



Current State of Practice in Software Requests for Proposals 2021

Prepared by:

Brian Lines, PhD, PE; Phuong Nguyen, PhD, and Omar Maali, MS
Civil, Environmental & Architectural Engineering (CEAE) Department
University of Kansas

PROCUREMENT & SUPPLY CHAIN

Current State of Practice in Software Requests for Proposals

Prepared by:

Brian Lines, PhD, PE, Phuong Nguyen, PhD, and Omar Maali, MS
Civil, Environmental & Architectural Engineering (CEAE) Department
University of Kansas

2021

The request for proposal (RFP) process is essential to the subsequent success of software project implementations. The RFP seeks to identify an appropriate software solution to best meet the client organization's scope and contractually establishes the project's baseline cost and schedule expectations. However, there is a lack of RFP-research related to the current state of practice in software projects. The objective of this study was to address this gap via a content analysis of 250 recent software RFPs across commonly implemented software categories. Results identified the most frequently used evaluation criteria and corresponding weights, procurement schedule durations, anticipated length of contract terms, quantity of itemized requirements documented in the owner's scope, and other pertinent information. Inferential testing found several differences among the different software categories. The findings may be helpful to owner project managers who are tasked with leading their project teams through the early stages of scope development and vendor selection.

Introduction

Software projects are often viewed as representing “risky” endeavors due to their documented potential for poor performance outcomes (Agiloft, 2019; Flyvbjerg and Budzier, 2011). Therefore, many studies, both academic and industry-lead, have sought to measure software project performance.

In one analysis, the Standish Group compiled software projects for three decades in their so-called CHAOS reports, wherein their data showed that roughly 46 percent of software projects “challenged” which is defined as projects that are complete and operational but over-budget, over-schedule, and offer fewer features than specified (Johnson, 2018). Additionally, 26 percent of their documented projects are “failed”, indicating that they are cancelled at some point or not used after being implemented. It is noted that the software projects are new and involved modern technologies which makes the projects highly complex and require advanced managerial skills (Abouzahra, 2011; Salah et al., 2017).

The challenges that software projects face include the need to meet end user requirements, achieve client satisfaction with the user experience, and avoid cost and schedule overruns. Many of these challenges have been tied to early project delivery stages of project scoping, project planning, selection and involvement of external vendors, and lack of early user input (Agiloft, 2019; Alami, et al. 2016). For instance, in the previously mentioned analysis from the Standish Group, the main factors that contributed to such challenged and failed projects were found to include a lack of user input, incomplete scope requirements and specifications, and improper planning, all of which are challenges encountered early in the project development process (Standish, 2015). Other studies have come to similar conclusions. For example, Kappelman et al. (2007) identified 50+ observable early warning signs of failed software projects. These early warning signs were organized into three main categories related to people-, process-, and product-risks. Their conclusion was to identify a “dominant dozen” of the most impactful early warning signs, all of which were linked to people- and process-related elements that presumably can be managed with concerted up-front planning and subsequent control actions. Agarwal and Rathod (2006) found that software functionality, cost, and time are the most critical factors that affect the performance of software projects; therefore, well-defined project goals and well-written requirements are among the most important factors to improve the acquisition and implementation processes of software projects.

One of the major steps in the early stages of software project delivery is the procurement process. The software procurement process is challenging and important to project success (Hassan et al., 2018; Jørgensen et al., 2017). The initial expectation of the owner's statement of work (SOW), project objectives, and other requirements are typically compiled and included in a request for proposal (RFP). The content of the RFP therefore sets the initial benchmark of project SOW expectations that will be utilized by the proposing vendors when developing their proposal responses (Moe, 2014, 2017; Cragg and Chraibi, 2020). The RFP is also the mechanism by which the owner will evaluate and select the specific software vendor (and associated product) that is perceived to be the "best" or "most optimal" to meet the owner's needs (Fayaz et al., 2017; Saito, et al. 2012). Therefore, the RFP phase is instrumental in establishing the major project performance parameters of (a) the owner's expectation of the project SOW, (b) the owner's selection of a vendor partner to provide the product and associated implementation services, and (c) the vendor's corresponding cost and schedule proposals, all which will serve as the foundation of the contractual expectations of project performance.

Low quality RFPs can have a "trickle-down" effect that results in problems during the project execution and implementation phases. For example, a gap between owner's requirements and the software vendor's understanding of those requirements is a frequent cause of change orders that can increase the project cost and delay the schedule (Wagner and Lederer, 2004). Johansson and Lahtinen (2012) indicated that the software procurement process varies across different project types, which creates the variation in software RFPs. Yet regardless of the software type, RFPs are a critical point in early software delivery because the vendor proposal responses and associated cost proposals are based on the client's SOW, which has oftentimes can be unclear (Johnson, 2018; Kappelman et al., 2007; Zaman et al., 2019). If it is based on a faulty RFP process, the selected vendor's cost proposal may set a false expectation in the project's contractual requirements. The sales and marketing information provided by software vendors during the RFP process can also set the client's expectation around the level of SOW and functionality each vendor will be able to deliver; however, that sales information is not always a reliable measure of what is realistically achievable within the client's environment (Roslina, 2013).

