Software Quality Journal (2019) 27:429–493
https://doi.org/10.1007/s11219-018-9419-5

Correlation of critical success factors with success of software projects: an empirical investigation

Vahid Garousi[1] · Ayça Tarhan[2] · Dietmar Pfahl[3,4] · Ahmet Coşkunçay[5] · Onur Demirörs[6,7]

Published online: 27 October 2018

Abstract
Software engineering researchers have, over the years, proposed different critical success factors (CSFs) which are believed to be critically correlated with the success of software projects. To conduct an empirical investigation into the correlation of CSFs with success of software projects, we adapt and extend in this work an existing contingency fit model of CSFs. To archive the above objective, we designed an online survey and gathered CSF-related data for 101 software projects in the Turkish software industry. Among our findings is that the top three CSFs having the most significant associations with project success were: (1) team experience with the software development methodologies, (2) team's expertise with the task, and (3) project monitoring and controlling. A comprehensive correlation analysis between the CSFs and project success indicates positive associations between the majority of the factors and variables, however, in most of the cases at non-significant levels. By adding to the body of evidence in this field, the results of the study will be useful for a wide audience. Software managers can use the results to prioritize the improvement opportunities in their organizations w.r.t. the discussed CSFs. Software engineers might use the results to improve their skills in different dimensions, and researchers might use the results to prioritize and conduct follow-up in-depth studies on those factors.

Keywords Software engineering · Project management · Software projects · Success and failure · Critical success factors · Empirical studies

1 Introduction

Software industry has grown with incomparable pace during the past decades. The challenges of today's software systems are different, in terms of increased complexity, size and diversity of demand, than the challenges of the systems 50 years ago (Boehm 2006). The challenges and advances in the field of software engineering have inspired process improvement models such

✉ Vahid Garousi
vahid.garousi@wur.nl

Extended author information available on the last page of the article

as CMMI and SPICE, as well as software development paradigm shifts such as lean and agile models of software development. Although there are improvements, software development projects continue to have larger failure rates than traditional engineering projects (Savolainen et al. 2012). As a result, understanding the key factors underlying the success and failure of software projects is a critical subject of contemporary software engineering research.

The patterns of success and failure, often referred to as the Critical Success Factors (CSFs), are some of the best indicators of the lessons learned, adapted and used by software industry today. Systematic analyses of success and failure in software projects have been performed in many studies, e.g., (Ahimbisibwe et al. 2015; Lehtinen et al. 2014; Whitney and Daniels 2013; Hashim et al. 2013; Stankovic et al. 2013; Subiyakto and bin Ahlan 2013; Sudhakar 2012; Nasir and Sahibuddin 2011; McLeod and MacDonell 2011; Cerpa et al. 2010; Chow and Cao 2008; Agarwal and Rathod 2006; Weber et al. 2003; Pereira et al. 2008; Drew Procaccino et al. 2002). The notion of CSFs was first introduced by Rockart in 1979 (Stankovic et al. 2013) (Rockart 1979) as a method to help managers determine what information is the most relevant in order to successfully attain their goals in an increasingly complex world.

In this study, we aim to characterize the CSFs and analyze their correlations with the success of software projects by means of a questionnaire-based survey. Our study aims at determining the ranking of CSFs, which is a measure of the importance of CSFs for project success. The ranking is derived by quantitative ordering of the CSFs based on their correlations with software project success, for various aspects of project success perceptions. To achieve this goal, we have selected a CSF model, compiled and developed previously, based on a large-scale literature review of 148 primary studies (Ahimbisibwe et al. 2015) as the baseline. Using the Goal, Question, Metric (GQM) approach, we then systematically developed an online questionnaire. We distributed the survey to a network of our industry contacts and various channels including social media and technology development zones in Turkey. The survey gathered responses from 101 software projects during the period of August–September of 2015.

The results of the survey are the basis for a detailed discussion of the relations between CSF factors and perceptions of project success. Interesting findings from this study include identification of the relative significance of team factors such as team's experience with the task and methodologies and organizational factors such as project monitoring/controlling, and insignificance of customer factors such as customer experience and almost all of the project factors such as level of specification changes. The contributions of this paper are three-fold:

- A review of the existing CSF classification schemes and applying necessary improvements to one of the most compressive models (Ahimbisibwe et al. 2015),
- Contributing further empirical evidence in the area of project outcomes and CSFs by assessing the correlations among the CSFs and various project success metrics,
- Serving as the first work to empirically assess changes in importance rankings of CSFs for different company sizes (larger versus smaller companies), different project sizes (in terms of number of team members) and also different software development methodologies (Agile versus traditional).

The remainder of this paper is structured as follows. A review of background and the related work is presented in Section 2 including the contingency fit model of CSFs that we used. Section 3 discusses the research goal and research methodology. Section 4 presents the demographics and independent statistics of the dataset. Section 5 presents the results and findings of the survey. We then provide the discussions and implications of the findings in Section 6. Finally, in Section 7, we draw conclusions, and discuss our ongoing and future works.

2 Background and related work

2.1 Related work

CSFs are defined as the key areas where an organization must perform well on a consistent basis to achieve its mission (Gates 2010). It is argued by (Caralli 2004) that managers implicitly know and consider these key areas when they set goals and as they direct operational activities and tasks that are important to achieving goals. When these key areas of performance are made explicit, they provide a common point of reference for the entire organization. The project management literature in general, and the software project management literature in particular, has seen a significant number of studies about CSFs.

There are a few studies in the literature that have attempted to reveal CSFs of software projects either by review of the related literature or by empirical investigation (e.g., using opinion surveys). There are many "primary" studies on success factors and CSFs in software projects and it is thus impossible to review and discuss each of those studies in this paper. Thus, we selected a subset of these studies as we list in Appendix Table 9. We provide a brief summary of those papers in the following.

The studies (Lehtinen et al. 2014; Whitney and Daniels 2013) investigated the reasons of project failures. The study (Weber et al. 2003) focused on predicting project outcomes. Six of the related studies are based on literature review of CSFs. Among these, studies (Ahimbisibwe et al. 2015; Sudhakar 2012; McLeod and MacDonell 2011) offered conceptual models of CSFs for software projects after reviewing and synthesizing the relevant literature. These studies ranked CSFs based on their occurrence frequencies in the literature. Some studies, such as (Ahimbisibwe et al. 2015), identified and ranked the CSFs for plan-driven and agile projects separately, while studies such as (Sudhakar 2012) did not consider such a distinction. Using a literature review of 177 empirical studies, the study in (McLeod and MacDonell 2011) proposed 18 CSFs in four different dimensions: institutional context, people and action, project content, development processes.

Some of the studies aimed at empirical validation of the CSFs. Such studies are investigations of CSFs either by surveys or case studies, e.g., (Lehtinen et al. 2014). One of the studies in the set (Chow and Cao 2008) conducted a literature review to first elicit a list of CSFs and then conducted a questionnaire-based survey for assessing the importance of each CSF in agile projects.

Among the studies that used opinion surveys as the empirical investigation method, the studies (Stankovic et al. 2013; Chow and Cao 2008) focused on the CSFs in agile software projects only. The study (Cerpa et al. 2010) proposed a regression model to predict project success based on data from multiple companies, and reported a prediction accuracy in the range of [67%–85%]. Another study (Agarwal and Rathod 2006) searched for a view of successful projects from the perceptions of different project roles in terms of generic factors such as time, cost, and scope. The study (Pereira et al. 2008) investigated regional/cultural differences in definitions of project success between Chilean and American practitioners, while the study (Drew Procaccino et al. 2002) identified early risk factors and their effect on software project success. The study (Weber et al. 2003) examined the accuracy of several case-based reasoning techniques in comparison to logistic regression analysis in predicting project success.

The study in (Lyytinen and Hirschheim 1987) presented a survey and classification of the empirical literature about failures in Information Systems (IS). It proposed a classification of IS failure based on the empirical literature, into four groups: process failure, interaction failure, and expectation failure. It also classified the sources of failure into the following categories: IS failure, IS environment failure, IS development failure, and IS development environment failure.

The study in (Poulymenakou and Holmes 1996) presented a contingency framework for the investigation of information systems failure and CSFs. The proposed CSFs were under these groups: users, social interaction, project management, use of a standard method, management of change, and organizational properties.

Another study (Scott and Vessey 2002) discussed how to manage risks in enterprise systems implementations. It proposed a classification of risk (and failure) factors, which were based on several case-study projects in which the authors were involved.

The study in (Tom and Brian 2001) assessed the relationship between user participation and the management of change surrounding the development of information systems. It also proposed a classification of CSFs which were based on a case study in a large organization.

Entitled "*Critical success factors in software projects*", the study in (Reel 1999) also proposed a classification of CSFs which were based on the author's experience. It discussed five essential factors to managing a successful software project: (1) Start on the right foot (Set realistic objectives and expectations; Build the right team; Give the team what they think they need); (2) Maintain momentum (Management, quality); (3) Track progress; (4) Make smart decisions; (5) Institutionalize post-mortem analyses. The paper included many interesting observations, e.g., "*At least seven of 10 signs of IS project failures are determined before a design is developed or a line of code is written*" (Reel 1999).

Last but not the least, the work in (Berntsson-Svensson and Aurum 2006) assessed whether there could be different perceptions about what effect various factors have on software project success among different industries. Based on an empirical investigation, the study concluded that there are indeed differences among the factors which are important for project/product success across different industries.

2.2 Reviewing the existing CSF classification schemes

To conduct our empirical investigation, we had to choose a suitable CSF classification scheme from the literature (as discussed above). In the set of studies that we had reviewed in the literature (as listed in Appendix Table 9 and summarized above), we specifically chose the papers which had presented CSF classification schemes. We found that the papers presenting CSF classification schemes are either regular surveys, SLR papers, or studies based on opinion-surveys (questionnaires). The advantage of looking at survey or SLR papers in this area was that each such paper had already reviewed a large number of "primary" studies on CSFs and thus the resulting CSF classification scheme in the survey or SLR paper was a synthesis of the literature which would prevent us from having to read each of those many primary studies. Thus, we decided to review the papers which had presented CSF classifications and chose the most suitable one.

At the end of our literature searches for CSF classification schemes, we populated our pool as shown in Table 1. We also show in Table 1 the coverage of CSFs in each existing CSF classification scheme. We should note that, while we were as rigorous as possible in search for the CSF classification schemes, we did not conduct a systematic literature review (SLR).

After a careful analysis and comparison, we decided to adopt the CSF classification by Ahimbisibwe et al. (Ahimbisibwe et al. 2015) due to the following reasons:

- Many of the studies have taken a coarse-grained approach by aggregating a subset of factors into one single CSF, while (Ahimbisibwe et al. 2015) has considered the factors in finer granularity. For example, "Organizational properties "has been considered as one single factor in many studies (e.g., (McLeod and MacDonell 2011)), while (Ahimbisibwe et al. 2015) has considered that factor in finer granularity, e.g., top management support, organizational culture, leadership, vision and mission. As another example, the "User" aspect has been considered as a single factor by many studies (e.g., (McLeod and MacDonell 2011)), while (Ahimbisibwe et al. 2015) has considered that aspect in finer granularity, e.g., user support, experience, training and education. We believe it is important to assess those aspects separately rather than seeing them as one single factor.
- The study (Ahimbisibwe et al. 2015) has the finest granularity and number of CSFs (38 factors) compared to other studies.
- We found that some of the groupings (clustering) done in many of the works are weak and/or impartial, and thus need to be improved, e.g., "project management" is placed in (McLeod and MacDonell 2011; Lyytinen and Hirschheim 1987; Poulymenakou and Holmes 1996; Scott and Vessey 2002; Tom and Brian 2001) under the "Development processes" group while it shall be placed under the "Project aspects" group. As another example, (McLeod and MacDonell 2011) placed "User participation" and "User training" under the "Development processes" group while they should be under the "Users" factor. Many classifications are also impartial (not providing full coverage of all CSFs) as we can see in Table 1 that many studies do not cover all the types of CSFs. The classification of the CSFs proposed by Ahimbisibwe et al. (Ahimbisibwe et al. 2015) seems among the most suitable one in this set of studies, which we choose in this study and review in more detail in Section 2.3.

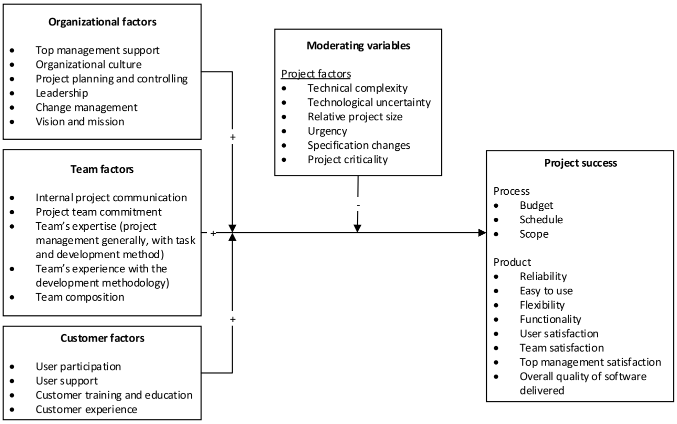
2.3 Reviewing and extending the CSFs model of Ahimbisibwe et al.

As discussed above, we systematically assessed and chose as the base model the CSF model developed by Ahimbisibwe et al. (Ahimbisibwe et al. 2015), which provides a framework for investigating the internal and external factors leading to success or failure in software projects.

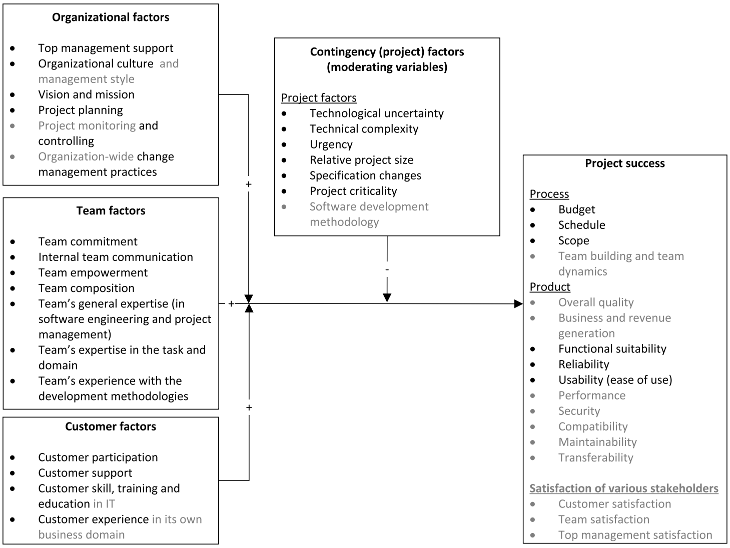
After reviewing Ahimbisibwe et al.'s model (Ahimbisibwe et al. 2015), we found several improvement opportunities which we decided to perform and then use it in our study, e.g., we added several new CSFs and revised a few of the CSFs. Fig. 1 shows both the original model as developed by Ahimbisibwe et al. (Ahimbisibwe et al. 2015) (part a) and also our improved version (part b). In this model, CSFs are categorized into organizational factors, team factors, customer factors, and moderating variables (contingency / project factors). The variables describing project success are included under Project Success. In our model, organizational factors, team factors, and customer factors are the factors that could be correlated with the project success (i.e., the dependent variable). Moderating variables (contingency / project)

Table 1 Coverage analysis of the CSF classification schemes proposed in the literature to date

CSFs covered	Studies											
	[21]	[22]	[24]	[23]	[11]	[3]	[6]	[8]	[9]	[10]	[13]	[67]
People and human aspects												
Developers	✓				✓	✓		✓	✓	✓	✓	✓
Users	✓	✓	✓		✓	✓(in finer granularity: commitment, expertise)			✓	✓	✓	
Top management			✓	✓	✓	✓		✓	✓	✓	✓	✓
External agents				✓	✓	✓		✓	✓	✓	✓	
Project team				✓	✓	✓(in finer granularity: commitment, expertise)		✓	✓	✓	✓	
Social interaction		✓	✓	✓	✓	✓	✓(commination)	✓	✓	✓	✓	
Project aspects												
Project characteristics	✓		✓	✓	✓	✓(in finer granularity: complexity, uncertainty, size, urgency)	✓	✓	✓	✓	✓	
Project scope, goals and objectives	✓			✓	✓	✓	✓		✓	✓	✓	✓
Resources			✓	✓	✓	✓	✓(schedule, budget)	✓(time, cost)	✓	✓	✓	✓
Technology	✓			✓	✓	✓			✓	✓	✓	✓
Development processes												
Requirements determination					✓	✓	✓		✓	✓	✓	
Project management		✓		✓	✓	✓		✓	✓	✓	✓	
Use of a standard method	✓	✓			✓	✓			✓	✓	✓	
User participation			✓		✓	✓			✓	✓	✓	
User training				✓	✓	✓			✓	✓	✓	
Management of change	✓	✓	✓	✓	✓	✓	✓		✓	✓	✓	✓
Institutional context												
Organizational properties		✓	✓	✓	✓	✓(in finer granularity: organizational culture, leadership, vision and mission)		✓	✓	✓	✓	✓
Environmental conditions (e.g., product and capital markets)				✓	✓	✓			✓	✓	✓	



(a): The contingency fit model of CSFs for software development projects, as proposed in [3]



(b): The slightly-revised contingency fit model used in this work, as compared to (a). Revised/added aspects are shown in gray font color.

