement specifications
[S81]	The article presents a practical approach for requirements elicitation and prioritization based on realistic user behaviors observation
[S82]	The article specifies the fundamental requirements for a prioritization process and describes a framework that incorporates many aspects of prioritizing requirements
[S85]	The article proposes an agile technique for prioritizing features in environments with multiple stakeholders and reports a successful experience in its usage
[S86]	The article proposes a novel framework for prioritizing requirements in software projects based on derivation of quality measurements
[S87]	The article proposes an interactive approach to the problem of prioritization based on Satisfiability Modulo Theory (SMT) techniques and pairwise comparisons
[S88]	The article proposes a system called the Distributed Collaboration Priorities Tool (DCPT) to assist in collaborative prioritization of development items in software projects
[S91]	The article introduces a method that derives latent semantic centrality and relative information specificity scores from requirements and calculates relative priorities of individual requirements and semantically coherent groups of requirements
[S96]	The article highlights some serious shortcomings related to existing requirement prioritization techniques and proposes an intelligent fuzzy logic-based technique for requirements prioritization based on the perceived value of each requirement
[S97]	The article presents an industrial case study where a distributed prioritization process is proposed, observed, and evaluated
[S99]	Mentioned in Table 16 above
[S100]	The article addresses the idea of applying different models during requirement prioritization so as to take into account all possible idiosyncrasies of business process-driven software development
[S107]	The article describes an algorithm based on sample selection for prioritizing requirements. The algorithm selects a subset of items to be presented to the next participant based on previous responses, the size of the requirements set, and the number of customers
[S109]	The article describes instruments for distributed priority ranking of strategic preliminary requirements for MISs in organizations, profit making or nonprofit making, that are involved in the economy
[S110]	Mentioned in Table 16 above
[S113]	The article presents a quantitative framework involving respondents of both the project and the business organization, in order to determine the priority of a list of NFRs to be considered for implementation during software development
[S114]	The article uses an algorithm to produce a requirement ordering which complies with the existing priorities, satisfies the technical constraints, and takes into account the relative preferences elicited from the user
[S115]	The article proposes an Interactive Genetic Algorithm (IGA) that includes incremental knowledge acquisition and combines it with the existing constraints, such as dependencies and priorities
[S117]	The article investigates an approach for software developers to deal in a formal and systematic way with software product quality, in particular the specification, the prioritization and the metrication of software quality
[S120]	The article presents a method of arriving at a measure of requirement priority by taking into consideration benefit, penalty, cost, and risk associated with the individual requirements
[S122]	The article presents an algorithm to arrive at a priority ranking of software requirements based on considerations of the type of requirements, the role who proposes requirements, the frequency of requirements proposed and the time of requirements proposed
[S123]	The article presents a method of ranking the importance degree of requirements in quality function deployment (QFD), by regarding QFD as a two-person zero-sum game with an uncertain payoff matrix
[S127]	The article presents an approach based on the hybrid assessment method (HAM) to achieve prioritization of functional requirements and NFRs simultaneously during software development process
[S128]	The article describes a new goal-based prioritization technique taking into account the relative weight of the requirements with respect to the goals identified by the different stakeholders
[S130]	The article introduces a prioritization technique based on the generalized Yager's algorithm with the aim of fusing the preference orderings by multiple respondents into a single ordering
Requirement selection	
[S24]	The article focuses on designing, implementing, and evaluating a support tool for release planning as a means for provoking a rich understanding of the task of release planning
[S35]	The article develops a decision analysis method that assists in making trade-offs in requirement selection in the absence of quantitative data
[S36]	The article describes a novel approach to converging upon near-optimal attainment of requirements in large-scale requirements models