Given the high rate of failure that is documented in the information technology industry and the stated importance of early project stages such as the software procurement process, it is important to understand the current state of practice in software project RFPs. Such information will enable practitioners to identify trends, strengths, and weaknesses in the current state of practice. The objective of this study was therefore to identify the characteristics and differences in several RFP elements (such as their stated evaluation criteria, procurement timing, and level of itemized requirements) across five common software project categories, including enterprise resource planning (ERP), financial systems, asset management systems, common business applications (CBA), and specialized business applications (SBA). The study aimed to analyze a large number of software RFPs to support project owners in understanding the current state of practice in software procurement and identify opportunities to improve future RFP efforts.

Literature Review

Software Project Performance

Studies of software project performance have found challenges in terms of not meeting end users' goals and requirements as well as finishing over budget and with schedule overruns (Belfo and Trigo, 2013; Fayaz et al., 2017; Johansson and Lahtinen, 2012; Moe, 2014; Thomas and Fernández, 2008). Several studies have documented high rates of project cancellation or deployment of products that are ultimately not utilized by end users (Alami, 2016, Fayaz et al., 2017, Flyvbjerg and Budzier, 2011, Johnson, 2018). Based on feedback from 600 U.S. businesses and software executives, Geneca (2011) found that 75 percent of respondents admitted their projects were either always or usually "doomed" right from the start, 61 percent reported their projects take longer than anticipated, and 57 percent of projects are not considered a success. Additionally, 80 percent of respondents admitted they spend at least half their time on rework, which is the result of unclear objectives, confusion of roles and responsibilities, and lack of stakeholder involvement. McKinsey and the University of Oxford studied 5,400 software projects and found an average 17 percent shortfall in the benefits (or scope) achieved compared with the original plan. The study also identified average cost and schedule overruns of 66 percent and 33 percent, respectively. Perhaps most alarming, 17 percent of the documented projects performed so poorly that they threatened the very existence of the company (Bloch et al., 2012).

Early Software Project Stages as a Cause of Failure

A variety of failure causes have been identified in the literature. According to one study, these causes of failure can be arranged into three primary groups: people, process, and product-related risks (Kappelman et al., 2007). The people-related factors refer to managerial issues, including top management, team members, and project stakeholders. The process-related factors refer to fundamental project management processes, including requirements, change controls, scheduling, communications, and resources. The product-related factors refer to technical capabilities of software projects, including product size and complexity, technology risks, and software functionalities. Accordingly, the most common factors are related to people and process, which include poorly defined project requirements and SOW, unclear expression of milestones and deliverables, and ineffectively assigned project resources.

The leading causes of failure in software projects are commonly linked to early stages of project development and vendor procurement (Kronbichler et al., 2009; Morris and Pinto, 2007). Alami (2016) found that many software projects lack a well-developed set of itemized requirements and require subsequent SOW modifications after vendor proposals are received. Amjad et al. (2017) highlighted the importance of SOW definition in software projects to help proposing software vendors account all SOW and project goals when preparing their proposal responses. Arcidiacono (2017) indicated that software project failure occurs due mainly to poor project management (54 percent of failures captured in this study cited this cause), lack of defined project delivery activities (21 percent), project teams that lack necessary skills (8 percent), inefficient implementation approaches (3 percent), and project funding issues (14 percent). Agiloft (2019) recommended that software project owners should be sure to define the requirements, engineering specifications, software architecture, functionality, project timelines, evaluation criteria and weights, and effective training plans. This information varies in RFPs across different software project categories, from large ERP projects to a specialized classroom event scheduling software (Umble et al., 2003). Another study of large software projects found that they often cost more than planned due to their complexity, propensity for under-defined SOW expectations, lack of clarity in functional and technical requirements, absence of critical budgetary information, and unclear implementation timelines (Bloch et al., 2012).

Software RFP Content Analysis

Software project delivery is a series of activities often combined into a single process, which typically includes three main phases: (1) planning and SOW development, (2) procurement/acquisition, and (3) contract execution/implementation (Linman, 2010). First, the planning process includes the establishment of project goals and requirements via the preparation of SOWs that ultimately feed into the project RFP. Second, the procurement process includes acquisition of an appropriate vendor through evaluating their proposals and negotiating appropriate delivery terms before a formal contract is signed with the selected vendor. Third, the implementation includes the completed software installation and proper execution.

Regularly, the owner's project team spends substantial time and effort to establish SOW requirements and formulate the RFP evaluation criteria by which they will assess formal proposals from prospective software vendors (Jamieson et al., 2005). A typical software RFP consists of SOW information which often includes itemized requirements to document the owner's expectations of technical, functional, and security requirements (Morris and Pinto, 2007). In turn, the software vendor also spends a considerable time and effort to analyze the requirements and project objectives, estimating their cost and schedule offers, and preparing formal proposals or company quotations before the RFP deadline. The RFP is the main step that sets the foundational expectations around project cost, schedule, SOW, and functionality. Therefore, it is important that the owner's RFP thoroughly captures all the requirements, evaluation criteria, and other product-related information to prepare a complete and decent proposal.