Fig. 1 A contingency fit model of CSFs in software projects which was used in this study

factors) potentially have a moderating effect on the relations between the factors (variables) describing project success, and also have a negative effect on project success (Ahimbisibwe et al. 2015), e.g., more technical complexity of a project would challenge the success of the project.

Project success in Fig. 1 is a set of variables that measure and describe success. Revised/ added aspects are shown in gray color. The rationale/justifications behind the changes to the base model are explained next.

- As a fundamental issue, as a result of feedback from our industry partners, we added "Software development methodology" (a spectrum between traditional and Agile) as a contingency (project) factor.

- “Organizational culture” in the base model was expanded to “Organizational culture and management style” since management style was actually discussed in the work’ of Ahimbisibwe et al. (Ahimbisibwe et al. 2015) but was not included in the model explicitly.
- “Project planning and controlling” in the base model was divided into separate items: (1) project planning, and (2) project monitoring and controlling, since we believe these two factors are better separated as their quality/impacts could differ in the context of a typical project, e.g., project planning could be perfect, while project monitoring and controlling is conducted poorly.
- “Customer skill, training and education” and “Customer experience” in the base model are somewhat unclear and overlapping with one another, and as a result, based on reviews (Ahimbisibwe et al. 2015) and the relevant literature, we expanded them to “Customer skill, training and education in IT”, and “Customer experience in its own business domain” since the two aspect shall be separated as a customer may have a good skill level in one and not in the other.
- As a result of our recent research (Karapıçak and Demirörs 2013; Garousi and Tarhan 2018) on studying the need to consider personality types in software team formations, and discussion with and feedback from our industry partners in the course of our long-term industry-academia collaborations (e.g., (Garousi and Herkiloğlu 2016; Garousi et al. 2016a; Garousi et al. 2016b; Garousi et al. 2015a)), we added “Team building and team dynamics” as a new CSF to the ‘process’ category of project success. Throughout our industrial projects, we have observed many times that one side product of successful projects is the project’s influence in positive team building, improving team dynamics (compared to the state before the project), and enhancing team members’ opinions about each other. Previous studies have also reported the importance of this factor (Hoegl and Gemuenden 2001).
- We revised the independent-variable category of “project success” to make it better align with well-known software quality attributes (both functional and non-functional) by adding several other quality attributes: performance, security, compatibility, maintainability, and transferability. We revised “ease of use” to “usability” to ensure a broader meaning.
- According to many studies and also the belief of many practitioners, customer satisfaction, e.g., (Ellis 2005; Pinto et al. 2000), and team satisfaction, e.g., (Thompson 2008), are often among the most important project metrics. To address this issue in the model, we added a new category under project success about satisfaction of stakeholders including customer, team and top-management satisfaction.

3 Research goal and methodology

3.1 Goal and research questions

The goal of the empirical investigation reported in this paper was to characterize and analyze the relationship of CSFs (as proposed by Ahimbisibwe et al. (Ahimbisibwe et al. 2015) and modified in our work as discussed above) with the success level of software projects, from the point of view of software project managers in the context of software projects, for the purpose of benefitting software project managers, and also advancing the body of evidence in this area.

Based on the above goal, we raised the following research questions (RQs):

- RQ 1: How strongly are CSFs and project success metrics associated (correlated)?
- RQ 2: What is the ranking of the proposed CSFs based on their correlations with project success metrics? Note that RQ 2 is a follow-up to RQ 1. Once we numerically assess the correlation between the CSFs and project success metrics, we can rank the CSFs based on their correlation values.
- RQ 3: What is the ranking of the project success metrics based on their correlations with the CSFs? Similar to RQ 2, RQ 3 is also a follow-up to RQ 1. Once we numerically assess the correlation between the CSFs and project success metrics, we can rank the project success metrics.

As the research goal and the RQs show, our investigation is an "exploratory" study (Runeson and Höst 2009). To collect data for our study, we designed and executed an online questionnaire-based opinion survey. We discuss the survey design and execution in Section 3.2.

3.2 Survey design and execution

We discuss next the sampling method, survey questions, and survey execution.

3.2.1 Sampling method

In survey-based studies, there are two types of sampling methods, which are probabilistic sampling, and non-probabilistic sampling (Punter et al. 2003; Linåker et al. 2015; Groves et al. 2009). The six sampling methods under probabilistic and non-probabilistic sampling sorted in descending order by the level of sample randomness are (Punter et al. 2003; Linåker et al. 2015; Groves et al. 2009): (1) random sampling, (2) stratified sampling, (3) systematic sampling, (4) convenience sampling, (5) quota sampling, and (6) purposive sampling. Probabilistic sampling is a systematic approach where every member of the entire population is given an equal chance to be selected. Where probabilistic sampling is not possible, researchers use non-probabilistic sampling and take the charge in selecting the sample. In this study, probabilistic sampling and its methods (i.e., stratified sampling) were not possible options since we did not have access to a large dataset to apply probabilistic sampling (also discussed in Section 6.2).

A-posteriori probability-based (systematic) sampling, where data for various companies is grouped and then a sample is extracted from the data so that every member of the population has statistically an equal chance of being selected, is a way to mimic probability sampling. However, since the participants would have hesitated to report honest opinions if their names or employing companies were not anonymous (such experience have been observed many times before, e.g., (Groves et al. 2009)), such revealing information would have seriously damaged the quality of the project related data reported by participants.

In convenience sampling, a type of non-probabilistic sampling method, "*Subjects are selected because of their convenient accessibility to the researcher*" (Lunsford and Lunsford 1995). In all forms of research, it would be ideal to test the entire population, but in most cases, the population is just too large that it is impossible to include every individual. This is the reason why most researchers rely on sampling techniques like convenience sampling, the most common of all sampling techniques used in software engineering research (Sjoeberg et al. 2005). We decided to use “convenience sampling”, after careful review of the pros and cons and also given the constraints that we elaborated above.

Another issue in our survey design, which is inter-related with the sampling method, is the unit of interest (analysis). It has been shown in prior research that participants assuming different roles have different perceptions of project success for the same project (Agarwal and Rathod 2006; Davis 2014). Therefore, the unit of analysis in this survey is the survey participants’ perception of completed or near-completion software projects. Thus, for all the statistics and analysis that we report in Section 4, software projects as perceived by survey participants are the unit of analysis, and the implications shall be tied to the Turkish software projects.

3.2.2 Survey questions

We designed an online survey based on the contingency fit model of CSFs for software projects as explained in Section 2.3. We designed a draft set of questions aiming at covering the CSFs in the contingency fit model while also economizing the number of questions.

Based on our experience in conducting online surveys (Garousi and Zhi 2013; Garousi and Varma 2010; Garousi et al. 2015b; Akdur et al. 2015) and also based on several survey guidelines and experience reports, e.g., (Punter et al. 2003; Linåker et al. 2015; Torchiano et al. 2017); after initial design of the survey, we sent the questions to eight of our industrial contacts as the pilot phase. The goal behind this phase in our survey design was to ensure that the terminology used in our survey was familiar to the participants. This is because, as other studies have shown, e.g., (Garousi and Felderer 2017), researchers and industrial practitioners often use slightly different terminologies and, when conducting joint studies/projects, it is important to discuss and agree on a set of consistent terminology. The feedback from the industrial practitioners was used to finalize the set of survey questions, while their answers to the survey questions were not included in the survey data. Early feedback from the practitioners was useful for ensuring the proper context for the survey and also the familiarity of its terminology for industrial respondents.

After improvement of the questions based on the feedback from the industry practitioners in the pilot study, we ended up with having 53 questions. The complete list of the questions used in the survey is shown in Appendix Table 10. The first set of 12 questions gathered profiles and demographics of the projects, participants, and companies. Most of these questions had quantitative pre-designed single or multiple response answers. For example; the first question about participants’ gender required a single response answer (male or female), while participants were able to provide several answers to the second question (the participant’s current position) as they could have more than one single role in their employer organizations.

The remaining of the questions probed data related to software project in the context of the set of CSFs and metrics describing project success. There was a one-to-one relationship between CSFs/variables describing project success and the questions #13–53. For brevity, we are not showing the pre-designed possible answers of the questions in Appendix Table 9, but they can be found in an online resource (Garousi et al. 2016c). As we can see in Appendix Table 10, these questions (questions 13–53) had answers of type 5-point Likert scale and participants were also given “Not applicable” and “I do not know” choices in each question.

3.2.3 Recruitment of subjects and survey execution

To execute the survey, the survey instrument was implemented by using the Google Forms tool (docs.google.com/forms) and hosted on the online file storage service by Google Drive (drive.google.com). Research ethics approval for the survey was obtained from Hacettepe University's Research Ethics Board in August 2015. Immediately after getting ethics approval, the survey was made available to participants in the time period of August–September 2015. Participants were asked to provide their answers for software projects that were either completed or were near completion. Participation was voluntary and anonymous. Respondents could withdraw from the survey at any time.

To ensure that we would gather data from as many software projects as possible, we established and followed a publicity and execution plan. We sent email invitations to our network of partners/contracts in 55 Turkish software companies (and their departments). In the email invitations, we asked the survey participants to forward the invitation email (which included the survey URL) to their contacts as well. The goal was to maximize the outreach of the survey. Thus, although we were able to monitor the progress weekly, it was not possible for us to determine the ratio of the practitioners who received the email actually filled out the survey (i.e., the response rate). We also made public invitations to the Turkish software engineering community by posting messages in social media, e.g., LinkedIn (we found seven software-engineering-related LinkedIn groups), Facebook, and Twitter, and by sending emails to management offices of more than 30 Technology Development Zones and Research Parks in Turkey.

As we also discussed above, we established and followed a publicity and execution plan in our survey. In order to not to disturb practitioners with multiple duplicate invitations (received from multiple sources), we communicated our invitation with each organization through a single point of contact. Our plan included the target sectors, contact person, publicity schedule and status for each organization. We did not keep track of the response rates since participation in our survey was anonymous, however an iterative publicity schedule (with three main iterations) allowed us to not only monitor the changes in survey population but also to get a feeling about the response rates after each iteration.

An issue about recruitment of subjects (i.e. completed or near-completion software projects) was that, in some cases, various participants might have provided data for the same project. This problem is called "duplicate entries", where two or more data points represent the same unit under study (Linåker et al. 2015). After careful review and discussions, we were convinced that although this issue could lead to a validity threat, the nature of our study that assesses the success factors of software projects as perceived by the participants would allow these kinds of duplicate entries in our data set. This is so since, we assumed that different project stakeholders of the same project would have different perceptions regarding the success factors. So that, as long as each data unit is about a different project and/or from a different participant, it represents a different unit in our data set. After all, we did not expect to have duplicate project data provided by the same respondents.

Also, similar to previously-conducted online SE surveys, e.g., (Garousi and Zhi 2013; Garousi and Varma 2010; Garousi et al. 2015b), we noticed that a number of participants left some of the questions unanswered. This is unfortunately a usual phenomenon in almost any online SE survey. Similar to our surveys in Canada (Garousi and Zhi 2013; Garousi and Varma 2010) and Turkey (Garousi et al. 2015b; Akdur et al. 2015), we found partial answers useful, i.e., even if a given participant did not answer all the 53 questions, her/his answer to a subset of questions were useful for our data set. The survey included data from 101 software projects across Turkey. For replicability and also to enable other researchers to conduct further analysis on our dataset, we have made our raw dataset available in an online resource (Garousi et al. 2016d) (https://dx.doi.org/ https://doi.org/10.6084/m9.figshare.4012779).

3.3 Quality assessment of the dataset

We did sanity checks on the data set and ensured the quality of the dataset. One important issue was the issue of partially-answered questions and their impact on the dataset. We looked into the number of provided (complete) answers to each question of the online survey and show the numbers in Fig. 2. We see that generally, among the 101 data points, many questions had almost close to complete answers, except five questions, which were:

- Q8: Is your company/team certified under any of the process improvement models (e.g., CMMI, ISO 9000, ISO/IEC 15504)? (72 of the 101 respondents answered this), question completion rate = 71.2%: We suppose that, since majority of the respondents were developers working on the project basis, and process improvement activities are usually conducted organization-wide, we could interpret that many respondents (close to 30) simply were not aware of (formal) process improvement activities based on standard models in their companies and thus they left this question unanswered.
- Q9: If your company/team has been certified under CMMI or ISO/IEC 15504, what is its process maturity certification level? Question completion rate = 80.1%. This question is similar and a follow-up to Q8 (process improvement models) and thus the case of many unanswered relies to it.
- Q18: Organization-wide change management practices: Question completion rate = 86.1%. Since change management is usually conducted organization-wide or by dedicated configuration-management teams, we felt that the developers, whose involvement and awareness in change-management practices were low, would have hesitated to provide answers.
- Q37: To what extent did the project finish on budget? Question completion rate = 74.2%. Since budgeting is arranged in management levels, the respondents who were developers could most probably not answer this.
- Q42: To what extent has the software product/service been able to generate revenues for the organization? Question completion rate = 85.1%.

Since there was no cross-dependency among the questions except Q8 and Q9, partial data would not and could not have impacted the internal and construct validity of our study. Also,

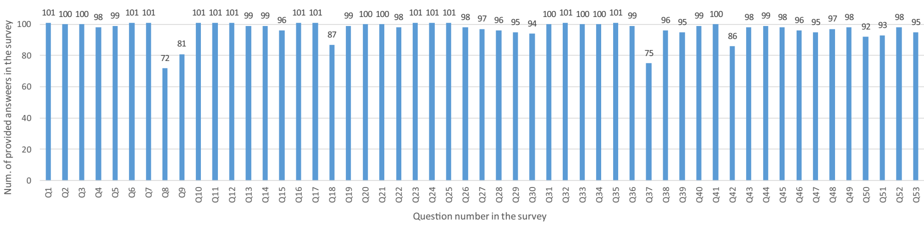


Fig. 2 Number of provided (complete) answers to each question of the online survey

since empty answers were not all overlapping on the same data points, excluding partial answers would have excluded a large ratio of our dataset. Therefore, we did not exclude partial answers.

Furthermore, we wanted to assess the potential impacts of partially-completed answers on results of study in Section 5. Out of the above five partially-answered questions, two are about general profiles and demographics and were not included in results analysis: Q8 (certifications of process improvement practices) and Q9 (company's process maturity certification level). Only three questions, i.e., Q18 (change management practices), Q37 (finishing projects on budget) and Q42 (products/services generating revenues) are included in the dataset used for results analysis in Section 5. When we inspected the data for these questions in particular, we found that empty answers were not all overlapping on the same data points. For the purpose of correlation analysis, when a given CSF or project success measure was not provided in a data point, only for correlation analysis of that CSF, that data point was not considered. In this way, we ensured the soundness of our correlation analysis approach.

4 Demographics of the dataset and independent statistics of the factors

4.1 Profiles and demographics

In order to assess the nature of the set of projects and participants appearing in this study, we extracted and analyzed the profiles and demographics of the participants and the projects in the survey dataset. Since high diversity within the data set was something we wished to achieve, we wanted to understand the level of diversity within the data set and to determine the profiles of the respondents that participated in the projects and the companies that carried out the projects.

4.1.1 Participant profile

In terms of the respondents' gender profile, 70% of the project data were provided by male participants, while 30% were provided by female participants.

We were interested about the positions held by the respondents in projects, since people in different software engineering positions often have different perceptions in rating the importance of success factors (Garvin 1988). It is possible that as participants might have more than one position in a given project (e.g., a person can be a developer and tester at the same time). Thus, respondents were allowed to select more than one position. As presented in the bar chart in Fig. 3, most of the respondents were software developers (programmers). We can see that software designer, software architect, and projects manager positions were also common positions in the dataset.