Table 17 continued

ID	Brief objective of the article
[S42]	Mentioned in Table 16 above
[S61]	The article presents a two-step value-based approach for selecting the requirements based on TOPSIS, a decision analysis framework that tightly integrates decision theory with the process of requirements prioritization
[S72]	The article investigates two integer linear programming models that integrate requirement scheduling into software release planning
[S73]	The article investigates techniques to prioritize and predict NFRs to facilitate their selection in the early stages of agile software requirements gathering along with the functional requirements
[S89]	The article discusses author's experience working with a large healthcare company writing software for use in their hospital's new-born intensive care unit (NICU)
[S102]	The article proposes a method known as Quantitative WinWin which uses an evolutionary approach to provide support for requirements negotiations
[S103]	Mentioned in Table 16 above
[S124]	Mentioned in Table 16 above
[S134]	The article proposes a requirement prioritization technique to support the selection of candidate requirements for value-based software (VBS) development
[S135]	The article presents a mathematical formulation of the Next Release Problem (NRP) that uses requirement prioritization modeling to select requirements for the next releases
Categorizing requirements	
[S26]	The article discusses how to group requirements into "chunks" of revenue-generating functionality known as Minimal Marketable Features (MMFs), and how to carefully sequence those MMFs in order to maximize the overall value of the project
[S38]	The article discusses categorizing requirements based on a TQM-style process for prioritizing initiatives
[S40]	The article introduces a fuzzy decision model named as Multi-Person Decision-Making Model to negotiate and integrate the requirements of various stakeholders
[S55]	The article presents a cost-value approach for classifying requirements on a cost-value diagram for management decision making
[S119]	The article discusses the MoSCoW method in which the items are roughly classified in priority groups depending on importance. The letters stay for: M—MUST have this, S—SHOULD have this if at all possible, C—COULD have this if it does not affect anything else, W—WON'T have this time but would like in the future
[S125]	Mentioned in Table 16 above
Requirement value assessment	
[S27]	The article discusses the QFD approach for developing a design quality aiming at satisfying the consumer and then translating the consumer's demand into design targets and major quality assurance points to be used throughout the production phase
[S33]	The article discusses a methodology designed to appraise and measure the users' (future) satisfaction with identified requirements in the requirements engineering phase
[S106]	The article presents a fuzzy quality function deployment approach for determining which of the non-functional requirements reported by earlier studies are important to a company's software selection decision based on and integrated with its functional requirements
[S118]	The article applies the Interval Evidential Reasoning (IER) algorithm in order to assess the degree of satisfaction with the requirements
[S126]	The article discusses an approach based on story mapping to inject value thinking into feature prioritization
Others	
[S15]	The article analyzes the efficiency of a requirement prioritization approach using B-Tree in prioritizing requirements
[S16]	The article investigates attributes of prioritization techniques that affect stakeholders' threshold for judging essential product features
[S22]	The article presents an extension of an existing prioritization method in order to explicitly account for the stakeholder value and to cater to multiple stakeholders
[S56]	Mentioned in Table 16 above
[S60]	The article describes a minimum spanning tree (MST) of an edge-weighted graph as a spanning tree whose weight (the sum of the weights of its edges) is no larger than the weight of any other spanning tree. The MST technique has been used to achieve prioritization of software requirements
[S74]	The article discusses on representation and reasoning about preferential requirements ("nice-to-have") in requirements engineering
[S78]	The article proposes a statistical methodology to manage uncertainty in the planning of what functionality to include in upcoming releases in agile projects
[S83]	Mentioned in Table 16 above
[S108]	Mentioned in Table 16 above
[S133]	Mentioned in Table 16 above