The content analysis of RFP practices has been an area of research inquiry in other industries. For instance, the quality of design and construction procurements have been evaluated through a content analysis of 78 RFPs in public construction projects with a total contract value of over \$3 billion (Gransberg and Molenaar, 2004). The study also recommends construction contractors to craft their proposals in a more responsible manner to the owner's requirements to improve their evaluation scores. Gransberg and Barton (2007) investigated 110 RFPs in public construction projects to explore what federal owners are seeking from the awarded contractors in terms of cost and project personnel. Xia et al. (2013) identified 26 evaluation criteria in public design and construction projects through a content analysis of 94 RFPs advertised between 2000 and 2010. Lopez Del Puerto et al. (2013) studied 115 RFPs to assess the role of safety management in public design and construction projects. However, relatively younger field of software procurement is understudied in comparison with other industry sectors.

Point of Departure and Research Objective

Previous studies have documented the performance outcomes of software projects and identified early project stages of SOW development, project planning, and procurement steps as being areas of deficiency in software project delivery. However, little research has documented the current state of practice in the software procurement stage. To address this gap, this study aimed to investigate the current state of software project procurement by performing a content analysis of major RFP elements. The RFP process is a critical juncture in software projects because it is the point at which the software solution is selected based on an expectation of the SOW and functionality it promises to deliver along with the budget and schedule parameters within which the delivery will be accomplished. In this sense, the RFP process sets many of the foundational expectations around SOW, cost, and timeframe. Since previous literature has shown these areas to ultimately perform substantially lower than initial expectations, an investigation is warranted of the RFP elements that are responsible for forming these initial expectations. The current literature of software projects also lacks analysis of RFPs to investigate the extent to which the client provides pertinent information of the software product to the prospective software vendors.

This study aimed to identify the difference in RFP-related elements – evaluation criteria, procurement timing, and itemized requirements – across five common software project categories, including ERP, financial, asset management, CBA, and SBA. A number of software RFPs were collected and analyzed to support project owners and software vendors in understanding the essentials and needs in the procurement process may help enhance the performance of software projects.

Research Methodology

The research methodology of this study included four main steps. First, a comprehensive literature review of RFP analysis in software projects as well as other industries was performed to form the study's research questions. Second, a content analysis of recently collected software RFPs was conducted using descriptive and inferential statistics to investigate the common elements in software RFPs. Third, the results of current software RFP practices in five common software categories were discussed. Finally, conclusions and recommendations were drawn to help project owners and software vendors enhance the preparations and responses to software RFPs, respectively.

Research Questions

RQ1: What are the common evaluation criteria in procuring software projects? How are the common evaluation criteria weighted differently across five common software categories?

RQ2: What are the differences in procurement timelines (i.e., bidding, evaluation, negotiation, and implementation durations) between software categories?

RQ3: How do the itemized requirements in RFPs vary across the different software categories?

Data Collection

Initially, this study collected 309 software RFPs across North America, and then the data was filtered to 250 RFPs using three criteria. First, this study aimed at analyzing software RFPs released in the past decade (from 2010 to 2019) in the United States. Second, only public project owners, including government (municipal, county, and state), education, transportation, and healthcare, were analyzed. Third, this study removed several RFPs with scopes that strongly emphasized non-software elements.

This study investigated five common software categories, including ERP, financial, asset management, CBA, and SBA. First, the ERP category includes software and systems used to manage the core supply chain, manufacturing, and services of an organization and bring them together to enable a flow of data between the applications, typically through common databases either on-premise or in the cloud (Ahmad, 2013). Second, the financial category includes integrated financial management, accounting, and human resources/payroll software and system projects. This category involves several modules that may be (but are not necessarily) included as sub-components of a typical ERP system (Marnewick, 2005). However, the implementation of only the financial modules for the purposes of particular business units and end user needs is quite different than the implementation of a comprehensive ERP solution across the entire organization (Markus et al., 2000; Hoermann, 2011; Sudhaman and Thangavel, 2015). Third, the asset management category includes software which plan and manage the purchase, deployment, maintenance, execution, and removal of both physical and electronic assets owned by an organization (Jørgensen, 2006). This category concentrates on administering the maintenance and lifecycle of assets (Botta-Genoulaz and Millet, 2006). Fourth, the CBA category includes common business software applications used by multiple-owner types, such as tracking and planning, permitting and inspection, work-orders, scheduling, information technology management, inventory management,

and performance management and reporting software. All of these solutions can be used in different organizational contexts. Fifth, the SBA category includes specialized business software applications used by single-owner types, such as electronic health record (EHR) systems, integrated care management system, nutrition management, patient identification, and disease registry for healthcare organizations; flight information display system (FIDS), IP video surveillance and video management system, and ticketing system for airports; intelligent transportation system (ITS), Computer-Aided Dispatch/Automatic Vehicle Location (CAD/AVL) system, and fixed route scheduling software for public transportation agencies; and classroom and event scheduling, parking management, student lead tracking, online catalog and curriculum management, and student attendance management for institutions.