The survey also asked about the city of residence of each respondent. As most of the authors and most of our industry contacts were located in Ankara (the capital city of Turkey), it is not surprising that 71% of the projects were reported by participants from Ankara. Istanbul, the largest city of Turkey, followed Ankara with 23% of the dataset. A 2012 study reported that 47 and 33% of the Turkish software companies were located in Istanbul and Ankara, respectively (Akkaya et al. 2012). Although the geographic distribution of our sample pool does not perfectly fit the regional breakdown of Turkish software companies, there is a

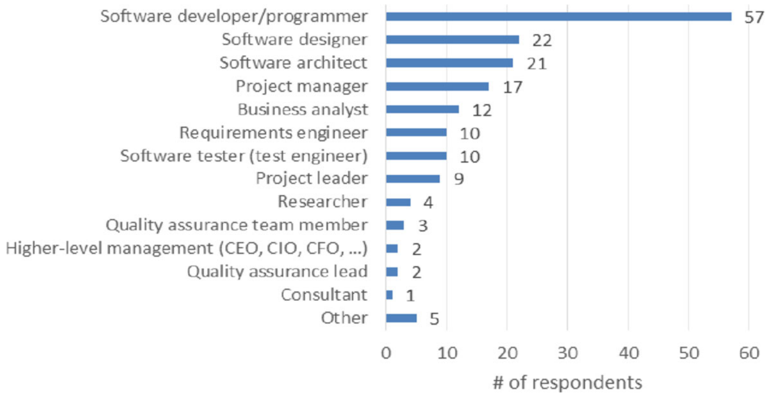


Fig. 3 Respondents' positions

reasonable mix of respondents from Turkish cities where the majority of the software industry is located.

Figure 4 shows the individual-value plot of the respondents' years of work experience. The plot and data's mean and median values (8.0 and 6.5, respectively) denote a young pool of participants, but still there is a good mix of participants with various levels of experience in software projects.

Another survey question was about the certifications held by respondents. Most popular certifications held by respondents are Agile-related certifications, certifications offered by the Project Management Institute (PMI) and by Microsoft Corporation. Certifications reported by our respondents in the "other" category included: certification for the Open Group Architecture Framework (TOGAF), the COSMIC (Common Software Measurement International Consortium), and Capability Maturity Model Integration (CMMI) certifications and certifications offered by SAP Corporation, ISACA (Information Systems Audit and Control Association), and Offensive Security.

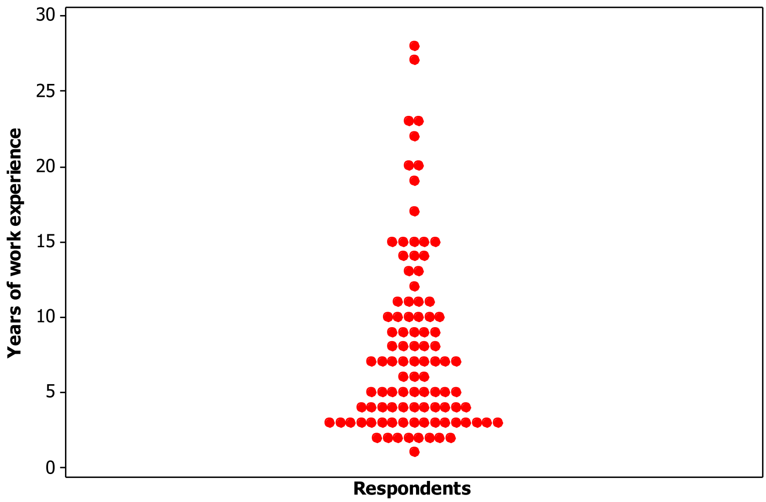


Fig. 4 Individual value plot of respondents' years of work experience

4.1.2 Company profile

Figure 5 shows the results of the next question asking about the target sectors of the products developed (or services offered) by the companies that the projects were performed by. We acknowledge the positive impact of diversity in target sectors of projects, as project information from various target sectors would help increase the representativeness of the dataset and the generalizability of our results. As shown in Fig. 5, projects from companies producing software products for government, IT and telecommunication, military and defense, banking and finance, engineering and manufacturing and business are all represented in our data set.

We wanted to have projects from various company sizes. Thus, another survey question gathered that data. Figure 6 shows the percentage distribution of data points by company size. We can see that there is a good mix of projects from various company sizes.

We also wanted to know about which process improvement models are used in the companies where the projects in our data set are carried out. It makes sense to expect that using process improvement models by a company or team would have a positive impact on project outcomes. CMMI and ISO9000 family of standards, with 46 and 41 data points respectively, were the highly used models in the companies in our survey sample. Other process improvement models were also mentioned, e.g., ISO/IEC 15504: Information technology — Process assessment, also known as Software Process Improvement Capability Determination (SPICE), and ISO/IEC 25010: Systems and Software Quality Requirements and Evaluation. Note that as a company might use more than one process improvement model, participants were allowed to report multiple models.

4.1.3 Project profile

As discussed in Section 3.2, we asked participants to fill the survey in the context of their completed or near completion projects. However, we realized that not all software projects are in the scope of the entire software development life-cycle phases, e.g., in several of our current industry-academia collaborative projects, the project scopes have only been in software maintenance or only in testing. Thus, we wanted to know the number and types of phases followed in projects. 88 of the projects in our survey followed the entire software development life-cycle. Most of the remaining projects followed at least two phases/activities. Among projects that were consisted of only one phase/activity, five were software requirements / business analysis projects, and one was a software testing project.

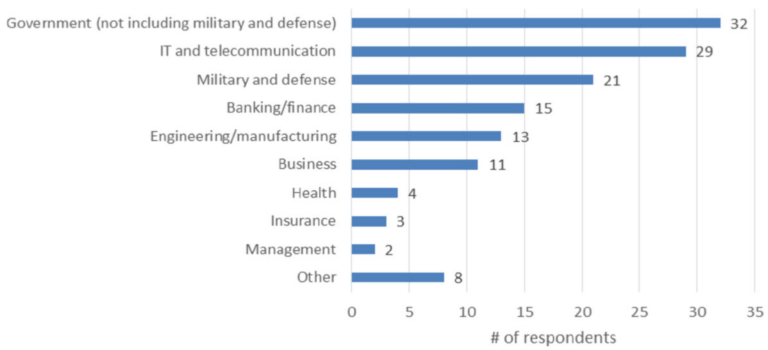
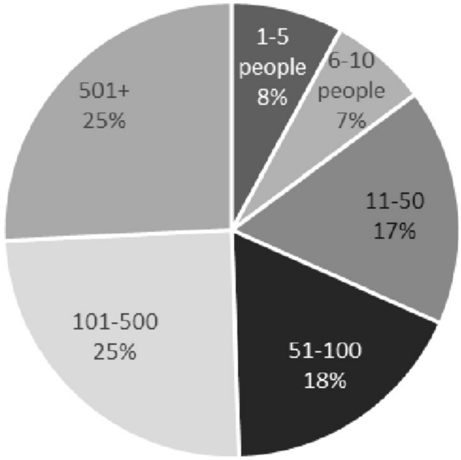


Fig. 5 Target sectors of the products developed (or services offered) by companies in the dataset

Fig. 6 Company size (number of employees)



Participants were asked about the nature of the projects based on development type, where they were allowed to report multiple development types, e.g., a project might be a new "development" project or a software maintenance project. The results reveal that new development projects are prevalent in our dataset with 82 data points. However, there is also a good representation of software maintenance projects of adaptive, perfective, corrective and preventive types with 38, 21, 21, and 13 data points respectively.

As discussed in Section 3.2, the measurement unit of this study is a single software project, but in many cases software development activities exist as a part of a larger project. In order to understand the frequency of such cases in our survey data and ensure that participants would provide answers related only to 'software' aspects of larger system-level projects, participants were asked about the independence level of the projects. Results show that about 62% of the projects were stand-alone whereas the remaining 38% were part of larger projects. Note that we asked participants to answer rest of the questions in the survey based on only the project/activity that is part of the larger project, if they had selected the latter choice.

4.2 Independent statistics of the factors and variables

In this study, as explained in Sections 2 and 3, we grouped the factors and variables under three categories: Critical success factors (CSFs), contingency (project) factors, and project success metrics. Similar to what we expected in profiles and demographics, we wanted diversity in the projects in our survey in terms of these factors and variables. With this aim, we did not encourage participants to report about successful or failed projects, but we let them free to determine the projects that they wished to report in the survey, as long as projects were either completed or near completion. We discuss next the factors under each of the three categories.

4.2.1 Critical success factors

As discussed in Section 2.3, the revised CSF model groups factors that are could relate to the project success under three categories: (1) organizational factors, (2) team factors, and (3) customer factors. We report next the independent statistics of the factors in each these groups.

Organizational factors Based on the CSF model, we asked participants about the organizational factors below:

- The level of top-management support
- (Maturity of the) organizational culture and management style
- (Maturity of the) vision and mission
- (Maturity of the) project planning
- (Maturity of the) project monitoring and controlling
- (Maturity of the) organization-wide change management practices

As discussed in Section 3.2.2, the answers to these questions could be selected from 5-point Likert scales, which was designed carefully for each question, e.g., answers for the question about top-level management support could be chosen from this scale: (1) very low, (2) low, (3) average, (4) high, and (5) very high. Organizational culture and management style could be chosen from: (1) rigid organizational culture (i.e., bureaucratic command and control), (2) slightly rigid, (3) moderate (between rigid and adaptive), (4) slightly adaptive / flexible, and (5) fully adaptive / flexible collaborative organizational culture (i.e., family-like mentoring and nurturing). These categories and Likert scales were largely adopted from the literature, e.g., (Ahimbisibwe et al. 2015).

Distributions (histograms) of the data points in the dataset w.r.t. each of the above factors are shown in Fig. 7, along with median values for each factor. The medians for all cases, except project planning and monitoring, are in "medium" level. For project planning and monitoring, the medians are "Planned" and "High", respectively.

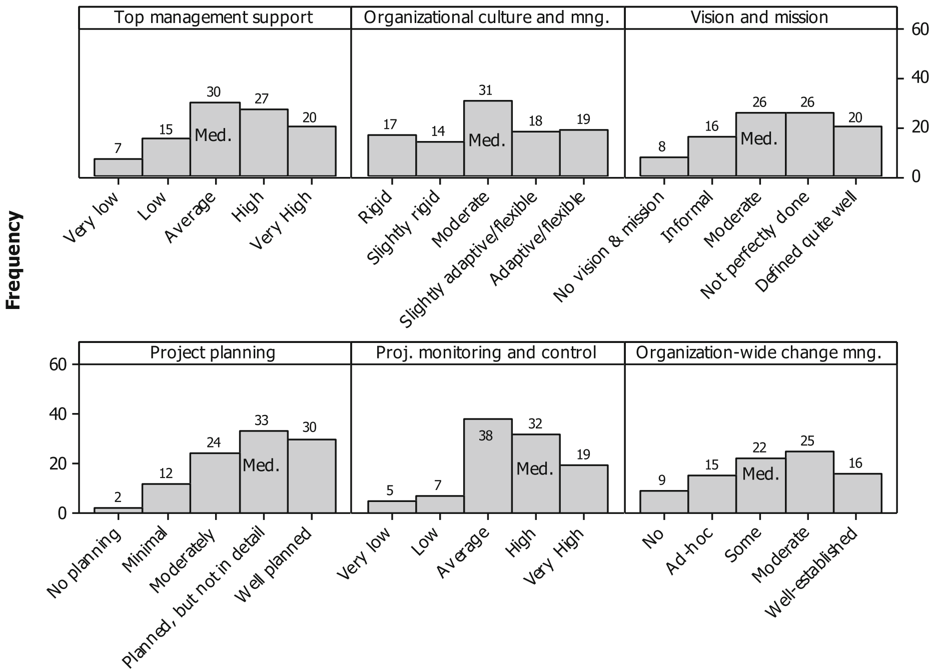


Fig. 7 Distribution (histogram) of the data points w.r.t. organizational factors. *Med.*: median values

The results show that among the organizational factors project planning and project monitoring and controlling were the factors that had a median at position 4 of the 5-point Likert scale. This indicates that participants generally rated the level of these factors in their projects as medium or higher. On the other hand, scores for the organization culture and management style ranged from “rigid” to “adaptive/flexible” almost symmetrically, meaning that we had projects from across that spectrum in the dataset.

Team factors Participants were asked about following team factors:

- Team commitment
- Internal team communication
- Team empowerment
- Team composition
- Team’s general expertise (in software engineering and project management)
- Team’s expertise in the task and domain
- Team’s experience with the development methodologies

Distributions (histograms) of the data points are shown in Fig. 8. Project team commitment, project communication among internal stakeholders during the project under study, and level of team members’ expertise and competency with the tasks assigned are rated generally high, according to the median values shown in Fig. 8. It is noteworthy that median of scores for project teams’ composition and team empowerment are lower than they are for other team

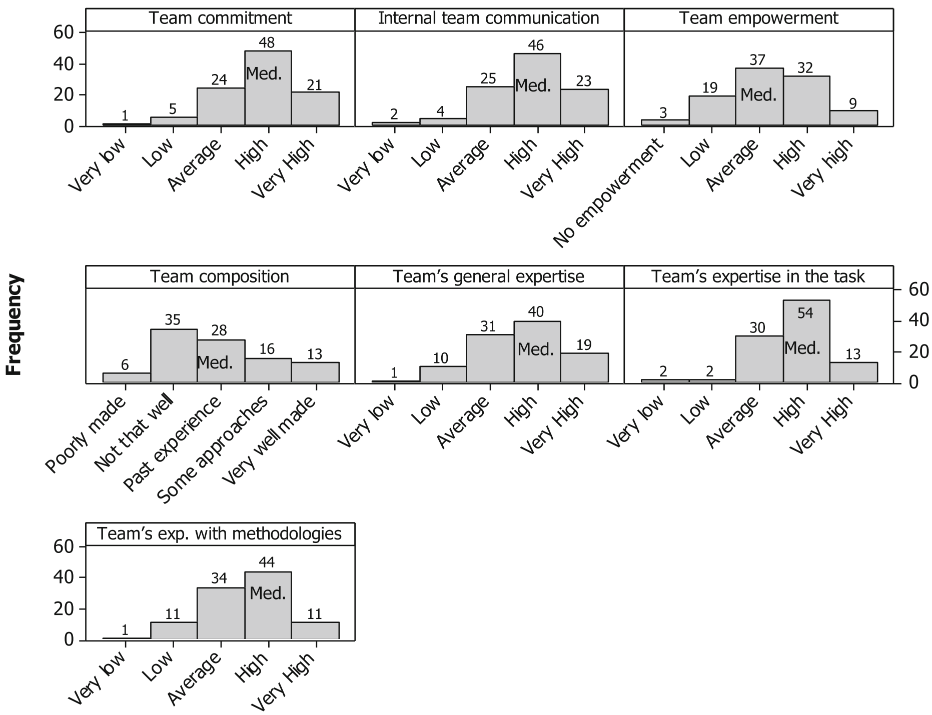


Fig. 8 Distribution of the data points w.r.t. team factors

factors. This denotes that some participants seemed not to be pleased with the formation of the team in terms of cohesion, team sense, harmony, and respect among team members in some projects.

Customer factors The survey asked about the following customer factors:

- Customer (client) participation
- Customer (client) support
- Customer skill, training and education in IT
- Customer experience in its own business domain

Figure 9 shows the distributions (histograms) of the data points w.r.t the above customer factors. We can see that level of customer (client) experience in their own business domain in projects is rated highest among customer factors. We have a good mix of scores for customer participation, customer support, and customer IT skills, training and education in projects. We see that while there was high customer participation and support in some projects, those factors were quite poor in some other projects.

4.2.2 Contingency (project) factors

Based on the revised CSF model, the survey asked the participants about following contingency (project) factors (i.e., moderating variables) in their projects:

- Technological uncertainty
- Technical complexity
- Urgency
- Relative project size
- Specification changes

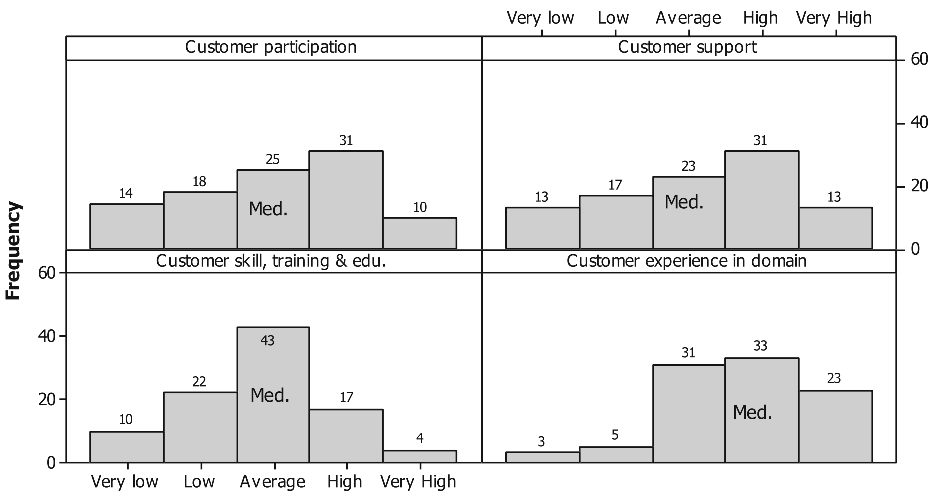


Fig. 9 Distribution of the data points w.r.t. customer factors

- Project criticality
- Software development methodology

Distribution of the data points including the median values for each factor are shown in Fig. 10. Technological uncertainty in the projects mostly ranges between very few and moderate, whereas project complexity has mostly moderate to high scores. The response histograms for both the level of urgency and level of specification changes are slightly skewed towards the right, indicating that the level of these two factors as experienced in most projects was moderate to high. Project size in terms of number of team members are mostly between 11 and 50 people or lower. However, there exist also projects in our survey that are very large (this refers to projects having with more than 100 team members).