Table 18 Requirement prioritization artifacts

ID	Article type	Source	Artifact type	Artifact name
[S1]	Conference	IEEE Xplore	Framework	<Perform prioritization by combining existing techniques and approaches>[a]
[S4]	Book	Article [S51]	Method	Bubble Sort
[S5]	Conference	IEEE Xplore	Tool	<Based on artifact traceability information>[a]
[S6]	Conference	IEEE Xplore	Framework, Tool	Case-Based Ranking (CBRank) Framework
[S7]	Journal	ACM Digital Library	Framework	Value-Oriented Prioritization (VOP)
[S9]	Conference	Springer	Method, Tool	Stratified Analytic Hierarchy process (S-AHP)
[S12]	Conference	REFSQ Proceedings	Method, Tool	Binary Priority List (BPL)
[S13]	Book	Article [S45]	Method	Planning Game
[S14]	Journal	ScienceDirect	System	Priority Handler (PHandler)
[S15]	Conference	IEEE Xplore	Method	<Approach using B-Tree>[a]
[S16]	Symposium	IEEE Xplore	Method, Tool	Pairwise comparisons & ranking, Simple Dropdown, Drag into bins, Sortable table
[S17]	Journal	Springer	Method	Hierarchical Cumulative Voting (HCV)
[S19]	Book Chapter	Springer	Method	Ranking
[S20]	Journal	ScienceDirect	Method	<Approach that Augments HCV>[a]
[S21]	Website	Article [S95]	Method	Round-the-group prioritization
[S22]	Book Chapter	Article Pergher and Rossi [10][b]	Method	<Extension of impact estimation prioritization method>[a]
[S23]	Journal	ACM Digital Library	Methodology	Fuzzy AHP
[S24]	Journal	Article [S19]	Method, Tool	Release Planner Prototype (RPP)
[S26]	Conference	IEEE Xplore	Method	Incremental funding method (IFM)
[S27]	Conference	Article [S95]	Method	Quality functional deployment QFD
[S28]	Conference	IEEE Xplore	Framework	<Value-oriented Framework>[a]
[S33]	Conference	IEEE Xplore	Methodology, Tool	AMUSE (appraisal and measurement of user satisfaction)
[S34]	Journal	ACM Digital Library	Method	<Data mining and machine learning-based technique>[a]
[S35]	Journal	ACM Digital Library	Method	<Heuristic algorithm>[a]
[S36]	Conference	Article [S19]	Method, Tool	<Requirement optimization approach>[a]
[S38]	Website	Article [S95]	Method	Ranking based on product definition
[S39]	Conference	IEEE Xplore	Method	<Method for ranking requirements>[a]
[S40]	Conference	IEEE Xplore	Model	Multi-Person Decision-Making Model
[S41]	Website	Article [S95]	Method	Pair-wise analysis, Weighted criteria analysis, Dot voting
[S42]	Journal	Article [S19]	Method	Evolutionary and Iterative (EVOLVE) Approach
[S43]	Journal	Springer	Method	CCR″ technique (where First C—Customer satisfaction, C—cost, R—Regression count)
[S46]	Journal	ACM Digital Library	Method	<Risk-based Requirement Prioritization>[a]
[S47]	Conference	IEEE Xplore	Model	<Model for requirement prioritization using AHP>[a]
[S48]	Thesis	Article [S51]	Method	<AHP-based approach>[a]
[S49]	Conference	IEEE Xplore	Method	Numerical Assignment (Grouping)
[S50]	Journal	Springer	Method, Tool	<Improvement of an existing cost-value approach>[a]
[S51]	Journal	ACM Digital Library	Method	Priority Groups, Hierarchy AHP
[S55]	Journal	ACM Digital Library	Method	Cost-value approach
[S56]	Journal	Article [S45]	Method	Weighting Methods
[S57]	Book	Article [S45]	Method	Multi-Attribute Utility Theory (MAUT)
[S58]	Book	Article [S45]	Method	<Prioritization approach based on use cases and quality models>[a]
[S59]	Conference	REFSQ Proceedings	Method	<Architecture-driven method>[a]
[S60]	Journal	Article [S51]	Method	Minimal Spanning Tree