Four commonly used elements of software RFPs were analyzed in this study via a content analysis: procurement timelines, contract terms, level of SOW development, and evaluation criteria. Procurement timelines included the planned durations of bidding, evaluation, negotiation, and implementation processes. The contract terms including the initial term length, the total duration of all renewals, and the total number of renewal options. The level of SOW development was defined via the total number of itemized requirements published in the RFP. A review of eight typical evaluation criteria in software projects was conducted as defined below:

- **Cost Proposal**: the financial proposal inclusive of the system, installation, conversion, training, licensing, and annual software maintenance (Araújo et al., 2017).
- **Response to RFP Requirements**: assesses the ability of the vendor to deliver required elements in RFP. In other words, responsiveness of the proposal offering to the purpose and scope of service. This criterion includes quality, clarity and responsiveness of proposal in conformance with instructions condition and format contained.
- **Implementation Approach**: shows the proposed implementation methodology of the software vendor (i.e., project management approach for deploying the software system). It also details the ability to provide training of the proposed system for multiple users, workflows, and scenarios as well as provide services and complete required work within the mutually agreed upon schedule.
- **Company Qualifications**: shows the strength, stability, and technical experiences in performing closely similar work.
- **Project Team Qualifications**: indicates the expertise, experience, and qualifications of the specific project team members who will provide services as related to the SOW as requested in the RFP. This criterion mainly focuses on the qualifications of personnel assigned to the project for the software implementation phase.
- **System Capability**: shows the ability of the vendor to meet to the software requirements listed in the RFP, including functional, business, technical, and security requirements. It defines the proposed integration with other modules/systems in RFP's SOW as well as the compliance with functional/technical specifications and upgradeability. This criterion also considers "Post Go-Live" customer services.

- **Software Demonstration**: typically includes a presentation of the capability of product as well as the flexibility of the products and its ease of use. This criterion also explains products and processes in an understandable manner and demonstrates a sound and tested approach to implementing the solution required by the owner.
- **Other Criteria**: all other criteria beyond those defined above, most commonly inclusive of the vendor's financial capability and business structure, vendor's local reputation, vendor's past relationship with the owner, sustainability efforts, small business practices, and privacy and security considerations.

Data Analysis

This study investigated the difference between software RFPs elements, including evaluation criteria and weights, procurement durations, contract terms, and number of itemized requirements, across the five common software project types using descriptive and inferential statistics. Specifically, a set of mean difference testing methods and Spearman's correlation coefficients were utilized. The normally distributed datasets were tested using parametric tests (t-test and Welch's test) while the non-normally distributed datasets were tested using nonparametric tests (Kruskal-Wallis H-test and Mann-Whitney U-test).

Results and Discussion

This section discusses the results from the descriptive and inferential analyses of the five common software categories. The difference between five software categories was tested in terms of evaluation criteria, procurement durations, and level of SOW development. The collected data was not normally distributed (the significance values of Shapiro-Wilk test were less than 0.05); therefore, the Kruskal-Wallis H-test and Mann-Whitney U-test were used. The results are provided with discussion in the following sub-sections.

Differences in Evaluation Criteria and Weights across Software Categories

Among the eight common evaluation criteria identified in the content analysis, the three criteria of the cost proposal, implementation approach, and company qualifications were included with the greatest frequency; in fact, these criteria were listed in more than 92 percent of the collected RFPs. The remaining criteria of project team qualifications, response to RFP requirements, and software demonstration were included in roughly one-third of RFPs. The results showed that software RFPs did not frequently provide weights of the evaluation criteria by which they would judge software vendor proposals. Overall, less than half of the collected RFPs (43 percent of 250) provided information about evaluation criteria weights.

Depending on the software category being procured, public agencies tended to place a different emphasis on the evaluation weights that are used to make the selection. Table 1 shows that cost proposals, company qualifications, implementation approach, and software capability were the most frequently used evaluation criteria with the average weights being 21 percent, 21 percent, 27 percent, and 31 percent, respectively. On the other hand, response to RFP requirements, project team qualification, and software demonstration were less frequently defined and had average weights of 12 percent, 17 percent, and 17 percent, respectively. It is noted that the software demonstration might be considered as a separate step of the proposal's evaluation, where the short-listed vendors are invited to demonstrate their product separately from the other criteria listed in the RFP. In such cases the demonstration may not be explicitly defined in the RFP document.

Table 1 Common Evaluation Criteria and Weights Provided in Software RFPs

Evaluation Criteria	Frequency (n)	Average (%)	Median (%)	Min (%)	Max (%)
Cost Proposal	102	21	20	3	60
Response to RFP Requirements	32	12	10	5	40
Implementation Approach	103	27	25	10	61
Company Qualifications	98	21	20	4	60
Project Team Qualifications	30	17	16	4	45
System Capability	77	31	30	10	60
Software Demonstration	36	17	20	6	31
Other Criteria	23	8	10	1	20

Across the five common software categories, statistically significant differences in evaluation criteria weights were found at the 95 percent confidence level in three evaluation criteria: cost proposal, software demonstration, and response to RFP requirements. The differences in evaluation criteria weights were confirmed via post-hoc testing and the statistically significant pairwise comparisons are reported as follows. Table 2 shows that, in ERP projects, clients tended to assign higher weight (by 9 percent) to software demonstrations than the other software categories. The SBA category tended to have higher weights (24 percent) than the CBA category (16 percent) in the area of cost proposals. The financial category generally had higher weights (9 percent) than the ERP category (2 percent) in the response to RFP requirements.