The response histogram for project criticality, on the other hand, is more visibly skewed towards the right showing that our project portfolio includes mostly critical projects. Software development methodologies used in projects are distributed quite evenly in the 5-point Likert scale between Agile and plan-based (traditional).

4.2.3 Project success

Variables describing project success in our contingency fit model (Fig. 1) include: (1) process success, (2) product-related factors, and (3) stakeholders' satisfaction. In order to understand the process outcomes and process success, we asked in the survey about the following factors:

- Budget
- Schedule
- Scope
- Team building and team dynamics

Figure 11 shows the distribution (histogram) of the data points. We show the median values for each histogram. We can see that most projects were in good shape regarding

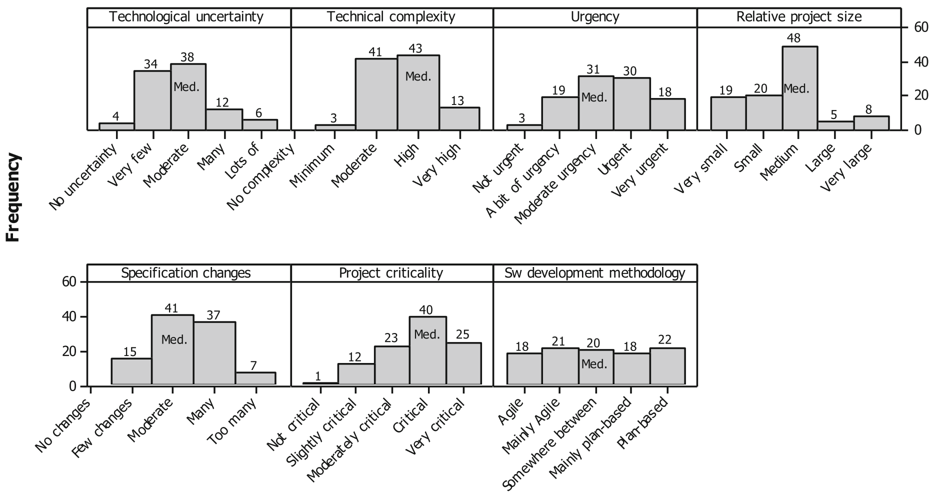


Fig. 10 Distribution (histogram) of the data points w.r.t. contingency (project) factors

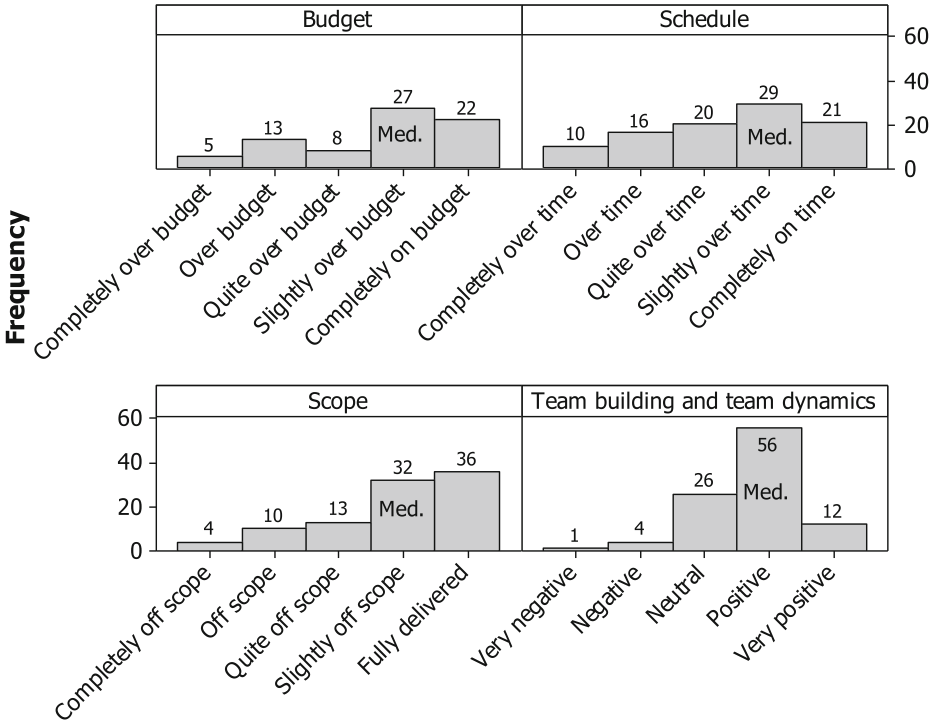


Fig. 11 Distribution (histogram) of the data points w.r.t. process characteristics

team building and team dynamics factors. Scores for other three project success variables show that most projects were also in reasonable shape and not challenged in terms of conforming to project scope, budget, and schedule. In terms of projects finishing on budget, 21 and 5 (of 101) of the corresponding responses were "I do not know" and "not applicable" options, respectively. This seems to denote that survey participants were relatively challenged in assessing that aspect or, as expected, they were not in the top management team and thus were understandably not aware of the projects' budget-related issues.

According to the CSF model (Fig. 1), metrics describing product-related factors were:

- Overall quality
- Business and revenue generation
- Functional suitability
- Reliability
- Performance
- Usability (ease of use)
- Security
- Compatibility
- Maintainability
- Transferability

Variables describing satisfaction of various stakeholders were as follows:

- Customer (user) satisfaction
- Team satisfaction
- Top management satisfaction

The survey asked the respondents about product characteristics and level of stakeholders' satisfaction. Figure 12 shows the results. Overall quality of the products/services perceived by project members were mostly high or average, as shown by median values.

Business and revenue generation in projects is also rated high by participants. However, we should note that a few participants were challenged in assessing this project success variable, as 9 and 6 of the data points were "I do not know" and "not applicable" options respectively.

Transferability is the lowest rated project success variable among rest of the product related variables with the median of 'average'. Besides transferability, histograms for rest of the product-related variables are visibly skewed towards right and these variables are rated mostly high by most participants. Scores of performance and reliability are relatively high which indicates that respondents were mostly happy with these quality aspects of the software/service delivered.

On the overall, level of satisfaction of various stakeholders (i.e., top management, customer, and team members) with the project outputs is high or average in most projects.

We also wanted to have a single metric to measure "overall" success of a given project (data point in the dataset) to be able to assess and differentiate the success distributions of the projects. For this purpose, we calculated an aggregated metric, for each project, by calculating the average on all of its 17 success factors and then normalizing the value into the range of [0,1]. Since we used Likert scales, we assumed that the response data could be treated like an interval scale.[1] We assigned the value '4' to the most positive endpoint of the scale, i.e., when the response indicated that the project was 100% successful with regard to the indicated factor, and the value '0' to the most negative endpoint of the scale, i.e., when the response indicated that the project was 0% successful (or 100% unsuccessful) with regard to the indicated factor. For normalization, we divided the data by four. For example, for a data point with a list of 17 values for its success factors {4, 3, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 4, 1, 4, 4, 4}, the aggregated metric is 3.76 out of 4, or 0.94 after normalization, indicating a highly successful project.

As we can see in Fig. 13, overall success measures of the majority of the projects (in the original dataset) are above the middle line, denoting that the data pool in general is slightly skewed towards more successful projects. Since we did not filter the projects entered in the survey based on their success (the survey was open to any project), this has been the outcome of the data collection. The purpose of the 'balanced' dataset in Fig. 13 and the need for it will be discussed in the next section.

4.3 Creating a balanced dataset based on project success metrics

As reported above, our original dataset is skewed towards successful projects and thus was not "balanced" in terms of the aggregated project success metric as visualized in Fig. 13. For the analysis reported in the rest of this paper, in addition to the original dataset, we wanted to have another balanced dataset in terms of the aggregated success metric in which we would have

[1] We are aware of the debate on calculating the average (mean) of ordinal variables. This is permissible with certain caveats (Belle 2002) G. v. Belle, *Statistical Rules of Thumb*. John Wiley & Sons, 2002., e.g., assuming that the ordinal data are normally distributed, which we also assume in this case.

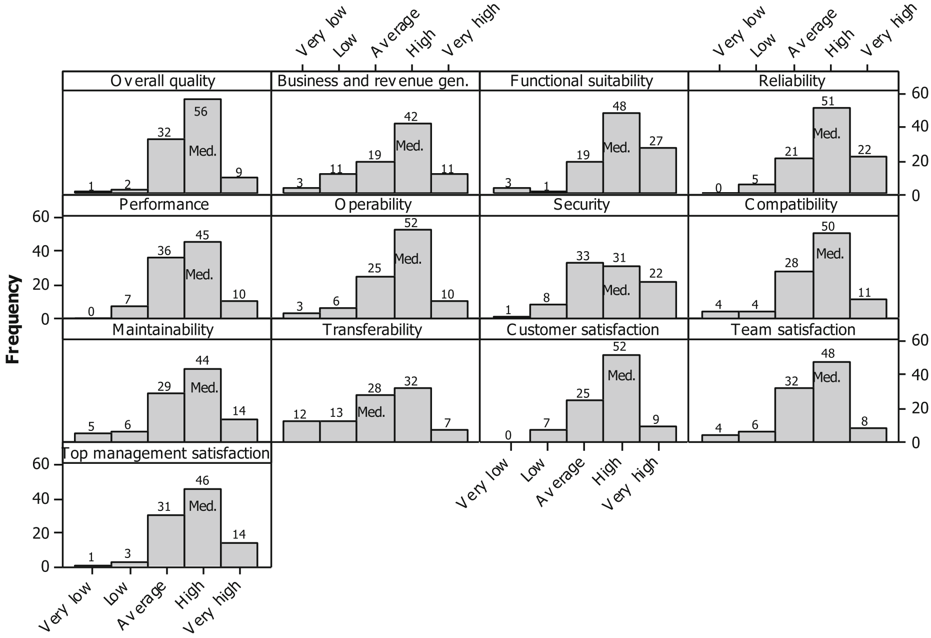


Fig. 12 Distribution of the data points w.r.t. product characteristics and satisfaction of stakeholders

quite similar number of more successful projects and less successful projects. We felt that it is important to also have the balanced dataset so to be able to compare the findings of our study based on original dataset (skewed towards successful projects) with the balanced dataset (one which has similar number of more successful projects and less successful projects). Our approach to derive that balanced dataset from the original dataset was as follows.

We sorted all the data points (projects) in the original dataset by their aggregated success metric. The minimum and maximum values were 0.53 and 3.76 (out of 4) respectively. We then put the middle cut-off (partition) point at 2.0. The category of more successful projects was those with success metric >2.0. Similarly, the category of less successful projects was those with success metric <2.0. Out of 101 projects, based on their success metrics, 80 projects were placed in the more successful category, and the remaining 21 projects were placed in the less successful category. To create a balanced (sub)dataset, we then kept the 21 projects in the less successful category, and randomly sampled 21 projects from the more successful category and joined the two sets to create a balanced dataset with 42 projects. Figure 13 shows the boxplots of the aggregated success metrics for both the original and the balanced dataset. As shown in Fig. 13, the balanced dataset is 'almost' balanced in terms of the aggregated project success metrics. Note that the points (projects) across the two sides of the middle line are not exactly symmetric, thus we do not have the 'perfectly' balanced case. As a result, in going further with our analysis, we use an original dataset that includes all the projects (i.e., 101 data points) and a balanced dataset that includes an equal number of less and more successful projects (i.e., 42 data points).

Also, we should mention any repetition of the random sampling of 21 projects from the set of 80 more successful projects could provide a slightly different balanced dataset. To prevent further complexity in our analysis, we picked one possibility and chose it as the balanced dataset to be used in the analysis of our RQs. Note that as discussed in Section 3.2, for

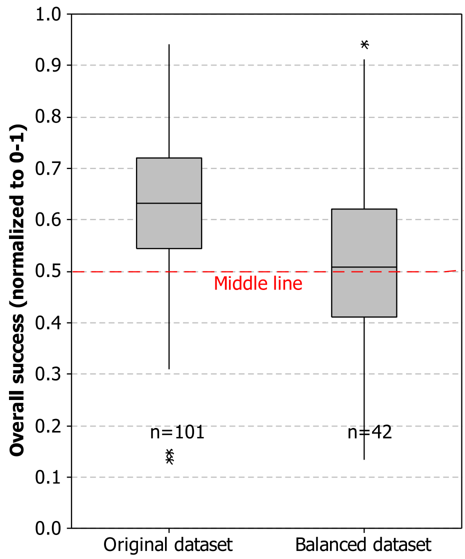


Fig. 13 Boxplot of aggregated normalized overall success of projects

replicability purposes, we have made our raw dataset available in an online resource (Garousi et al. 2016d) which also includes the balanced dataset.

5 Results and findings

We present the results of our study and answer the three RQs in this section. For brevity and to prevent repeating the long names of factors again and again throughout the study, we use codes in this study. The table of codes for factors and variables is provided in Appendix Table 11 (e.g., OF.01 for 'Top management support' factor under organizational factors).

5.1 RQ 1: Correlations between CSFs and project success

For RQ 1, we wanted to assess how strongly CSFs and project success metrics are correlated. To tackle RQ 1, we conducted a correlation analysis between the four categories of CSFs and the three categories of variables describing project success. We do this first for the original dataset (in Section 5.1.1) and then for the balanced dataset (in Section 5.1.3). How we created the balanced data set was described in Section 4.3. In addition, we analyzed the presence of cross-correlation among the proposed CSFs both for the original and balanced datasets in Section 5.1.3.