Table 18 continued

ID	Article type	Source	Artifact type	Artifact name
[S61]	Conference	IEEE Xplore	Framework, Tool	Technique of Ordered Preference by Similarity to Ideal Solution (TOPSIS)
[S63]	Conference	IEEE Xplore	Method	<Using standard hierarchical clustering algorithm>[a]
[S65]	Book	Article [S19]	Method	Top Ten
[S71]	Journal	ACM Digital Library	Method	<Integrates minimal deviation-based method (MDBM), balanced scorecard (BSC), analytic hierarchy process (AHP), scale method>[a]
[S72]	Conference	REFSQ Proceedings	Model	<Integer linear programming models>[a]
[S73]	Conference	IEEE Xplore	Framework	$\alpha\beta_{\gamma}$-Framework
[S74]	Journal	ACM Digital Library	Constructs	<Extends traditional goal modeling notation>[a]
[S75]	Journal	ACM Digital Library	Method	StakeRare
[S76]	Conference	Springer	Framework	<Formal framework using fuzzy logic>[a]
[S77]	Conference	IEEE Xplore	Framework	<Framework incorporating interperspective relationships>[a]
[S78]	Conference	IEEE Xplore	Methodology	<Statistical methodology that allows for uncertainty>[a]
[S80]	Journal	Springer	Method	<Priority-based approach>[a]
[S81]	Conference	IEEE Xplore	Method	<Log-based approach>[a]
[S82]	Conference	SETE Proceedings	Framework, Tool	<Framework & Tool:>[a] Requirement Prioritization Tool (RPT)
[S83]	Journal	ACM Digital Library	Framework	<Prioritized merging-based framework>[a]
[S84]	Book	Article [S45]	Method	Cost–Benefit Analysis
[S85]	Conference	ACM Digital Library	Method	<Agile Technique>[a]
[S86]	Conference	IEEE Xplore	Framework	<Quality-Based Requirement Prioritization Framework>[a]
[S87]	Conference	ACM Digital Library	Method	<Interactive Approach based on Satisfiability Modulo Theory (SMT) techniques and pairwise comparisons>[a]
[S88]	Conference	IEEE Xplore	System, Tool	Distributed Collaboration Priorities Tool (DCPT)
[S89]	Conference	Article [S95]	Method	Bucketing requirements
[S91]	Conference	ACM Digital Library	Method	<Using latent semantic centrality and relative information specificity scores of requirements>[a]
[S96]	Conference	IEEE Xplore	Method	<Intelligent fuzzy logic-based technique>[a]
[S97]	Journal	ACM Digital Library	Method	Cumulative Voting (Hundred-dollar test)
[S99]	Journal	ScienceDirect	Method	Hybrid Assessment Method (HAM)
[S100]	Conference	IEEE Xplore	Model, Framework	<BPRE (Business Process Requirement Engineering) Prioritization Framework>[a], Value Model, Issue Model, Stakeholder/Role Model
[S101]	Book	Article [S45]	Method	Outranking
[S102]	Journal	Springer	Method	Quantitative WinWin
[S103]	Conference	Springer	Method	<priority-based decision heuristics>[a]
[S104]	Book	Article [S51]	Method, Tool	Analytic Hierarchy Process (AHP)
[S105]	Book	Article [S95]	Method	Ping-Pong Balls
[S106]	Journal	ACM Digital Library	Methodology	<fuzzy quality function deployment approach>[a]
[S107]	Conference	IEEE Xplore	Method	Sample Selection
[S108]	Symposium	ACM Digital Library	Method	Attribute Hierarchy-based Evaluation of Architectural Designs (AHEAD) method
[S109]	Journal	ScienceDirect	Methodology, Tool	<Distributed priority ranking instrument>[a]
[S110]	Journal	ScienceDirect	Framework	<SPI framework based on the CMMI using QFD>[a]
[S112]	Book	Article [S95]	Method	Multi-voting system
[S113]	Journal	Springer	Framework	<NFR Prioritization Framework>[a]
[S114]	Symposium	ACM Digital Library	Method	<Interactive Optimization of Requirements>[a]
[S115]	Journal	ScienceDirect	Method	Interactive Genetic Algorithm (IGA)
[S117]	Journal	Springer	Method	<Approach based on ISO9126 and AHP>[a]

Table 18 continued

ID	Article type	Source	Artifact type	Artifact name
[S118]	Conference	Springer	Method	Interval Evidential Reasoning (IER) Algorithm
[S119]	Website	Article [S95]	Method	MoSCoW (M: Must, S: Should, C: Could, W: Won't)
[S120]	Journal	Article [S19]	Method	Prioritization Matrix
[S122]	Conference	IEEE Xplore	Method	<Algorithm for ranking priority of individualized functional requirements dynamically>[a]
[S123]	Conference	IEEE Xplore	Method	<Fuzzy game approach>[a]
[S124]	Journal	ACM Digital Library	Method	<Multi-objective evolutionary optimization algorithms>[a]
[S125]	Conference	IEEE Xplore	Method, Tool	<Approach and tool support for prioritization by automatically grouping similar requirements>[a]
[S126]	Journal	IEEE Xplore	Method	<Value-based feature prioritization>[a]
[S127]	Conference	IEEE Xplore	Method, Tool	<Approach and tool implementation based on Hybrid Assessment Method (HAM)>[a]
[S128]	Conference	IEEE Xplore	Method	<Goal-based requirement prioritization technique>[a]
[S130]	Journal	Springer	Method	<Prioritization technique based on generalized Yager's algorithm>[a]
[S133]	Conference	IEEE Xplore	Method	<AHP technique implemented in an Agile RE tool>[a]
[S134]	Conference	IEEE Xplore	Method	<Multi-aspect requirements prioritization technique for value-based software>[a]
[S135]	Conference	IEEE Xplore	Model	<Requirement prioritization modeling of additive and non-additive valuation conditions>[a]