Table 2 Differences in Evaluation Criteria Weights across Software Categories (n=250)

Evaluation Criteria	Average Weight (%)					Significance (p)	
	ERP	Financial	Asset	CBA	SBA	Normality	K-W H Test
Cost Proposal	20	23	20	16	24	0.001	0.142
Response to RFP Requirements	2	9	3	4	4	0.000	0.155
Implementation Approach	22	23	28	29	26	0.024	0.316
Company Qualifications	15	21	20	21	19	0.000	0.410
Project Team Qualifications	3	6	5	6	3	0.000	0.984
System Capability	27	17	21	22	19	0.000	0.548
Software Demonstration	9	0	2	0	2	0.000	0.000*
Other Criteria	2	1	1	3	2	0.000	0.707

(*): Statistically significant at the 95 percent confidence level

Differences in Procurement Duration across Software Categories

Table 3 shows the various procurement durations that were most commonly specified in software RFPs. As can be observed, software project owners typically allocated a month for each stage of the procurement process (bid, evaluation, and negotiation durations). The bid duration was the most commonly identified of the procurement timelines and was defined in 98 percent of RFPs. The evaluation and negotiation durations were published with somewhat less regularity and were found in 68 and 39 percent of RFPs, respectively.

Table 3 Project Durations Published in the RFP

Project Durations	Frequency (n)	Average (days)	Median (days)	Min (days)	Max (days)
Bid duration	245	34	33	9	80
Evaluation duration	171	49	37	1	434
Negotiation duration	97	34	30	3	140
Implementation duration	76	265	180	30	730

The implementation durations were published with must lower regularity. Among RFPs that published the owner's expectation of the implementation duration, the average duration was approximately nine months. Based on performance studies from the Standish Group (2015), Geneca (2011), and others, it is likely that major software implementations will experience some level of schedule delay. Therefore, based on a nine-month implementation schedule – which only represents the baseline expectation from the client (developed without vendor input, expertise, or guidance) – clients may be advised to add an extra time to their implementation expectations as a conservative internal expectation. This would enable more accurate forecasting of client level of effort and billing cycles.

There were statistically significant results at the 95 percent confidence level found in the bidding and evaluation durations across the five software categories as shown in Table 4. ERP projects had an approximately 1-week longer bidding duration (37 days in total) than asset management projects (32 days) and SBA projects (32 days). In addition, ERP projects also had a longer evaluation duration (62 days) than other categories, including asset management (48 days), financial (46 days), CBA (38 days), and SBA (52 days). In other words, it required roughly 2 months to evaluate and select an ERP system while the other software categories needed roughly 1 to 1.5 months of time. Although no statistically significant difference in negotiation and implementation durations was found, this is likely due to small sample size of RFPs that published these timelines.

Table 4 Differences in Procurement Durations across Software Categories

Procurement Durations	Average (days)					Significance (p)	
	ERP	Financial	Asset	CBA	SBA	Normality	K-W H Test
Bidding	37	34	32	33	32	0.000	0.034*
Evaluation	62	46	48	38	52	0.000	0.001*
Negotiation	41	29	32	33	31	0.000	0.636
Implementation	238	236	199	207	186	0.000	0.582

(*): Statistically significant at the 95 percent confidence level

Table 5 shows that software project owners typically targeted 3 to 5 years as the initial contract term of the agreement they intended to sign with the selected vendor. However, this information was provided in less than 15 percent of RFPs. The potential for contract renewals were identified with less frequency and were identified in less than 10 percent of RFPs. Due to the small frequency of occurrence, there were no statistically significant differences in the contract terms specified across the five common software categories. Based on observation of descriptive statistics, the RFPs of SBA and CBA provided the contract terms somewhat more frequently than ERP, financial, and asset management projects.

Table 5 Differences in Contract Terms across Software Categories

Software Category	Statistics	ERP	Financial	Asset	CBA	SBA
Initial Terms	Frequency (n)	25	23	20	32	36
	Average	5	4	3	4	3
	Median	4	5	3	3	3
Total Renewals	Frequency (n)	9	7	9	13	26
	Average	5	4	4	4	3
	Median	2	3	4	3	3
Total Renewal Options	Frequency (n)	9	7	9	15	26
	Average	2	2	3	3	2
	Median	1	2	3	2	2

Differences in Itemized Requirements across Software Categories

Table 6 shows that software project owners often defined a greater quantity of itemized requirements in financial and ERP projects than CBA and asset management projects. However, there was no statistically significant difference in SOW development across the five software categories. Additionally, no correlation between the number of itemized requirements and any of the procurement timelines was found. The level of SOW development had a statistically significant correlation with the system capability; however, the correlation coefficient (0.238) was so weak that it represented no association of practical significance to the industry. There was also no statistically significant difference between the level of SOW development and the year of RFP release which indicates that more recent RFPs do not necessarily produce more itemized requirements. The level of SOW development included in the RFP was not associated with changes in procurement timelines nor evaluation criteria weights.