5.1.1 Analysis using the original dataset

Table 2 presents the correlation coefficients between proposed CSFs and the variables describing project success. As reviewed in previous sections and shown in Table 2, there are 24 CSFs, categorized into four categories: organizational factors (OF in Table 2), team factors (TF), customer factors (CF), and contingency (project) factors (PF). In the CSF model (Fig. 1), there are 17 variables describing project success, categorized into three categories: (1) process

Table 2 Correlation coefficients (Spearman's rho) between proposed CSFs (rows) and project success metrics (columns) - original dataset

		Process factors (ProcF)				Product-quality factors (ProdF)						
		ProcF.01	ProcF.02	ProcF.03	ProcF.04	ProdF.01	ProdF.02	ProdF.03	ProdF.04	ProdF.05	ProdF.06	ProdF.07
Organizational factors	OF.01	0.21*	0.02	0	0.07	0.13	0.04	0.18	0.21*	0.23*	0.07	0.37**
		W	N	N	N	W	N	W	W	W	N	M
	OF.02	0.12	0.01	0	0.05	0.06	0.07	0.01	0	0.03	0.06	0.02
		W	N	N	N	N	N	N	N	N	N	N
	OF.03	0.1	0.09	0.04	0.23	0.29**	0.08	0.21*	0.39	0.19	0.23	0.39
		W	N	N	W	W	N	W	M	W	W	M
	OF.04	0.30**	0.25*	0.16	0.40**	0.39**	0.31**	0.25*	0.31**	0.17	0.19	0.13
		M	W	W	M	M	M	W	M	W	W	W
	OF.05	0.14	0.15	0.11	0.41**	0.44**	0.33**	0.36**	0.38**	0.29**	0.23*	0.20*
		W	W	W	M	M	M	M	M	W	W	W
	OF.06	0.15	0.18	0.24*	0.34**	0.31**	0.33**	0.35**	0.36**	0.1	0.29**	0.25*
		W	W	W	M	M	M	M	M	W	W	W
Team factors	TF.01	0.03	0.06	−0.14	0.46**	0.25*	0.22*	0.28**	0.24*	0.16	0.27**	0.18
		N	N	W	M	W	W	W	W	W	W	W
	TF.02	0.06	0.08	0.11	0.48**	0.31**	0.17	0.24*	0.25*	0.11	0.22*	0.28**
		N	N	W	M	M	W	W	W	W	W	W
	TF.03	−0.05	0.15	0.18	0.24*	0.28**	0.07	0.25*	0.23*	0.21*	0.1	0.16
		N	W	W	W	W	N	W	W	W	W	W
	TF.04	0.02	−0.01	0.09	0.39**	0.34**	0.23*	0.17	0.21*	0.23*	0.32**	0.24*
		N	N	N	M	M	W	W	W	W	M	W
	TF.05	0.05	0.03	0.22*	0.32**	0.37**	0.28**	0.24*	0.25*	0.29**	0.14	0.19
		N	N	W	M	M	W	W	W	W	W	W
	TF.06	0.05	0.06	0.31**	0.36**	0.45**	0.31**	0.33**	0.34**	0.26**	0.2*	0.16
		N	N	M	M	M	M	M	M	W	W	W
	TF.07	0.06	0.05	0.09	0.28**	0.44**	0.34**	0.29**	0.29**	0.21*	0.26**	0.37**
		N	N	N	W	M	M	W	W	W	W	M
Customer factors	CF.01	0.09	0.11	−0.05	0.26**	0.16	0.16	0.29**	0.29**	0.14	0.12	0.05
		N	W	N	W	W	W	W	W	W	W	N
	CF.02	0.06	0.05	0.03	0.27**	0.15	0.11	0.13	0.23*	0.05	0.07	−0.05
		N	N	N	W	W	W	W	W	N	N	N
	CF.03	0.15	0.09	−0.03	0.25*	0.16	0.06	0.08	0.24*	0.3	0.18	0.05

Table 2 (continued)

	Process factors (ProcF)				Product-quality factors (ProdF)						
	ProcF.01	ProcF.02	ProcF.03	ProcF.04	ProdF.01	ProdF.02	ProdF.03	ProdF.04	ProdF.05	ProdF.06	ProdF.07
	W	N	N	W	W	N	N	W	M	W	N
CF.04	0.08	−0.09	0.18	0.09	0.11	0.16	0.23*	0.18	0.06	0.17	0.17
	N	N	W	N	W	W	W	W	N	W	W
PF.01	−0.1	−0.05	−0.18	−0.11	−0.04	−0.1	−0.2*	−0.15	0.08	−0.12	0.02
	W	N	W	W	N	W	W	W	N	W	N
PF.02	−0.12	−0.15	0.13	0.04	0.09	0.31**	0.2*	0.06	−0.19	−0.02	0.11
	W	W	W	N	N	M	W	N	W	N	W
PF.03	−0.13	−0.03	−0.08	0.03	−0.15	0.01	−0.11	0	0.05	0.04	0.2*
	W	N	N	N	W	N	W	N	N	N	W
PF.04	−0.01	0.01	0.01	0.13	0.05	0.32**	0.2*	0.06	−0.07	0.01	0.14
	N	N	N	W	N	M	W	N	N	N	W
PF.05	−0.14	−0.04	−0.13	0.13	−0.14	−0.03	−0.1	−0.06	−0.1	−0.12	−0.1
	W	N	W	W	W	N	W	N	W	W	W
PF.06	0	−0.09	0.2*	0.09	0.02	0.18	0.17	0.22*	−0.1	0.02	0.19
	N	N	W	N	N	W	W	W	W	N	W
PF.07	−0.13	−0.15	0	−0.12	−0.01	−0.13	−0.1	0.02	0.03	0.06	0.22*
	W	W	N	W	N	W	W	N	N	N	W
None	12	17	11	6	6	7	2	6	7	8	5
Weak	11	7	12	10	10	10	19	13	16	15	16
Moderate	1	0	1	8	8	7	3	5	1	1	3
Strong	0	0	0	0	0	0	0	0	0	0	0
Avg.	0.01	0.02	0.06	0.21	0.19	0.15	0.16	0.19	0.11	0.12	0.16
Rank	17	16	14	2	5	9	6	4	13	11	7

	Product-quality factors (ProdF)			Satisfaction factors (SF)			# of corr. Classifications				
	ProdF.08	ProdF.09	ProdF.10	SF.01	SF.02	SF.03	N	W	M	S	Avg. Corr.
Organizational factors	0.03 N	0.11 W	0.05 N	0.14 W	0.17 W	0.11 W	7	9	1	0	0.10

Table 2 (continued)

	Product-quality factors (ProdF)			Satisfaction factors (SF)			# of corr. Classifications				
	ProdF.08	ProdF.09	ProdF.10	SF.01	SF.02	SF.03	N	W	M	S	Avg. Corr.
	0.02	0.06	0.04	0.04	0	0.2*					
	N	N	N	N	N	W	15	2	0	0	−0.02
	0.27	0.2	0.06	0.15	0.28	0.28					
	W	W	N	W	W	W	4	11	2	0	0.18
	0.35**	0.25*	0.15	0.25*	0.30**	0.49**					
	M	W	W	W	M	M	0	9	8	0	0.27
	0.34**	0.23*	0.21*	0.30**	0.37**	0.54**					
	M	W	W	M	M	S	0	8	8	1	0.30
	0.21*	0.19	0.05	0.20*	0.42**	0.47**					
	W	W	N	W	M	M	1	9	7	0	0.26
Team factors	0.21*	0.22*	0.15	0.23*	0.38**	0.20**					
	W	W	W	W	M	W	2	13	2	0	0.20
	0.35**	0.32**	0.11	0.23*	0.45**	0.29**					
	M	M	W	W	M	W	2	10	5	0	0.24
	0.28**	0.20*	0.09	0.24*	0.36**	0.25*					
	W	W	N	W	M	W	3	13	1	0	0.19
	0.38**	0.20*	0.19	0.19	0.35**	0.25*					
	M	W	W	W	M	W	3	9	5	0	0.22
	0.35**	0.32**	0.02	0.27**	0.32**	0.32**					
	M	M	N	W	M	M	3	8	6	0	0.23
	0.22*	0.33**	0	0.34**	0.34**	0.33**					
	W	M	N	M	M	M	3	4	10	0	0.26
	0.52**	0.38**	0.26**	0.23*	0.27**	0.29**					
	S	M	W	W	W	W	3	9	4	1	0.27
Customer factors	0.07	0.04	0.08	0.24*	0.43**	0.4**					
	N	N	N	W	M	M	6	9	2	0	0.17
	0.06	0.02	−0.05	0.24*	0.33**	0.33**					
	N	N	N	W	M	M	9	6	2	0	0.12
	0.15	0.16	0.22*	0.1	0.32**	0.16					
	W	W	W	W	M	W	5	10	2	0	0.16
	0.15	0.09	−0.01	0.15	0.25*	0.13					

Table 2 (continued)

	Product-quality factors (ProdF)			Satisfaction factors (SF)			# of corr. Classifications				
	ProdF.08	ProdF.09	ProdF.10	SF.01	SF.02	SF.03	N	W	M	S	Avg. Corr.
	W	N	N	W	W	W	6	11	0	0	0.12
	−0.08	0.01	0.03	−0.13	−0.11	−0.15					
	N	N	N	W	W	W	7	10	0	0	−0.08
	0.12	−0.02	−0.16	−0.12	0.07	0.16					
	W	N	W	W	N	W	6	10	1	0	0.03
	0.13	0.05	0.18	−0.14	−0.03	−0.06					
	W	N	W	W	N	N	10	7	0	0	0.00
	0	−0.03	−0.21*	−0.03	0.03	0.12					
	N	N	W	N	N	W	11	5	1	0	0.04
	−0.17	−0.15	−0.16	−0.13	−0.09	0					
	W	W	W	W	N	N	5	12	0	0	−0.09
	0.1	0	−0.18	−0.04	0.18	0.31					
	W	N	W	N	W	M	7	9	1	0	0.07
	−0.21*	0.1	0.05	−0.2*	−0.23*	−0.26**					
	W	W	N	W	W	W	6	11	0	0	−0.06
	6	9	12	3	5	2					
	12	11	12	19	7	14					
	5	4	0	2	12	7					
	1	0	0	0	0	1					
	0.16	0.13	0.05	0.11	0.22	0.20					
	8	10	15	12	1	3					

quality (*ProcF.01*, …, *ProcF.04* in Table 2), (2) product quality (*ProdF.01, ..., ProdF.10*), and (3) satisfaction of stakeholders (*SF.01, ..., SF.03*). Since our data is ordinal, we used Spearman's rank correlation (rho) to assess the association (relationship) among the variables. We also calculated the level of significance for each correlation and show them in Table 2: '*' (*p* value <= 0.05) and '**' (p value <= 0.01).

We should note the issue of possible family-wise error rate in this analysis, which is "*the probability that the family of comparisons contains at least one Type I error*" (Howell 2013). Since we have compared and assessed the pair-wise correlations (Spearman's rho) between proposed CSFs (rows) and variables describing project success (columns) in Table 2, one should be careful about the family-wise error rate here. One of the widely-used controlling procedures for this possible error is the Bonferroni correction (Howell 2013), in which the threshold for acceptable *p* values for the original pair-wise correlation values should be divided by the total number of comparisons (see page 261 of (Malhotra 2016)). The number of comparisons in Table 2 is $24 \times 17 = 408$. We checked and found that when applying this Bonferroni correction for the chosen p value levels of 0.05 and 0.01, in Table 2, none of the correlations are significant anymore.

For easier analysis, and to not exclusively rely on the level of significance, we classified the correlation values using a widely-used four-categories scheme (Cohen 1988), as shown in Table 3 (None or very weak, Weak, Moderate and Strong). Classified correlation values in our analysis are directly shown by columns N, W, M, S in Table 2. Based on the classified correlation values, we have added the total for each row and column in Table 2. Those total values show, for the case of each CSFs (rows) and project success variables (columns), how strongly it is correlated with the factors in the other group.

Out of the total of 408 combinations in Table 2, we found 168 statistically significant ($p <= 0.05$) associations (98 of which had a p value less than or equal to 0.01) between proposed CSFs and variables describing project success. This meant that about 41% of the combinations (168/408) have statistically significant correlation with variables describing project success. Most of the associations were positive, with only a very small number of negative associations at a statistically significant level (i.e., eight statistically-significant associations in total, with one negative association having $p <= 0.01$). According to the Ahimbisibwe et al.'s CSFs model (Ahimbisibwe et al. 2015), one would generally expect a negative correlation of contingency (project) factors with project success metrics (referring to Fig. 1). However, we found that this expectation was not observed in our dataset. One possible root cause could be the specific mix of successful/unsuccessful projects in our dataset (as reviewed in Section 4.2.3).

Only one contingency (project) factor (out of seven), i.e., PF.07 (software development methodology) is negatively correlated with four (out of 14) of the variables describing project success at a statistically significant level. This means that, considering the Likert scale used for PF.07 (software development methodology), ranging between (1) Agile and

Table 3 Classifying strength of correlation values

Absolute correlation value	Strength of relationship
[0.0–0.1]	None or very weak (N)
[0.1–0.3]	Weak (W)
[0.3–0.5]	Moderate (M)
[0.5–1.0]	Strong (S)

(5) traditional plan-driven, we can say that the more traditional (plan-driven) a development methodology is, according to the correlation findings, it seems that there is a tendency for less satisfaction among all three stakeholder types (user, team members, and top-management). Let us remind in this context that, as it has been widely discussed in the research community, a high correlation does not necessarily mean causality. In the literature, “correlation proves causation” is considered a cause logical fallacy, also known as “cum hoc ergo propter hoc”, Latin for “with this, therefore because of this” (Hatfield et al. 2006). Especially the issue is a subject of active discussions in the medical community, e.g., (Novella 2009; Nielsen 2012). At the same time, however many in the community believe that: “*Correlation may not imply causation, but it sure can help us insinuate it*” (Chandrasekaran 2011). Thus, we only discuss the existence (or lack) of correlations among various project factors and project success in this paper, and we do not intend to imply causality, i.e., if the development methodology is plan-driven, we cannot necessarily say that the stakeholder satisfaction will be low. In fact, one CSF alone cannot guarantee project success. By focusing on and improving all CSFs, a project team can only increase the chances of success.

Interestingly, one of the proposed CSFs, PF.05 (Project Factor – Specification Changes), was not associated with any of the variables describing project success at a statistically significant level ($p <= 0.05$). On the other hand, among the variables describing project success, ProcF.02 (Characteristics of the Process – Schedule), was associated with only one of the proposed CSFs at a statistically significant level ($p <= 0.05$), i.e., with OF.04 (Organizational Factor – Project Planning). This actually makes sense since project planning and project schedule are two closely-related entities and, in general, a more careful planning would lead to a more controlled schedule and project’s on-time delivery. We see that empirical data is confirming the generally believed expectations in this regard.

In Table 2, we have also calculated the average values of correlations for each row and each column. These average values show, for the case of each CSF (row) and project success variable (column), how strongly it is correlated with the factors in the other group. The highest value of this average correlation metric for a CSF could be 1.0 which would denote that it has strongest correlation (rho value = 1.0) with “all” success factors. Thus, the average correlation metric is a suitable one to rank the CSFs. We will use these average correlation values when answering RQ 2 (Ranking of CSFs) in Section 5.2.

Almost all team factors (TF.01 … TF.07) are correlated with all variables describing satisfaction of various stakeholders. Interestingly, among all CSFs, only project monitoring and controlling (OF.05) is significantly and positively correlated with all variables describing project success in product characteristics and stakeholder satisfaction categories. We identified the five most-correlated pairs in Table 2 as follows:

- OF.05 (Monitoring and controlling) and SF.03 (Top management satisfaction), with rho = 0.54. This denotes that better project monitoring and controlling usually is associated with higher top-management satisfaction.
- TF.07 (Team’s experience with the development methodologies) and ProdF.08 (Compatibility), with rho = 0.52, which denotes that the higher the team’s experience with the development methodologies, the higher the ability of the delivered software to exchange information and/or to perform their required functions with other software systems/environments, which is also expected.

- OF.04 (Project planning) and SF.03 (Top management satisfaction), with rho = 0.49, which denotes that better project planning is closely related to top management satisfaction.
- TF.02 (Internal team communication) and ProcF.04 (Team building and team dynamics), with rho = 0.48, which denotes that the higher the quality of internal team communication, the higher the team building and team dynamics at the end of the project.
- OF.06 (Organization-wide change management practices) and SF.03 (Top management satisfaction), with rho = 0.47, which denotes that more mature organization-wide change management practices is closely related to top management satisfaction.

5.1.2 Analysis using the balanced dataset

We also wanted to know the extent to which the correlations among proposed CSFs and project success would differ, if we did the analysis on the balanced dataset instead of the original dataset, since the project success factors are skewed towards more successful projects in the original dataset, as discussed in Sections 4.2.3 and 4.3. Table 4 shows the correlation values and has the same layout as Table 2 (for the original dataset). However, different to our analysis results shown in Table 2, in Table 4 we indicate whether correlations are significant even after applying Bonferroni correction.

We looked for strong correlations in Table 4, and we identified the nine correlation coefficients (rho values) above 0.6 and show them as red bold values. Even with Bonferroni correction, these nine correlations were significant at level 0.05.[2] The strong correlations are between the following CSF and project success metrics:

1. OF.04 (Project planning) vs. ProcF.04 (Team building and team dynamics) (with rho = 0.65)
2. OF.05 (Project monitoring / control) vs. ProcF.04 (Team building and team dynamics) (with rho = 0.68)
3. OF.05 (Project monitoring / control) vs. ProdF.03 (Functional suitability) (with rho = 0.60)
4. OF.05 (Project monitoring / control) vs. SF.02 (Team satisfaction) (with rho = 0.62)
5. OF.05 (Project monitoring / control) vs. SF.03 (Top management satisfaction) (with rho = 0.61)
6. OF.05 (Project monitoring / control) vs. SF.02 (Team satisfaction) (with rho = 0.62)
7. TF.01 (Team commitment) vs. ProcF.04 (Team building and team dynamics) (with rho = 0.61)
8. TF.06 (Expertise in task and domain) vs. ProdF.01 (Overall quality) (with rho = 0.67)
9. TF.06 (Expertise in task and domain) vs. ProdF.03 (Functional suitability) (with rho = 0.62)

When comparing the above listed pairs of entries and their correlations with the corresponding correlations in Table 2, it turns out that none of them had a correlation of 0.6 or higher. This indicates that using the balanced dataset yields stronger (and significant) top correlations. This seems to imply that, when studying a population of projects which are “balanced” in terms of success metrics, one could expect to observe stronger top correlations among CSFs and project success metrics.

[2] We also applied Holm’s Sequential Bonferroni Procedure and the result was the same.