<…>[a]Exact names of these prioritization artifacts have not been specified by the authors

[b] Pergher and Rossi [10] is a workshop article and hence not included in the final list of reviewed articles

Table 19 Description of factors influencing software requirement prioritization

Factor	Description
Requirement type	This refers to the various classes of requirements, for example, functional, non-functional, database, interface, derived, design, etc
Requirement risk	Requirements risks are risks that are associated directly with specific requirements. Certain requirements may open up risks of regulatory non-compliance, legal issues, PR issues, unexpected costs or process bottlenecks, etc
Requirement complexity	This refers to the difficulty associated with requirement identification, specification and processing
Requirement importance/priority/preference	This reflects the importance that has been assigned to individual requirements or specific requirement classes, for example, importance assigned to the class of functional requirements, etc
Requirement volatility/stability	Requirement volatility refers to the changes (addition, modification, or deletion) to the initially specified requirements set during project development
Requirement performance	Requirement performance describes how well a function is to be executed or achieved, or how well it is to be accomplished
Requirement value	Requirement value refers to some kind of utility of the requirements to its intended stakeholders. The notion of value may be in qualitative or in quantitative terms
Requirement cost	This indicates the development or life-cycle costs associated with implementing the concerned requirement
Requirement benefit	Requirement benefit refers to the advantages provided by the requirement. These advantages can be in monetary terms (e.g., the expected profits from implementing the requirements, etc.) or non-monetary terms (e.g., stakeholder satisfaction, etc.)
Requirement hierarchy	In this case, the requirements are arranged in a parent–child relationship generally forming a tree structure. The top-level requirements represent the general requirements and these are linked to the more specific requirements located at the bottom
Requirement aspects	Requirement aspects relate to the various technical and business attributes that are related to the prioritization process. The technical aspects include risk, value, cost, speed, effort, granularity, time, sophistication, dependencies, sensitivity, contradictory, volatility, penalty, resources and complexity while the business aspects include sales, marketing, competitive, strategic, customer retention, simplicity, innovative, resourceful, client focused and availability

Table 19 continued

Factor	Description
Quality of requirements	Quality is a measure of the clarity associated with the specified requirements. The characteristics that are used to evaluate requirement quality are: atomic, complete, traceable, logical and clear, consistent, measurable, compliant, feasible, necessary, and prioritized
Importance of views	The views here refer to the subjective evaluation of requirements by stakeholders. A priority ranking of these views has been referred as the importance of the views
Development cost	This is a measure of the cost associated with developing the system constituted of the identified requirements
Development benefit	This indicates a measure of expected benefits that can be derived from the system constituted of the identified requirements
Project duration (delivery date)	This refers to the duration of the project (delivery date is the calendar date of implementation of the project)
Effort	Effort indicates the person-day equivalent that is needed (or expended) to complete the project
Project size	Project size is a measure of the amount of work to be accomplished in order to complete the project in concern
Resources (competencies)	The workforce that is required or that has been assigned to work on the project involving the identified requirements
Project goal	The project goal is a statement that describes what the *project* will accomplish, or the business value the *project* will achieve
Project costs	This is a measure of costs that are likely to be incurred or has been incurred in implementation of the project constituted of the identified requirements
Project expectations	This relates to specific expectations connected to the project in concern. These expectations could be related to process, outcome, or environment charactering the project and having an influence on the project's requirements
Project influence	This refers to the influence exerted by specific stakeholders on the project's requirements as determined by the stakeholders' position on the social network
Project constraints	Constraints represent restrictions on specific parameters having an influence on the desired outcome. This can be in the form of restrictions on project's schedule, effort, quality, budget, etc
Project contexts	Project context defines the environment governing a project. Projects differ based on the type of the project, the size of the project, the stakeholders associated with the project, the standard operating procedures, etc
Application domain	This refers to the field of application of the concerned software project, for example Avionics, IT, Command & Control, Process Control, Real Time, Scientific, System Software, Telecommunication, etc
Requirement dependencies	This refers to the possible dependencies that might exist among the requirements constituting a project, for example, NFR operability (i.e., the state of being able to perform the intended functions) may negatively influence security, etc
Requirement trade-offs	Requirement trade-offs refer to the relative considerations of the requirements considered for prioritization. Certain requirements might have more preference over others given the trade-off considerations
Implementation dependencies	When developing systems, certain components of the system depend on the other components (often foundational and infrastructure components) of the system and should be implemented upfront. Such kind of specifications of the implementation order is implied here
Artifact traceability	Artifact traceability is the degree to which a relationship can be established between two or more artifacts of the development process, especially artifacts having a predecessor–successor or master–subordinate relationship with one another; for example, the degree to which the requirements and design of a given software component match
Legal complexity/mandate	The complexities in the form of laws and regulations governing the specific software system being planned, which influence the legal requirements considered for prioritization
Response time	It is the measure of time interval between an input and the receipt of an action, result, or feedback from the concerned system implementing the identified requirements
Regression count	The number of dependencies between parts of one requirement and parts of another one is referred to as regression count
Software architecture and relevant attributes	Software architecture can be defined as the set of structures needed to reason about the software system, which comprise the software elements, the relations between them, and the properties of both elements and relations
Requirement baselining	Requirement baselining refers to the act of freezing requirements at a point generally keeping in mind some planned release in the future. Any further changes to the requirements are then addressed as part of subsequent releases
Technical debt	Technical debt implies the amount of architecture redesign-related work specific to a software, which accumulates over a period of time