Table 6 Number of Itemized Requirements across Software Categories

Software Category	ERP	Financial	Asset	CBA	SBA
Frequency (n)	41	46	41	47	47
Average	70	79	60	51	63
Median	20	28	35	25	30
Min	3	4	8	4	6
Max	450	1020	270	388	255

Transparency in Software RFPs

The analysis of software RFP transparency was conducted by categorizing the collected RFPs into three main groups of transparency levels: advanced, modest, and novice, in terms of six common software RFP elements, including SOW development, procurement timelines, implementation schedule, budget information, evaluation process, and contractual terms. The modest group (n=39) included 20 or more itemized requirements, information of project bidding and evaluation durations, and detailed evaluation criteria and weights. The modest group (n=153) included information defining SOW development, project bidding duration, and specific evaluation criteria. The novice group (n=58) included only the bidding timelines, such as the dates of RFP release, questions/pre-proposal conference dates, and the time at which proposals were due. The difference between six software RFP elements across the three pre-defined groups of RFP transparency was analyzed with a set of t-Test, Welch's Test, and Mann-Whitney U-test.

A summary of software RFP transparency is provided below:

- Software RFPs rarely provided project budget information (2 percent of 250). None of the RFPs separated budget information into implementation costs and annual fees (license, maintenance, and customer support) and instead simply listed an overall total budget amount.
- Proposal templates, such as cost proposal and functionality requirements, were not frequently provided for vendors to use (< 50 percent of 250).
- The majority of software RFPs provided minimal information regarding the procurement timeline (i.e., generally limited to the bidding duration), specific evaluation criteria (but without the weights), and itemized requirements.
- Approximately 16 percent of software RFPs provided specific evaluation criteria and weights, detailed bidding and evaluation timelines, and more detailed SOW development (i.e., more than 20 itemized requirements listed).
- Software RFPs do not frequently provide implementation timelines (30 percent of 250) nor contract renewal options (26 percent of 250).

Accordingly, this study recommends a “checklist” of sorts for public agencies to evaluate the level of transparency of their software RFPs. Three suggested groupings of RFP transparency levels (advanced, modest, and novice) are suggested in Table 7. The transparency elements consist of the level of SOW definition (provision of a separate SOW section of the RFP, the number of itemized requirements defined by the owner, and provision of a dedicated description of the owner's current conditions), the level of clarity in the procurement timeline (assessed by whether the bid, evaluation, and negotiation durations were provided), the definition of the owner's expected implementation timeline, the level of detail in describing the owner's evaluation process, and whether the owner provided standardized proposal form templates for all vendors to use.

Table 7 Recommendation for Classifying RFP Transparency

RFP Element	Advanced	Modest	Novice
Itemized Requirements	> 60	20 - 59	< 20
Statement of Work (SOW)	Yes	Yes	Yes
Current Conditions	Yes	No	No
Procurement Timeline	Bidding, Evaluation, and Negotiation	Bidding and Evaluation	Bidding
Implementation Schedule	Yes	Yes	No
Budget Information	Yes	No	No
Evaluation Process	Criteria, Weights, and Templates for Cost and Itemized Requirements	Criteria and Weights	Criteria
Contract Term	Yes	No	No
Proposal Templates	Yes	Yes	No

The elements and the three groupings in Table 7 may serve as a guideline to assist owners in developing RFPs which provide software vendors with adequate and proper information for their proposal preparation. Greater transparency these RFP elements is important for several reasons. For example, owners should consider clarifying the implementation duration and specific the “Go-Live” date to enhance the transparency of their RFPs. An implication is that software vendors are less well-equipped to provide detailed or realistic implementation plans because they may be blind to the client’s expectations and constraints (Agiloft, 2019). Further, the provision of standardized proposal templates is important because they support an “apples-to-apples” comparison in the client’s evaluation process and give each software vendor an equivalent ability to provide their proposal information in a consistent format.

Conclusion

Software procurement has continued to garner attention due to the increasing volume of software purchases coupled with the challenging nature of software implementations. According to the literature, the accuracy and transparency in software RFPs may help improve the time, cost, and quality performance of software implementations. To better understand the current state of practice in the area of software procurement, this study conducted a content analysis with a sample size of 250 software RFPs released in the past decade (from 2010 to 2019) in the United States. The RFPs were collected across five common software categories, including ERP, financial, asset management, CBA, and SBA. A set of parametric (t-test and Welch’s test) and nonparametric tests (Kruskal-Wallis H-test and Mann-Whitney U-test) were used to investigate the difference in evaluation criteria, procurement timelines, and itemized requirements provided in software RFPs. The results found some differences between the software categories. Differences in evaluation criteria weights were found for cost proposals (SBA projects had a 6 percent higher weight than CBA), demonstrations (ERP projects has a 9 percent higher weight than other categories), and

response to RFP requirements (the financial category generally had higher 7 percent higher weight ERP). In terms of procurement timelines, ERP projects had a longer bidding duration than asset management and SBA projects and a longer evaluation duration than all other categories. Software project owners often defined more itemized requirements in financial and ERP projects than CBA and asset management projects.