Table 4 Correlations (Spearman's rho) between proposed CSFs (rows) and project success metrics (columns) - balanced dataset

		Process factors (ProcF)				Product-quality factors (ProdF)						
		ProcF.01	ProcF.02	ProcF.03	ProcF.04	ProdF.01	ProdF.02	ProdF.03	ProdF.04	ProdF.05	ProdF.06	ProdF.07
Organizational factors	OF.01	−0.20	0.18	−0.13	0.41**	0.19	0.10	0.21	0.35*	0.47**	0.12	0.45**
		W	W	W	M	W	W	W	M	M	W	M
	OF.02	−0.10	0.06	−0.01	−0.09	0.08	−0.22	−0.05	0.13	−0.12	0.10	−0.17
		W	N	N	N	N	W	N	W	W	W	W
	OF.03	−0.18	0.03	0.03	0.47**	0.23	0.36*	0.23	0.42*	0.27	0.19	0.27
		W	N	N	M	W	M	W	M	W	W	W
	OF.04	0.56**	0.36*	0.34*	0.65	0.41**	0.40*	0.37*	0.38*	0.31	0.38*	0.06
		S	M	M	S	M	M	M	M	M	M	N
	OF.05	0.27	0.28	0.32*	0.68	0.52**	0.51**	0.60**	0.49**	0.49**	0.45**	0.18
		W	W	M	S	S	S	S	M	M	M	W
	OF.06	0.25	0.09	0.39*	0.62**	0.48**	0.53**	0.54**	0.51**	0.31	0.47**	0.20
		W	N	M	S	M	S	S	S	M	M	W
Team factors	TF.01	−0.16	0.00	−0.10	0.61	0.22	0.13	0.26	0.35*	0.35*	0.33*	0.21
		W	N	N	S	W	W	W	M	M	M	W
	TF.02	−0.02	−0.01	−0.07	0.53**	0.27	−0.01	0.10	0.34*	0.25	0.28	0.20
		N	N	N	S	W	N	N	M	W	W	W
	TF.03	0.05	0.37*	−0.01	0.29	0.26	0.08	0.24	0.14	0.38*	0.23	0.11
		N	M	N	W	W	N	W	W	M	W	W
	TF.04	0.10	0.14	0.05	0.54**	0.21	0.13	0.10	0.24	0.41*	0.32	0.21
		W	W	N	S	W	W	W	W	M	M	W
	TF.05	0.41*	0.21	0.28	0.28	0.44**	0.25	0.31	0.21	0.33*	0.30	0.05
		M	W	W	W	M	W	M	W	M	W	N
	TF.06	0.22	0.16	0.49**	0.45**	0.67	0.43*	0.62	0.38*	0.47**	0.35*	0.16
		W	W	M	M	S	M	S	M	M	M	W
	TF.07	−0.09	0.06	0.07	0.34*	0.25	0.50**	0.31	0.18	0.23	0.09	0.21
		N	N	N	M	W	M	M	W	W	N	W
Customer factorsddsssd	CF.01	0.11	0.02	−0.02	0.19	0.20	0.26	0.25	0.34*	0.15	0.04	0.07
		W	N	N	W	W	W	W	M	W	N	N
	CF.02	0.19	0.08	0.05	0.26	0.20	−0.02	0.00	0.29	0.14	−0.01	0.02
		W	N	N	W	W	N	N	W	W	N	N
	CF.03	0.10	0.08	−0.05	0.26	0.16	0.11	0.13	0.25	0.39*	0.13	0.01

Table 4 (continued)

	Process factors (ProcF)				Product-quality factors (ProdF)						
	ProcF.01	ProcF.02	ProcF.03	ProcF.04	ProdF.01	ProdF.02	ProdF.03	ProdF.04	ProdF.05	ProdF.06	ProdF.07
	N	N	N	W	W	W	W	W	M	W	N
CF.04	−0.11	−0.05	0.17	0.19	0.26	0.39*	0.34*	0.37*	0.02	0.20	0.17
	W	N	W	W	W	M	M	M	N	W	W
PF.01	0.00	−0.05	−0.32	−0.16	−0.19	−0.32	−0.28	−0.16	−0.04	−0.16	0.08
	N	N	M	W	W	M	W	W	N	W	N
PF.02	0.08	0.02	0.21	−0.09	0.11	0.35*	0.17	0.20	−0.10	0.05	0.13
	N	N	W	N	W	M	W	W	N	N	W
PF.03	0.00	0.20	−0.03	−0.06	−0.26	0.07	−0.11	−0.01	0.02	−0.14	0.19
	N	W	N	N	W	N	W	N	N	W	W
PF.04	0.37	0.26	0.20	0.27	0.03	0.36*	0.18	0.14	0.07	−0.01	0.18
	M	W	W	W	N	M	W	W	N	N	W
PF.05	0.11	0.10	−0.06	0.14	−0.12	−0.22	−0.14	−0.04	0.01	−0.04	0.11
	W	N	N	W	W	W	W	N	N	N	W
PF.06	0.22	0.16	0.27	0.21	0.14	0.14	0.13	0.35*	0.18	0.09	0.20
	W	W	W	W	W	W	W	M	W	N	W
PF.07	−0.26	−0.24	−0.03	0.04	0.14	−0.18	0.01	0.24	0.26	0.12	0.44**
	W	W	N	N	W	W	N	W	W	W	M
None	7	13	13	4	2	4	4	2	6	7	6
Weak	14	9	6	10	17	10	13	11	8	11	16
Moderate	2	2	5	4	3	8	4	10	10	6	2
Strong	1	0	0	6	2	2	3	1	0	0	0
Avg.	0.08	0.10	0.08	0.29	0.20	0.17	0.19	0.25	0.22	0.16	0.15
Rank	15	14	15	1	6	10	7	3	4	12	13

	Product-quality factors (ProdF)			Satisfaction factors (SF)			# of corr. Classifications				
	ProdF.08	ProdF.09	ProdF.10	SF.01	SF.02	SF.03	N	W	M	S	Avg.
Organizational factors	0.10 W	0.36* M	0.21 W	0.18 W	0.32 M	0.08 N	1	10	6	0	0.20

Table 4 (continued)

	Product-quality factors (ProdF)			Satisfaction factors (SF)			# of corr. Classifications				
	ProdF.08	ProdF.09	ProdF.10	SF.01	SF.02	SF.03	N	W	M	S	Avg.
	0.24	0.12	0.13	0.07	0.05	−0.34*					
	W	W	W	N	N	M	7	9	1	0	−0.01
	0.32*	0.31	0.10	0.33*	0.43**	0.23					
	M	M	N	M	M	W	3	7	7	0	0.24
	0.58**	0.26	0.07	0.41**	0.51**	0.59**					
	S	W	N	M	S	S	2	1	9	5	0.39
	0.41*	0.40*	0.14	0.50**	0.62	0.61**					
	M	M	W	S	S	S	0	4	6	7	0.44
	0.13	0.32	0.12	0.46**	0.53**	0.45**					
	W	M	W	M	S	M	1	4	7	5	0.38
Team factors	0.23	0.31	0.10	0.43**	0.52**	0.08					
	W	M	W	M	S	N	3	7	5	2	0.23
	0.35*	0.35*	−0.02	0.31	0.47**	−0.05					
	M	M	N	M	M	N	7	4	5	1	0.19
	0.33*	0.28	0.14	0.26	0.29	0.14					
	M	W	W	W	W	W	3	11	3	0	0.21
	0.46**	0.14	0.14	0.31	0.49**	0.10					
	M	W	W	M	M	N	2	9	5	1	0.24
	0.25	0.10	−0.10	0.43**	0.20	0.44**					
	W	N	W	M	W	M	2	9	6	0	0.26
	0.13	0.33*	0.08	0.57**	0.44**	0.35*					
	W	M	N	S	M	M	1	4	9	3	0.37
	0.39*	0.16	0.17	0.32*	0.09	0.34*					
	M	W	W	M	N	M	5	6	6	0	0.21
Customer factorsddsssd	0.08	0.16	0.04	0.30	0.41**	0.20					
	N	W	N	W	M	W	6	9	2	0	0.16
	0.14	0.04	−0.07	0.29	0.40*	0.17					
	W	N	N	W	M	W	8	8	1	0	0.13
	0.23	0.23	0.18	0.20	0.47**	0.11					
	W	W	W	W	M	W	4	11	2	0	0.18
	0.21	0.34*	0.07	0.40*	0.34*	0.15					

Table 4 (continued)

	Product-quality factors (ProdF)			Satisfaction factors (SF)			# of corr. Classifications				
	ProdF.08	ProdF.09	ProdF.10	SF.01	SF.02	SF.03	N	W	M	S	Avg.
	W	M	N	M	M	W	3	8	6	0	0.20
	0.01	0.03	−0.01	−0.24	−0.30	−0.29					
	N	N	N	W	M	W	7	7	3	0	−0.14
	0.14	0.06	−0.40*	0.02	−0.02	0.22					
	W	N	M	N	N	W	8	7	2	0	0.07
	−0.11	−0.22	−0.13	−0.16	−0.12	0.17					
	W	W	W	W	W	W	6	11	0	0	−0.04
	0.15	−0.03	−0.36*	0.09	0.17	0.42*					
	W	N	M	N	W	M	5	8	4	0	0.15
	0.03	−0.24	−0.22	−0.12	−0.09	0.12					
	N	W	W	W	N	W	7	10	0	0	−0.04
	0.01	0.04	−0.23	0.10	0.22	0.39*					
	N	N	W	N	W	M	4	11	2	0	0.15
	−0.34*	0.30	0.15	−0.06	0.02	−0.17					
	M	M	W	N	N	W	5	9	3	0	0.03
	4	6	8	5	5	4					
	12	9	14	8	5	11					
	7	9	2	9	10	7					
	1	0	0	2	4	2					
	0.19	0.17	0.01	0.22	0.27	0.19					
	7	10	17	4	2	7					

5.1.3 Cross-correlations among the CSFs

Another issue that we looked into was the presence of cross-correlations among the CSFs themselves. For this purpose, we calculated the correlation coefficient of each pair of the 24 CSFs, resulting in 276 correlation values. Similar to the previous analysis, we calculate the cross-correlations among the CSFs for both the original and the balanced dataset.

Tables 5 and 6 show the breakdowns of correlation coefficients, for the two datasets, based on the classification of 'None', 'Weak', 'Moderate' or 'Strong' correlation values. Due to the large number of CSFs and their combinations, we show the correlation values among all CSFs, for the two datasets, in Appendix Table 12. As we can see in the tables in Appendix Table 12, as expected, many of the CSFs have cross-correlations with the other CSFs in both cases.

In the original dataset, about 66% (=49% + 16% + 1.5%) of the CSF pairs showed at least weak cross-correlations. We observe higher correlation values often among CSFs in the same CSF category, e.g., organizational factors, or team factors. For example, TF.01 (Team commitment) and TF.02 (Internal team communication) have a moderate correlation of 0.49, denoting that, as expected, when one is higher, the other would be higher too. Since weak to moderate internal cross-correlations among many CSFs are expected and normal, we do not believe these cross-correlations could have any negative impacts on our analysis nor on the results in any way. Actually, on the other hand, these cross-correlations enabled us to check the plausibility of answers by the respondents ("sanity check"), i.e., whether responses to CSFs, which we expected to be correlated, actually are correlated (e.g., see the above example).

Similarly, for the case of the balanced dataset, about 74% (=49% + 20% + 5%) of the CSF pairs showed at least weak cross-correlations. Thus, both cases confirmed the plausibility of answers by the respondents ("sanity check").

5.2 RQ 2: Ranking of CSFs based on their correlations with project success metrics

To better understand whether some of the proposed CSFs are associated with variables describing project success stronger than others, we ranked the proposed CSFs based on their average correlations as calculated in the Section 5.1.1 for the original dataset and showed in Table 2. For easier view of the factors' average correlations, we visualize the average values of Table 2 in Fig. 14. Since we felt that the weak correlations value could have had a noise effect on the regular average values, we also calculated "weighted" average values and show them in Fig. 14 too. For the weighted average, we used the following weight values: 1: for None or very weak (N) correlations, 2: for weak (W) correlations, 3: for moderate (M) correlations, and 4: for strong (S) correlations.

Table 5 Classification of strength of cross-correlations among CSFs – original dataset

Correlation strength	Correlation Range (absolute value)	# of pairs	% of all pairs ($n = 276$ pairs)
None	[0.0–0.1]	93	33.7%
Weak	[0.1–0.3]	135	48.9%
Moderate	[0.3–0.5]	44	15.9%
Strong	[0.5–1.0]	4	1.4%

Table 6 Classification of strength of cross-correlations among CSFs – balanced dataset

Correlation strength	Correlation Range (absolute value)	# of pairs	% of all pairs (n = 276 pairs)
None	[0.0–0.1]	72	26.1%
Weak	[0.1–0.3]	137	49.6%
Moderate	[0.3–0.5]	54	19.6%
Strong	[0.5–1.0]	13	4.7%

We can see in Fig. 14 that, according to average values, the top-three CSFs having the highest average correlations are: (1) OF.05 (Project monitoring and controlling), (2) OF.04 (Project planning), and (3) TF.07 (Team's experience with the development methodologies). The top-three CSFs w.r.t. average and weighted average values almost overlap, differing in one factor only. According to weighted average values, the top-three CSF list changes slightly. OF.05 is still #1. However, the 2nd and 3rd ranked CSF are TF.07 and TF.06 (Expertise in task and domain).

We also compare in Table 7 our ranking of proposed CSFs with the ranking of CSFs as reported in Ahimbisibwe et al. (Ahimbisibwe et al. 2015). The definitions of our CSFs correspond largely to those in (Ahimbisibwe et al. 2015) (as discussed in Section 2.3). We have, however, only 24 CSFs as compared to the 27 CSFs in (Ahimbisibwe et al. 2015), since as discussed in Section 2.3 we "merged" a few similar CSFs of Ahimbisibwe et al. (Ahimbisibwe et al. 2015). The rankings of CSFs in (Ahimbisibwe et al. 2015) were based on frequency count, i.e., how often a CSF is mentioned in the surveyed literature. Instead of asking our survey participants what CSFs they propose and their importance which could provide subjective and imprecise results from their side, we provided them with the set of factors and asked participants to what degree the factors correlated with success of projects.

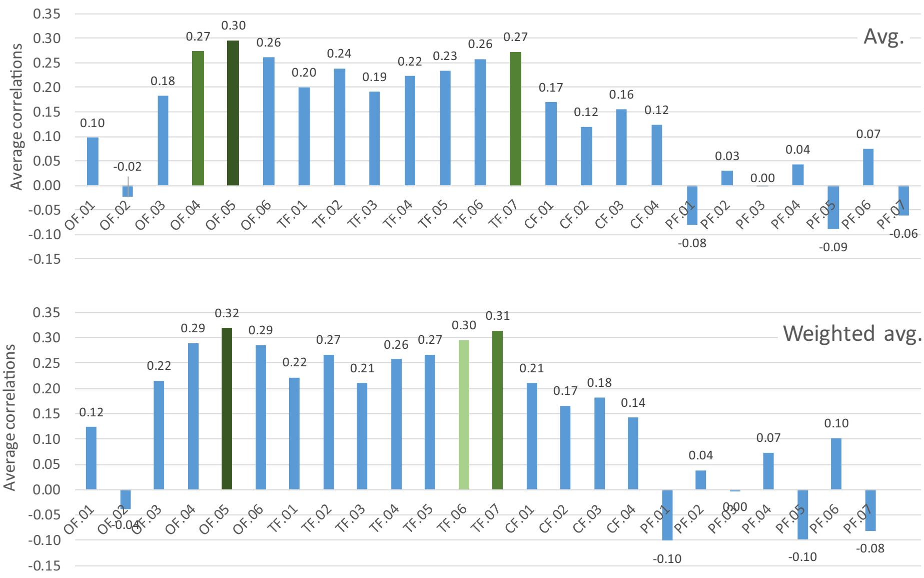


Fig. 14 Average correlations of independent factors – original dataset, based on normalized average and weighted average values

Table 7 Ranking of proposed CSFs (from original dataset) and comparison with ranking in Table IV in (Ahimbisibwe et al. 2015)

CSFs			Rank in our study	Rank in [3]
Category	Code	Explanation		
Organizational Factors	OF.01	Top-level management support	16	1
	OF.02	Organizational culture	21	4
	–	Leadership[a]	(21)	6
	OF.03	Vision and mission	11	7
	OF.04	Level of project planning	2	5
	OF.05	Monitoring and controlling	1	8
	OF.06	Change management skills	4	9
Team Factors	TF.01	Project team commitment	9	3
	TF.02	Internal project communication	6	10
	TF.03	Project team empowerment	10	15
	TF.04	Project team's composition	8	16
	TF.05	Project team's general expertise	7	20
	TF.06	Project team's expertise with the task	5	19
	–	Lack of development team skill[b]	(5)	21
	TF.07	Project team's experience with SDM	3	25
Customer Factors	CF.01	User participation	12	2
	CF.02	User support	15	11
	CF.03	Customer training and education	13	17
	CF.04	Customer (client) experience	14	18
	–	Lack of end user experience[c]	(14)	27
Project Factors	PF.01	Technological uncertainty	23	12
	PF.02	Project complexity	19	14
	PF.03	Urgency	20	22
	PF.04	Relative project size	18	23
	PF.05	Specifications changes	24	24
	PF.06	Project criticality	17	26
	PF.07	Development methodologies	22	13

[a] The CSF 'Leadership' in (Ahimbisibwe et al. 2015) is subsumed in our proposed factor OF.02 (Organizational culture and management style). Therefore, we assume that the same rank number applies (i.e., rank = 21).