Table 19 continued

Factor	Description
System impact	The term system impact has been used to refer to the impact of the product or service characteristics on its intended beneficiaries
Requirement usability	Usability refers to making products and systems easier to use, and matching them more closely to user needs and requirements
Frequency of use	This refers to the expected frequency or volume of usage of requirements
Quality attributes/characteristics	This refers to the specific indicators of quality characterizing the product or service that has been proposed or delivered
Penalty	Penalty is a representation of loss related to the absence of feature(s) from the product/service under consideration
Harm avoidance	This refers to the consideration of the harm that can or will occur if the concerned requirement(s) is not implemented in the project. This would especially be true of safety and security requirements, which are specifically specified to avoid accidental and malicious harm to valuable assets due to hazards and threats of attack, respectively
Requirement's issuer	The stakeholder group with the responsibility of identifying and specifying requirements to be subsequently considered for implementation in a project
Requirement hits	This refers to the total number of accesses to specific requirements, which are accessible as product/service characteristics to its intended users
Number of unique users	This refers to the number of different users that had accessed at least once a specific web resource over a designated timeframe
Multiple groups	The different stakeholder groups that are associated with the identified concerns related to prioritizing project requirements
Stakeholder roles	The prescribed or expected behaviors of stakeholders in regard to the concerned project
Stakeholder's relative importance	This indicates the importance assigned to specific stakeholders or specific stakeholder groups in regard to the identified objectives
Stakeholder preferences	This refers to preferences of the individual stakeholders on the choice of requirements. This is especially true when practical reasons such as schedule and budget mean that all of the requirements cannot be implemented and released all at one time
Stakeholder perspectives	This refers to the viewpoints of a stakeholder or a stakeholder group associated with the concerned project
Business values	Business value refers to all forms of value that determine the health and well-being of an organization in the long run. Business value expands the concept of value of the organization beyond economic value (i.e., economic profit) to include other forms of value such as employee value, customer value, supplier value, channel partner value, alliance partner value, managerial value, and societal value
Business goals	Business goals, sometimes referred to as mission statements, are the things an organization hopes to achieve during its time in operation
Organization size	This is an indicator of the size of the organization based on some acceptable measure, for example number of employees on its payroll
Time to market	Time to market (TTM) is the length of time it takes from a product or service being conceived until its being available for sale
Competitive benchmarking	Competitive benchmarking refers to the act of comparing the product or service characteristics with competitive offerings in order to identify possible improvement opportunities