Overall, results showed a relative lack of transparency in the current state of practice among software RFPs. In reviewing the current state of practice, software RFPs rarely provide their project budget, define their implementation timeline expectations, provide information about the owner's current conditions, and contract terms. Further, most software RFPs did not publish the weights of their evaluation criteria. This study recommends that owners consider releasing the above information in their RFPs to provide a more solid benchmark for vendors to bid to and understand the owner's project priorities. Increasing the accuracy and project-specific nature of vendor proposals is advantageous given the importance that the literature places on the RFP stage in setting the major project parameters of cost, schedule, and scope requirements.

Research Contribution

This study contributes to the body of knowledge of software project delivery by investigating the current state of practice in developing software RFPs. Major differences in evaluation criteria and weights, procurement timelines, and itemized requirements provided in software RFPs are described among five common software categories (ERP, financial, asset management, CBA, and SBA). This study is the first study that concentrates on examining contemporary software procurement procedures with the analysis of a large number of software RFPs. The recommendations of this study in addressing the lack of owner-provided information provided in RFPs may improve the transparency of software project procurement.

To practitioners, this study provides practical applications to help project owners gain awareness of the current state of practice and increase the level of transparency in their software RFPs. In turn, this may support software vendors in preparing a more accurate proposal and bid information. Three recommendations are noted software project procurement. First, information regarding budget and implementation schedule should be clearly specified in the software RFPs, which can help software vendors be aware of potential restrictions in and more effectively plan for the availability of their project resources. Second, specific evaluation weights can be helpful to inform software vendors about the aspects that the owner is most interested in assessing as part of their evaluation decision. Finally, software demonstration processes and associated weights should be more clearly clarified in the RFPs so that software vendors can sufficiently demonstrate the capability and functionality of the product in an effective manner that is consistent and fair across bidders.

Limitations and Recommendations for Future Work

The software demonstration category was often not defined as a separate evaluation criterion with its own weight. Presumably, most software procurements would be reasonably expected to consider software demonstration. Therefore, there is a limitation in the manner by which software RFPs describe their demonstration expectations which limits the ability of a content analysis – as conducted in this study – to report the frequency and specificity of software demonstration procedures that are being used across the country.


Future work is recommended to analyze the performance of software project implementations. This would enable researchers to link the factors in software RFPs that have the greatest relationship with successful project performance outcomes. Based on the importance of the initial stages of software project delivery, some amount of causal linkages would be expected between up-front procurement practices and eventual performance of the software implementation effort. The collected procurement timelines in this study were only the planned durations, which could be expanded to include other sub-tasks in the procurement process.

References

- Abouzahra, M. (2011). Causes of failure in Healthcare IT projects. In *3rd International Conference on Advanced Management Science*. Singapore: IACSIT Press.
- Agarwal, N., and Rathod, U. (2006). Defining 'success' for software projects: An exploratory revelation. *International Journal of Project Management*, 24(4), 358-370.
- Agiloft (2019). *How to Ensure the Success of IT Projects*. White Paper.
<https://www.agiloft.com/how-to-ensure-success-it-projects.pdf>
- Alami, A. (2016). Why Do Information Technology Projects Fail? *Procedia Computer Science*, 100, 62-71.
- Amjad, S., Ahmad, N., Saba, T., et al. (2018). Calculating Completeness of Agile Scope in Scaled Agile Development. *IEEE Access*, 6, 5822-5847.
- Araújo, M., Alencar, L., and de Miranda Mota, C. (2017). Project procurement management: A structured literature review. *Int. J. of Project Management*, 35(3), 353-377.
- Arcidiacono, G. (2017). Comparative research about high failure rate of IT projects and opportunities to improve. *PM World Journal*. 6(2).
- Belfo, F., and Trigo, A. (2013). Accounting Information Systems: Tradition and Future Directions. *Procedia Technology*, 9, 536-546.
- Bloch, M., Blumberg, S., and Laartz, J. (2012). Delivering large-scale IT projects on time, on budget, and on value. Retrieved 22 September 2019, from
<https://www.mckinsey.com/business-functions/mckinsey-digital/our-insights/delivering-large-scale-it-projects-on-time-on-budget-and-on-value>
- Botta-Genoulaz, V., and Millet, P. (2006). An investigation into the use of ERP systems in the service sector. *Int. J. of Production Economics*, 99(1-2), 202-221.
- Cragg, T., and Chraïbi, S. (2020). Reviewing procurement: a change management model based on action research. *International Journal of Procurement Management*, 1(1), 1.