[b] The CSF 'Lack of development team skill' in (Ahimbisibwe et al. 2015) is subsumed in our proposed factor TF.06 (Project team's expertise in the task and domain). Therefore, we assume that the same rank number applies (i.e., rank = 2).

[c] The CSF 'Lack of end user experience' in (Ahimbisibwe et al. 2015) is subsumed in our proposed factor CF.04 (Customer (client) experience in their own business domain). Therefore, we assume that the same rank number applies (i.e., rank = 20).

The differences in the two ranking methods have resulted in two quite different ranking results. Our way of ranking the proposed CSFs can be considered as complementing the way of ranking proposed in (Ahimbisibwe et al. 2015).

We visualize in Fig. 15 the CSF rankings in our study versus those in (Ahimbisibwe et al. 2015). We see that the two ranks are quite different and we cannot notice a visual similarity. We wanted to see if a CSF is high in one ranking, is it necessarily ranked high in the other rank too? A correlation analysis using Pearson's correlation coefficient did not show any significant correlation between the two ranks ($r = 0.17$). Despite the lack of significant overall correlation between the rankings, it is visible from the scatter plot shown in Fig. 15 that some factors situated in the quadrant bisector of the scatter plot have similar rankings in our study and Ahimbisibwe et al. (Ahimbisibwe et al. 2015). For example, OF.04, OF.05 and OF.06 were all ranked on top 10 in both ranks, while PF.03, PF.04, and PF.05 were ranked in the last 10 in both ranks.

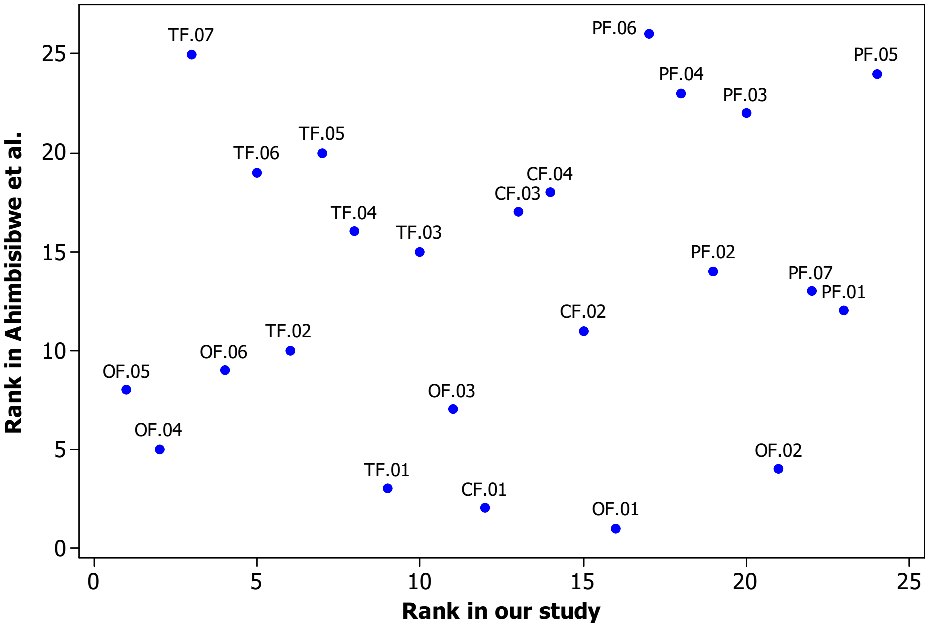


Fig. 15 Visualization of rankings of CSFs in our study (with original dataset) versus the ranking in (Ahimbisibwe et al. 2015)

5.3 RQ 3: Ranking of project success metrics based on strength of association with CSFs

To better understand whether some of the project success metrics are more strongly associated with the set of proposed CSFs than others, we calculated the average of correlation values for each project success metrics for both the datasets, i.e., average of all values for each column in Tables 2 and 4.

Table 8 shows the average values and the rankings using both datasets. Using the original dataset, the top three project success metrics, having the largest average associations with proposed CSFs, are: (1) SF.02 (Team Satisfaction), (2) ProcF.04 (Team building and team dynamics), and SF.03 (Top management satisfaction). Using the balanced dataset, the top three project success metrics are: (1) ProcF.04, (2) SF.02, and (3) ProdF.04 (Reliability).

It is interesting that the two ranks are quite similar. To cross compare the two rankings, we visualize the two rankings as a scatter plot in Fig. 16, which depicts the similarity of the two rankings. Two metrics are outside the general trend in this scatter plot, i.e., SF.01 (User satisfaction) and ProdF.05 (Performance efficiency). SF.01 is ranked 12th in one dataset while ranked 4th in the other one. ProdF.05 is ranked 13th in one dataset while ranked 4th in the other one. We could not find any interpretations for these observations in our data.

6 Discussions and implications of the findings

In this section, we discuss the summary of our findings, lessons learned, and potential threats to the validity of our study.

Table 8 Ranking of variables describing project success

Project success category	Variable describing project success		Using the original dataset		Using the balanced dataset	
			Avg. of correlations	Rank	Avg. of correlations	Rank
Process	ProcF.01	Budget	0.01	17	0.08	15
	ProcF.02	Schedule	0.02	16	0.1	14
	ProcF.03	Scope	0.06	14	0.08	15
	ProcF.04	Team building and team dynamics	0.21	2	0.29	1
Product	ProdF.01	Overall quality	0.19	5	0.2	6
	ProdF.02	Business and revenue generation	0.15	9	0.17	10
	ProdF.03	Functional suitability	0.16	6	0.19	7
	ProdF.04	Reliability	0.19	4	0.25	3
	ProdF.05	Performance efficiency	0.11	13	0.22	4
	ProdF.06	Usability	0.12	11	0.16	12
	ProdF.07	Security	0.16	7	0.15	13
	ProdF.08	Compatibility	0.16	8	0.19	7
	ProdF.09	Maintainability	0.13	10	0.17	10
	ProdF.10	Transferability	0.05	15	0.01	17
Satisfaction of stakeholders	SF.01	User satisfaction	0.11	12	0.22	4
	SF.02	Team satisfaction	0.22	1	0.27	2
	SF.03	Top management satisfaction	0.20	3	0.19	7

6.1 Summary of findings and implications

Summary of findings regarding RQ 1 is as follows. Note that all these observations and results elaborated in Section 5 are based on certain characteristics (profile) of the dataset, e.g., as discussed in Section 4.2, our dataset mostly includes small to medium projects, concerning mostly new development projects, with low to medium technological uncertainty, and moderate to high technological complexity. Thus, all our observations are only specific to the specific dataset and the generalization of our findings should be done with care.

- The five most-correlated pairs of CSF and project success metrics (from the original dataset) were (Table 2): (1) OF.05 (Monitoring and controlling) and SF.03 (Top management satisfaction); (2) TF.07 (Team's experience with the development methodologies) and ProdF.08 (Compatibility); (3) OF.04 (Project planning) and SF.03 (Top management satisfaction); (4) TF.02 (Internal team communication) and ProcF.04 (Team building and team dynamics); and (5) OF.06 (Organization-wide change management practices) and SF.03 (Top management satisfaction). These findings clearly denote our suggested priority of issues, needing attention by project managers if they want to increase their chances of project success.
- Level of specification (requirements) changes (i.e., volatility) was not correlated with any of the project success metrics. This would require further research to find out whether project success is not related to the level of specification changes or this is a case specific to our sample population.
- Schedule as a variable describing projects success is related to only one CSF that is project planning. Level of project planning in projects being a factor for schedule variance might be an anticipated outcome.

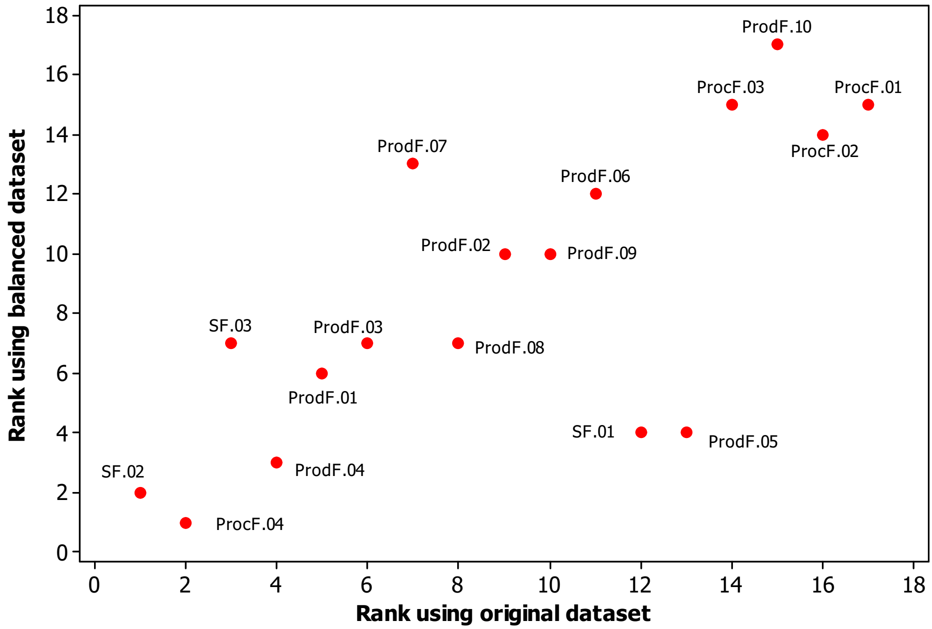


Fig. 16 Visualization of ranking of project success metrics based on strength of association with CSFs using the original dataset versus the balanced dataset

- Interestingly, among all CSFs, only project monitoring and controlling is significantly and positively correlated with all variables describing project success in product characteristics and stakeholder satisfaction categories.
- In general, CSFs in organizational and team factors categories have more moderate to strong correlations with the variables describing project success than the ones in customer and project CSF categories.
- In the balanced dataset, top 6 strong correlations between CSFs and project success metrics were: (1) OF.04 (Project planning) and ProcF.04 (Team building and team dynamics); (2) OF.05 (Project monitoring/control) and ProcF.04 (Team building and team dynamics); (3) OF.05 (Project monitoring/control) and SF.02 (Team satisfaction); (4) TF.01 (Team commitment) and ProcF.04 (Team building and team dynamics); (5) TF.06 (Expertise in task and domain) and ProdF.01 (Overall quality); and (6) TF.06 (Expertise in task and domain) and ProdF.03 (Functional suitability).
- In both the original dataset and the balanced dataset, we observed higher correlation values often among CSFs in the same CSF category, e.g., organizational factors, or team factors.

Summary of the findings about RQ 2 are as follows:

- The top three CSFs, having the largest count of significant associations with variables describing project success, were: (1) OF.05 (Project monitoring and controlling), (2) OF.04 (Project planning), and (3) TF.07 (Team's experience with the development methodologies). The implications of this result for software managers is that, when resources are limited, they should prioritize the improvement opportunities in their organizations based on these highly factors to ensure success chances. We believe that our findings are similar to the following observation reported in (Reel 1999): "*At least seven of 10 signs of IS project failures are determined before a design is developed or a line of code is written*", since OF.05 (Project monitoring and controlling), OF.04 (Project planning) were the top CSFs and were ranked even higher than the 3rd CSF: TF.07 (Team's experience with the development methodologies). Thus, while technical abilities of software engineers are important, project management seems to be even more important. We believe having empirical evidence for such an issue, as provided by our study, is an important take-away for software managers.

- When we compared our ranking of CSFs with the ranking of CSFs mentioned in the academic literature, we could not observe many obvious similarities. Among the groups of CSFs, only customer factors in our study are correlated significantly with their counterparts in (Ahimbisibwe et al. 2015), as a group. This might be an indication that the most popular CSFs in academic publications might not be the ones that are most important for project success in software industry in practice, since our data were based on direct opining of practitioners.

Regarding RQ 3, we summarize the findings as follows:

- The top three project success metrics, having the largest count of significant associations with proposed CSFs, were variables: (1) SF.02 (Team Satisfaction), (2) ProcF.04 (Team building and team dynamics), and SF.03 (Top management satisfaction). Using the balanced dataset, the top three project success metrics are: (1) ProcF.04, (2) SF.02, and (3) ProdF.04 (Reliability). From a review of the literature, we see a relatively low attention by the research community on "team satisfaction" as an important project success metric. Thus, our study highlights the need for further attention to this important metric. However, we are seeing that the Agile community is putting further emphasis on team satisfaction and morale, e.g. (Tessem and Maurer 2007).

In addition to the findings and implications summarized above, practitioners can benefit from our results as follows:

- General empirical findings of our study indicate that project team's experience with SDM (TF.07), project team's expertise with the task (TF.06), and monitoring and controlling (OF.05) are the top CSFs for software projects. It might be a good practice for the practitioners to start with these CSFs. It is interesting to see that, for example, project team's experience with SDM (TF.07) has a rank of 25 in Ahimbisibwe et al. (Ahimbisibwe et al. 2015)'s study which is based on a literature review. It is also interesting to see that the same CSF (TF.07) is ranked almost top in our study in both groups that apply pure-agile and pure plan-based methodologies.
- Ahimbisibwe et al. (Ahimbisibwe et al. 2015)'s study, as the most extensive and granular one, had ranked the list of CSFs based on reported studies in scientific literature. Our empirical findings indicate that the rankings of the CSFs are different in practice. Still, the practitioners might pay attention to the CSFs that are ranked at the top in Ahimbisibwe et al. (Ahimbisibwe et al. 2015)'s study and this study. Figure 15 in Sec 0, for example, shows the relation between the rankings of the CSFs in both studies. When we check the CSFs ranked in top ten in both studies, we see that vision and mission (OF.03), project planning (OF.04), OF.05 (project monitoring and controlling), organization-wide change

management practices (OF.06), and internal team communication (TF.02) are the high-ranked factors. These might be good points for the practitioners to direct improvement efforts.

- The ranking of project success metrics in our study might also give an idea while planning for the improvements. Table 8 in Section 5.3 shows such a ranking, and according to the table; team satisfaction (SF.02), top management satisfaction (SF.03), team building and team dynamics (ProcF.04), reliability (ProdF.04), and functional suitability (ProdF.03) have strong associations with the CSFs. Accordingly, the practitioners can consider team-related issues and basic product attributes such as reliability and functional suitability to plan for and assess their project success.
- Although various studies in the literature reported a number of CSFs for software projects, the factors might change in accordance with organizational, process, and project dynamics. Our empirical study clearly shows that CSFs differ as company size, project size, and development methodology change. The practitioners should be aware of this situation, and carefully select the factors for their projects.

As mentioned in Section 5.1.1, it has been widely discussed in the research community that a high correlation does not necessarily mean causality (Hatfield et al. 2006). At the same time, however many in the community believe that: "*Correlation may not imply causation, but it sure can help us insinuate it*" (Chandrasekaran 2011). Thus, we only discussed the existence (or lack) of correlations among various project factors and project success in this paper, and did not intend to imply causality, i.e., if the development methodology is plan-driven, we cannot necessarily say that the stakeholder satisfaction will be low. In fact, one CSF alone cannot guarantee project success and there could be many other confounding factors in play in such a complex cause-effect relationship. One take-away message from this paper is that, by focusing on and improving all CSFs, a project team can only increase the chances of project success.

Last but not the least, we would like to mention the work of Wieringa and Daneva (Wieringa and Daneva 2015) in which they discussed strategies for generalizing software engineering theories. They mentioned that "*Generalization should be based on architectural similarity, and generalizations can be made more robust by analytical induction*". "*Analytical induction consists of a series of case studies, where all cases have a similar architecture, but also differ from each other*". Thus, we believe that, by using analytical induction, CSFs in software projects can be studied more systematically and rigorously. Further additional empirical studies, such as the one we report in this paper and also those in the past, e.g., (Stankovic et al. 2013; Chow and Cao 2008), can lay the foundation of future theories about CSFs in software projects.