Table 20 List of journals, conferences, and symposiums covered in the review

Conferences:	*Symposiums*:
ACM International Conference Companion on Object Oriented Programming Systems Languages and Applications Companion	ACM Symposium on Applied Computing
AGILE Conference	ACM-IEEE International Symposium on Empirical Software Engineering and Measurement
Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications	ACM/IEEE International Symposium on Empirical Software Engineering
Annual International Computer Software and Applications Conference	International Symposium on Empirical Software Engineering
Annual Quality Congress Proceedings	International Symposium on Search Based Software Engineering
Annual Southeast Regional Conference on Software Engineering	Symposium on Assessment of Quality Software Development Tools
Asia Pacific Software Engineering Conference	*Journals*:
Australasian Software Engineering Conference	Computer
Empirical Assessment in Software Engineering	Computer Standards & Interfaces
EUROMICRO Conference on Software Engineering and Advanced Applications	Decision Support Systems
European Conference on Foundations of Software Engineering	Empirical Software Engineering
European Conference on Software Maintenance and Reengineering	Expert systems with Applications
European SharePoint Conference	Fundamenta Informaticae
IEEE Conference on Software Engineering Education and Training	Human-centric Computing and Information Sciences
IEEE International Conference and Workshops on Engineering of Computer Based Systems	IEEE Software
IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing	IEEE Transactions on Software Engineering
IEEE/ACM International Conference on Automated Software Engineering	IIE solutions
India Software Engineering Conference	Information and Software Technology
International Requirements Engineering Conference	International Journal of Software Engineering and Knowledge Engineering
International Advance Computing Conference	Journal of Object Technology
International Conference on Computational Intelligence and Software Engineering	Journal of Systems and Software
International Conference on Computational Science and Technology	Knowledge-Based Systems
International Conference on Computer Networks and Information Technology	Proceedings of the American Mathematical society
International Conference on Computer Science and Automation Engineering	Requirements Engineering
International Conference on Computer Technology and Development	Research in Engineering Design
International Conference on Data Storage and Data Engineering	Software Development
International Conference on Emerging Technologies	Software Process: Improvement and Practice
International Conference on Grid and Cooperative Computing	Software Quality Journal
International Conference on Informatics and Systems	
International Conference on Information and Emerging Technologies	
International Conference on Information Management and Engineering	
International Conference on Information Science and Applications	
International Conference on Information Systems Design and Intelligent Applications	
International Conference on Innovative Computing, Information and Control	
International Conference on Management Science & Engineering	
International Conference on Mathematical/Analytical Modelling and Computer Simulation	
International Conference on Research Challenges in Information Science	
International Conference on Software Engineering	
International Requirements Engineering Conference	
Requirements Engineering: Foundation for Software Quality	
Search Based Software Engineering	
Software Product Lines: Going Beyond	
SoutheastCon Conference	
Systems Engineering, Test and Evaluation Conference	