- Fayaz, A., Kamal, Y., Amin, S., et al. (2017). Critical success factors in information technology projects. *Management Science Letters*, 73-80.
- Flyvbjerg, B., and Budzier, A. (2011). Why Your IT Project May Be Riskier than You Think. *SSRN Electronic Journal*. doi: 10.2139/ssrn.2229735
- Geneca (2011). *Doomed from the Start? Why a Majority of Business and IT Teams Anticipate Their Software Development Projects Will Fail*. Winter 2010/2011 Industry Survey.
- Gransberg, D., and Molenaar, K. (2004). Analysis of Owner's Design and Construction Quality Management Approaches in Design/Build Projects. *Journal of Management in Engineering*, 20(4), 162-169.
- Gransberg, D., and Barton, R. (2007). Analysis of Federal Design-Build Request for Proposal Evaluation Criteria. *Journal of Management in Engineering*, 23(2), 105-111.
- Hassan, I., Ahmad, N., and Zuhaira, B. (2018). Calculating completeness of software project scope definition. *Information And Software Technology*, 94, 208-233.
- Hoermann, S., Kienegger, H., Langermeier, M., et al. (2011). "Comparing Risk and Success Factors in ERP Projects: A Literature Review," in *Americas Conference on Information Systems*, Detroit, Michigan, USA, p. 241
- Jamieson, D., Vinsen, K., and Callender, G. (2005). Agile Procurement: New Acquisition Approach to Agile Software Development. In *Proceedings of the 2005 31st EUROMICRO Conference on Software Engineering and Advanced Applications*. IEEE.
- Johansson, B. and Lahtinen, M. (2012). Requirement Specification in Government IT Procurement. *Procedia Technology*, 5, pp.369-377.
- Johnson, J. (2018). *CHAOS Report: Decision Latency Theory: It Is All About the Interval*. Standish Group.
- Jørgensen, M. (2006). The effects of the format of software project bidding processes. *Int. J. of Project Management*, 24(6), 522-528.
- Jørgensen, M., Mohagheghi, P., and Grimstad, S. (2017). Direct and indirect connections between type of contract and software project outcome. *Int. J. of Project Management*, 35(8), 1573-1586.
- Kappelman, L., McKeeman, R., and Zhang, L. (2007). Early Warning Signs of it Project Failure: The Dominant Dozen. *EDPACS*, 35(1), 1-10. doi: 10.1080/07366980701238939
- Kronbichler, S., Ostermann, H., and Staudinger, R. (2009). A Review of Critical Success Factors for ERP-Projects. *The Open Information Systems Journal*, 3(1), 14-25.

- Linman, D. (2010). *What is IT Procurement Process and How to Manage It?* from <https://mymanagementguide.com/what-is-it-procurement-managing-the-it-procurement-process/#:~:text=IT%20procurement%20is%20a%20series,business%20processes%20within%20every%20organization>.
- Lopez Del Puerto, C., Strong, K., and Miller, M. (2013). Analysis of Owners Safety Management Approaches in Design-Build Projects. *Int. J. of Construction Education and Research*, 9(4), 307-316.
- Markus, M., Tanis, C., and van Fenema, P. (2000). Enterprise resource planning: multisite ERP implementations. *Communications of the ACM*, 43(4), 42-46.
- Marnewick, C., and Labuschagne, L. (2005). A conceptual model for enterprise resource planning (ERP). *Information Management and Computer Security*, 13(2), 144-155.
- Moe, C. (2014). Research on Public Procurement of Information Systems: The Need for a Process Approach. *Communications of the Association for Information Systems*, 34. doi: 10.17705/1cais.03478
- Moe, C., Newman, M., and Sein, M. (2017). The public procurement of information systems: dialectics in requirements specification. *European Journal of Information Systems*, 26(2), 143-163.
- Morris, P., and Pinto, J. (2007). *The Wiley Guide to Project Technology, Supply Chain and Procurement Management* (p. 256). Hoboken, New Jersey: Wiley.
- Roslina Ibrahim, D. (2013). An Investigation of Critical Failure Factors in Information Technology Project. *IOSR Journal of Business and Management*, 10(3), 87-92.
- Saito, Y., Monden, A., and Matsumoto, K. (2012). Evaluation of Non-Functional Requirements in a Request for Proposal (RFP). In *Joint Conference of the 22nd International Workshop on Software Measurement and the Seventh International Conference on Software Process and Product Measurement*.
- Salah, A., Ramadan, N., and Ahmed, H. (2017). Towards a Hybrid Approach for Software Project Management using Ontology Alignment. *Int. J. of Computer Applications*, 168(6), 12-19.
- Standish, G. (2015). CHAOS Report 2015. The Standish Group International, Inc. https://www.standishgroup.com/sample_research_files/CHAOSReport2015-Final.pdf
- Sudhaman, P., and Thangavel, C. (2015). Efficiency analysis of ERP projects—software quality perspective. *Int. J. of Project Management*, 33(4), 961-970.
- Thomas, G., and Fernández, W. (2008). Success in IT projects: A matter of definition? *Int. J. of Project Management*, 26(7), 733-742.

- 
- Umble, E., Haft, R., and Umble, M. (2003). Enterprise resource planning: Implementation procedures and critical success factors. *European Journal of Operational Research*, 146(2), 241-257.
- Wagner, W., and Lederer, Y. (2004), “An Analysis of the Imagine PA Public Sector ERP Project”, in *proceedings of the 37th Hawaii International Conference on System Sciences*, Big Island, Hawaii, 2004.
- Xia, B., Chan, A., Zuo, J., et al. (2013). Analysis of Selection Criteria for Design-Builders through the Analysis of Requests for Proposal. *Journal of Management in Engineering*, 29(1), 19-24.
- Zaman, U., Jabbar, Z., Nawaz, S., and Abbas, M. (2019). Understanding the soft side of software projects: An empirical study on the interactive effects of social skills and political skills on complexity – performance relationship. *Int. J. of Project Management*, 37(3), 444-460.