References

1. Herrmann A, Daneva M (2008) Requirements prioritization based on benefit and cost prediction: an agenda for future research. In: 16th IEEE international requirements engineering (RE'08). IEEE, pp 125–134
2. Kotonya G, Sommerville I (1998) Requirements engineering: processes and techniques. Wiley, Chichester
3. Committee ICSSES, Electronics Engineers I, Board I-SS (1998) IEEE recommended practice for software requirements specifications: approved 25 June 1998, vol 830. IEEE
4. Karlsson J, Wohlin C, Regnell B (1998) An evaluation of methods for prioritizing software requirements. Inf Softw Technol 39(14):939–947
5. Achimugu P, Selamat A, Ibrahim R, Mahrin MNR (2014) A systematic literature review of software requirements prioritization research. Inf Softw Technol 56(6):568–585
6. Kukreja N, Payyavula SS, Boehm B, Padmanabhuni S (2012) Selecting an appropriate framework for value-based requirements prioritization: a case study. In: 20th IEEE international requirements engineering conference, Chicago, IL, USA
7. March ST, Smith GF (1995) Design and natural science research on information technology. Decis Support Syst 15(4):251–266
8. Khan KA (2006) A systematic review of software requirements prioritization. Blekinge Institute of Technology, Ronneby
9. Babar MI, Ramzan M, Ghayyur SAK (2011) Challenges and future trends in software requirements prioritization. In: International conference on computer networks and information technology (ICCNIT). IEEE, pp 319–324
10. Pergher M, Rossi B (2013) Requirements prioritization in software engineering: a systematic mapping study. In: IEEE, pp 40–44
11. Perini A, Susi A, Ricca F, Bazzanella C (2007) An empirical study to compare the accuracy of AHP and CBRanking techniques for requirements prioritization. In: Fifth international workshop on comparative evaluation in requirements engineering. IEEE, pp 23–35
12. Pitangueira AM, Maciel RSP, Barros M (2015) Software requirements selection and prioritization using SBSE approaches: a systematic review and mapping of the literature. J Syst Softw 103(May):267–280
13. Riegel N, Doerr J (2015) A systematic literature review of requirements prioritization criteria. In: Fricker SA, Schneider K (eds) Requirements engineering: foundation for software quality. Springer, Berlin, pp 300–317
14. Sher F, Jawawi DNA, Mohamad R, Babar MI (2014) Requirements prioritization techniques and different aspects for prioritization a systematic literature review protocol. In: 8th Malaysian software engineering conference. IEEE, pp 31–36
15. Budgen D, Turner M, Brereton P, Kitchenham B (2008) Using mapping studies in software engineering. In: Proceedings of the PPIG. Lancaster University, pp 195–204
16. Petersen K, Feldt R, Mujtaba S, Mattsson M (2008) Systematic mapping studies in software engineering. In: 12th International conference on evaluation and assessment in software engineering (EASE), p 1
17. Downeâ-Wamboldt B (1992) Content analysis: method, applications, and issues. Health Care Women Int 13(3):313–321
18. Higgins JPT, Green S (2011) Cochrane handbook for systematic reviews of interventions version 5.1. 0 [updated March 2011]. The Cochrane Collaboration
19. Kitchenham B, Pretorius R, Budgen D, Pearl Brereton O, Turner M, Niazi M, Linkman S (2010) Systematic literature reviews in software engineering: a tertiary study. Inf Softw Technol 52(8):792–805
20. Berander P, Andrews A (2005) Requirements prioritization. In: Aurum A, Wohlin C (eds) Engineering and managing software requirements. Springer, Berlin, pp 69–94
21. Cavanagh S (1997) Content analysis: concepts, methods and applications. Nurse Res 4(3):5–13
22. Miles MB, Huberman AM (1994) Qualitative data analysis: an expanded sourcebook. Sage Publications, Thousand Oaks, CA
23. Hevner AR, March ST, Park J, Ram S (2004) Design science in information systems research. MIS Q 28(1):75–105
24. Offermann P, Blom S, Schönherr M, Bub U (2010) Artifact types in information systems design science—a literature review. In: Winter R, Zhao JL, Aier S (eds) Global perspectives on design science research. Springer, Berlin, pp 77–92
25. Berge C (1958) La theorie des graphes, vol II. Collection Universitaire de Mathématiques. Dunod, Paris
26. Köksalan MM, Wallenius J, Zionts S (2011) Multiple criteria decision making: from early history to the 21st century. World Scientific Publishing Co. Pte. Ltd., Singapore
27. Xu D-L, Yang J-B, Wang Y-M (2006) The evidential reasoning approach for multi-attribute decision analysis under interval uncertainty. Eur J Oper Res 174(3):1914–1943
28. Aasem M, Ramzan M, Jaffar A (2010) Analysis and optimization of software requirements prioritization techniques. In: International conference on information and emerging technologies (ICIET). IEEE, pp 1–6
29. Moisiadis F (2002) The fundamentals of prioritising requirements. In: Proceedings of the systems engineering, test and evaluation conference (SETE'2002). Citeseer, pp 108–119
30. Svensson RB, Gorschek T, Regnell Br, Torkar R, Shahrokni A, Feldt R, Aurum AK (2011) Prioritization of quality requirements: state of practice in eleven companies. In: IEEE, pp 69–78
31. Gregor S (2009) Building theory in the sciences of the artificial. Paper presented at the 4th international conference on design science research in information systems and technology, Philadelphia, PA, USA, May 7–8, 2009
32. Riedl C (2011) Tool-supported innovation management in service ecosystems. Gabler Verlag, Wiesbaden
33. Davis A (2013) Just enough requirements management: where software development meets marketing. Addison-Wesley, New York