6.2 Limitations and potential threats to validity

We discuss the potential threats to the validity of our study and steps taken to minimize them. We consider four types of validity threats (conclusion, internal, construct, and external) based on a standard checklist adopted from (Wohlin et al. 2012).

6.2.1 Internal validity

A major potential threat to internal validity of our study lies in the selection of the sample population. Our specific concern here is that whether our selected group of projects is representative for the whole population of software projects. Due to our limited resources, it was not feasible for us to reach out various software projects from every city and company in Turkey, which made probabilistic sampling not a possible option. After all, our main sampling method was convenience sampling, where respondents are selected based on their convenient accessibility to the researchers.

As discussed in Section 3.3, our publicity campaign was executed with respect to a pre-established publicity and execution plan. Our publicity and execution plan included a network of contacts, who were deliberately selected from various application domains and personal backgrounds with an intention to mitigate this validity threat. We also intended to inject some randomness in our sample population by asking our invited contacts to disseminate our invitation with their own contacts, posting messages in software engineering related groups in online social networking sites, and requesting Technology Development Zones to forward our invitation.

As a pitfall of such a survey execution, the number of organizations represented with at least one project in our study is unknown. Yet, not to risk data validity, we did not intent to collect any revealing information about organizations in this study.

As reported in Section 4.1, we ended up with a reasonably good mix of software projects in our data set in terms of variety in company and participant profiles. It is quite fair to say that our sample is reasonably representative for the software projects in Turkey, in terms of distribution of projects from different company sizes and target sectors. When compared with our another Turkey-wide study (Garousi et al. 2015b), it is also fair to say that our data about projects were gathered from participants having various experience levels and assuming different positions that is quite representative for Turkish software engineering community. According to a report in 2012 (Akkaya et al. 2012), 47 and 33% of the software companies are located in Istanbul and Ankara, respectively. Albeit the fact that all the efforts were made to minimize the selection bias, however, it is possible that software projects from all Turkish cities were not properly represented. Due to the location of most authors (i.e. Ankara) and the fact that most of our contacts were local, about 71% of the software project data were gathered from participants located in Ankara.

An ideal study in this context should include, as much as possible, a triangulation of different kinds of evidence and data about projects. For example, qualitative interviews, if not actual measures from the projects (which are often difficult to obtain) should be used for further understanding and complimenting the data from the opinion survey. We admit this weakness and due to the shortage of research resources, we were unable to conduct qualitative interviews.

6.2.2 Construct validity

A threat to construct validity is related to whether we actually measured the perception of success factors in software projects. We basically counted the votes for each question that was designed based on a contingency fit model of CSFs. As commonly done in other survey studies, e.g., (Stankovic et al. 2013; Chow and Cao 2008), we believe that results based on such voting data and statistical inferences made would reflect the perceptions of respondents about the values of CSFs and variables describing project success in their projects.

A phenomenon known as hypothesis guessing (Wohlin et al. 2012), where participants try to figure out the goal of the study and base their responses on their hypothesis, represents another threat to construct validity. In order to minimize this threat, we described our intention in this survey as to measure the perceptions of projects characteristics in software engineering, analyze the collected data in an aggregate form, and publish the findings as a research article. We also informed the participants that we will not collect any identifying information so all participants will remain anonymous.

6.2.3 Conclusion validity

A threat to conclusion validity lies in the reliability of measures that depend on factors like poor question wording and survey layout (Wohlin et al. 2012). We, as authors, reviewed the survey questions and also asked eight of our personal contacts from the software industry to respond to the questions in pilot study and provide feedbacks regarding the quality of our survey questions.

Another threat to conclusion validity in our study is related to whether the conclusions we drew were based on rigorous and repeatable treatment (Wohlin et al. 2012). In order to reduce the bias in reaching conclusions for each research question, we relied on statistical analysis results. Thus, interpretation of the findings and implications of our research depends on statistical significance and are strictly traceable to data. Here, we should caution the reader of the problem of multiple testing (Howell 2013). In our correlation analysis, when setting $p = 0.05$, we must expect 19 false positives (i.e., correlations where in reality there is none). However, this number reduces to 4 when setting $p = 0.01$.

Another issue is concerning possible biased conclusion due to the skewed dataset in terms of respondents' positions and project profiles (Sections 4.1 and 4.2). We saw that most of the respondents were software developers (programmers). We had a few software designers, software architects, and projects managers in the participants as well, but we only had 2 data points form higher-level management (CEOs). It could be that opinions of higher-level management could be different than technical staff (e.g., software developers). We have observed that it is in general challenging to get CEOs to fill out surveys, given their busy schedule. Thus, this issue stays as a limitation and future studies may focus on it.

Another threat to conclusion validity is that our conclusions might be biased by the background of the participants. As an example, let us consider the following case: Would it be possible that a developer who considered a project successful where the team was more satisfied during the project rather than based on customer satisfaction? After all, how does one know how much a developer participating in the survey knows about the customer satisfaction? A developer's specific goal might have been to have a functional team and positive working environment, and this could have influenced his/her view of the other success factors. In the end, this would not mean that the project was indeed successful in general (might have been successful for that specific person for his/her specific goal).

Also, we should explicitly state that our results were based on a dataset including mostly small to medium projects, concerning mostly new development projects, with low to medium technological uncertainty, and moderate to high technological complexity. All these considerations should be considered when interpreting the results and conclusions.

6.2.4 External validity

External validity of a study deals with the limitations of a research to generalize the results beyond the scope of the participants in our study. Our concern in this study is whether our findings can be generalized to Turkish software industry. We believe that we involved a large enough sample of Turkish companies to assume a certain degree of representativeness for the Turkish software industry, i.e., interviewees in Turkish companies that are of similar size and working in the same business sector as those included in our sample might well have given answers similar to those we received in our study. We acknowledge that some software projects in Turkey were under-represented in our data set, e.g. projects from cities other than the capital region Ankara where most of the authors are located in. Also, we acknowledge that most respondents gave answers leaning to the positive side with regard to project success. This lack of balance between perceived success is, however, not extreme. Also, we do not know what the actual success rate (or even the actual perceived success rates) in all Turkish companies are to judge whether the answers given by our respondents is representative or due to bias. We would like to point out, however, that we (knowingly) gave no signal to the respondents that might make them believe we were asking for characteristics of successful projects only. Also, our decision to also build a "balanced" sub-set of the original dataset was another action that we took to minimize this potential threat.

Generalizing our findings to the software industry worldwide requires additional care as certain regulations that apply to Turkish industry might not apply abroad. We do not have any intention to generalize our findings to other countries. However, demographic information of the software projects in our data set is covered within the study, and therefore assessing the applicability of our findings in different contexts is up to the readers. Nevertheless, a replication of this study in other countries or regions would be needed to confirm generalizability of our findings.

Also, the generalization (external) validity of our study is related to the work of Wieringa and Daneva (Wieringa and Daneva 2015) in which they discussed strategies for generalizing software engineering theories, as we discussed in Section 6.1. Generalizations can be made robust by analytical induction (Wieringa and Daneva 2015). Thus, while single empirical studies, such as the one we reported in this paper and also those in the past, cannot be individually used for generalization; multiple empirical studies and their cross-synthesis can lay the foundation of theories about CSFs in software projects.

7 Conclusions and future work

The software engineering community has, over the years, identified a set of critical success factors (CSFs) that seem to be correlated with success of software projects. In this work, we adapted and slightly revised a contingency fit model of CSFs from a recent work (Ahimbisibwe et al. 2015) and conducted an empirical investigation on the correlation of CSFs and their relative importance (ranking) with success of software projects. The empirical analysis was based on the data via an online questionnaire-based survey in the Turkish software industry, in which the dataset included data from 101 software projects. Among our findings was the following: the top three CSFs, having the largest count of significant associations with variables describing project success, were: (1) project team's experience with the software development methodologies, (2) project team's expertise with the task, and (3) project monitoring and controlling. By adding to the body of empirical evidence in this field, the results of the study will be useful for a wide audience. Software managers at all levels can use the results to prioritize the improvement opportunities in their organizations as identified by the ranked CSFs in this study. Software engineers and developers might use the results to improve their skills in different dimensions. Last but not the least, researchers might use the results to prioritize and conduct follow-up in-depth studies on various factors.

We also compared our ranking of proposed CSFs with the ranking of CSFs found in the literature by Ahimbisibwe et al. (Ahimbisibwe et al. 2015). Despite the lack of significant overall correlation between the rankings, we observed that a subset of the factors (e.g., level of project planning, monitoring and controlling, and change management skills) have similar rankings in both our study and Ahimbisibwe et al. (Ahimbisibwe et al. 2015).

We were interested about the positions held by participants in projects, since people in different SE positions differ in seeing and rating the importance of different issues (Garvin 1988). As other researchers have also reported, e.g., in (Drew Procaccino et al. 2002), one of the most compelling reasons to measure the perceptions of those involved with software development is that participant perceptions of project success, consistent or not with reality, can dramatically affect the health of perceptions of a project manager, a project team, and thus the health of a project. One of the lessons learned from a previous study (Drew Procaccino et al. 2002), for example, involves the perceptions that the software development staff has about management’s view of project success and failure. As reported in earlier work (Drew Procaccino et al. 2002), there is a vast difference between developers’ perceptions of project success factors, and their perceptions of how management personnel view project success and failure. We plan to investigate this issue based on our dataset in upcoming works.

On the other perspective, we considered whether our work could have the capability to revise and/or impact the “theory” in this field. However, it goes without saying that our work is only one exploratory case study (based on survey data) to rank the CSFs in this area. As discussed in the literature, theory is the means through which one may generalize analytically (Shadish et al. 2001), thus enabling generalization from situations in which statistical generalization is not desirable or possible, such as from multiple case studies, across populations. We reviewed several sources about theory building in software engineering, e.g., (Sjøberg et al. 2008). Accordingly, it is clear that more comprehensive studies (even at the “meta” level, e.g., secondary or even tertiary studies) to synthesize multiple case studies and from across populations are needed to improve the theory in the field of CSFs in software projects, e.g., using techniques such as meta-analysis (Cruzes and Dyb 2010).

Among our plans for future work directions are the followings: (1) to conduct semi-structured interviews to get more in-depth insights about the high ranked CSFs; (2) investigation of low ranked factors, e.g., software development methodology and other project factors, which were quite disregarded in both our study and also in (Ahimbisibwe et al. 2015); and (3) conducting cross-factor analysis (as we did in (Garousi et al. 2016e)) to consider factors such as role, experience, holding certifications, and target sector of respondents when analyzing CSFs. It may be interesting to study if these variables had an influence on CSFs.

Acknowledgements The authors would like to acknowledge the help of Özge Gürbüz, Rahime Belen Sağlam, Deniz Akdur, Fatma Akıncı, Elif Aleçakır, Betül Aygün, Kemal Dinçer, Mehmet Söylemez, and Rana Özakıncı during the survey piloting phase, and all the practitioners from the industry who voluntarily participated in the survey and helped us in the publicity of our survey. The survey used in this study received ethics approval from Hacettepe University’s Research Ethics Board, prior to its execution. Dietmar Pfahl was supported by the institutional research grant IUT20-55 of the Estonian Research Council.

Appendix 1

Table 9 A summary of a selected short-list of studies on CSFs in software projects. The table below presents a selected short list of studies on CSFs of software projects, sorted by year of publication (the most recent work first)

Ref.	Year	Study goal	Contribution type			Summary of findings
			Proposed a set of CSFs	An empirical study on CSFs	General discussions on CSFs	
(Ahimbisibwe et al. 2015)	2015	Identified and categorized CSFs and developed a contingency fit model contrasting perspectives of traditional plan-based and agile methodologies	Using an informal literature review			A total of 37 CSFs for software projects were extracted and synthesized from 148 articles. The CSFs were categorized into three major categories: organizational, team and customer factors. The individual CSFs were ranked based on how frequently they have appeared in the previous studies, and across the two development methodologies (traditional and agile).
(Lehtinen et al. 2014)	2014	Conducted an analysis of software project failures in four software companies in order to understand the causes of failures and their relationships		Empirical investigation (using the CSF framework presented in (McLeod and MacDonell 2011))		For each failure, the causes that interconnected different process areas and perceived as the most promising targets for process improvement were investigated by conducting root cause analysis.
(Whitney and Daniels 2013)	2013	Examined the root causes of IT project management failure stated in modern literature, and analyze these causes			Using an informal literature review	The degree of complexity within the projects is discussed from a systemic perspective related to emergence, non-monotonicity, and non-ergodicity.
(Hashim et al. 2013)	2013	Systematically reviewed the CSFs and project risks in the literature	Using a SLR approach	Using an opinion survey		By a SLR of 28 primary studies, they proposed 8 CSFs. The opinion survey also provided empirical data (frequencies) on those 8 CSFs.

Table 9 (continued)

Ref.	Year	Study goal	Contribution type			Summary of findings
			Proposed a set of CSFs	An empirical study on CSFs	General discussions on CSFs	
(Stankovic et al. 2013)	2013	Determined via a survey the CSFs that influence the success of agile software projects in IT companies located in the former Yugoslavia		Empirical investigation via survey (using the CSF framework presented in (Chow and Cao 2008))		The obtained results did not confirm that either of the factors from the CSF framework presented in (Chow and Cao 2008) can be considered as critical success factor in the space of former Yugoslavia IT companies.
(Subiyakto and bin Ahlan 2013)	2013	Developed a CSFs framework for helping project managers to deal with the complexity of project dimensions	Using an informal literature review			Proposed 18 CSFs in 4 dimensions: systematic, managerial, directional, environmental
(Sudhakar 2012)	2012	Developed a conceptual model of CSFs for software projects in these categorizes: product, team, project management and communication	Tertiary study (a literature review of secondary resources)			It presented a total of 35 CSFs in seven CSF categories: technical factors, organizational factors, environmental factors, product factors, project management factors, team factors, communication factors.
(Nasir and Sahibuddin 2011)	2012	To provide a comprehensive review on different project sizes in various domains and in multiple countries for the CSFs of software projects.	Literature review of primary studies			It presented a total of 26 CSFs by reviewing 43 articles from the years 1990 to 2010. Content analysis and frequency analysis were used. The authors also compared the set of CSFs to CSFs proposed in (Fortune and White 2006).
(McLeod and MacDonell 2011)	2011	Presented a classification framework that represents an abstracted and synthesized view of the types of CSFs	Using a literature review (not a SLR) of 177 empirical studies			Proposed 18 CSFs in 4 dimensions: institutional context, people and action, project content, development processes. It also conducted a comparison of the new CSF framework with four prior classificatory schemes (Lyytinen and Hirschheim 1987; Poulymenakou and

Table 9 (continued)

Ref.	Year	Study goal	Contribution type			Summary of findings
			Proposed a set of CSFs	An empirical study on CSFs	General discussions on CSFs	
						Holmes 1996; Scott and Vessey 2002; Tom and Brian 2001)
(Cerpa et al. 2010)	2010	Conducted an opinion survey to determine factors that affect project outcomes, built prediction models based on the data collected from multiple companies, and then tested the performance on data from a single organization.		Empirical investigation (via survey)		The set of CSFs was developed in the study without systematic usage of the literature.
(Chow and Cao 2008)	2008	Identified the CSFs specific for Agile projects	Literature review	Empirical validation (via survey)		A preliminary list of potential CSFs of Agile projects were identified and compiled based on existing literature. Reliability analysis and factor analysis were conducted to consolidate this preliminary list into a final set of 36 CSFs in five categories: organization, people, processes, technical and project. A survey was conducted among Agile professionals to validate the proposed model, gathering survey data from 109 Agile projects from 25 countries across the world.
(Agarwal and Rathod 2006)	2006	Examined the views of different groups of stakeholders about project success criteria		Empirical investigation (via survey)		An exploratory survey is conducted to determine a view of a successful project from the perceptions of project stakeholders (e.g. developers, project managers, and customer account managers). The study found surprising uniformity in different constituents of the group of stakeholders and all of them

Table 9 (continued)

Ref.	Year	Study goal	Contribution type			Summary of findings
			Proposed a set of CSFs	An empirical study on CSFs	General discussions on CSFs	
						overwhelmingly considered meeting the 'scope' of software projects.
(Berntsson-Svensson and Aurum 2006)	2006	Assessing whether there could be different perceptions about what effect various factors have on software project success among different industries		Empirical investigation		There are differences in which factors are important for project/product success across industries. This indicates that we need to take into account the industry type when addressing project/project success factors.
(Weber et al. 2003)	2003	Examined several Case-Based Reasoning (CBR) techniques to assess their accuracy in predicting software development project outcomes and identify CSFs.		Empirical investigation (via survey)		Data was collected from software developers who answered a questionnaire targeting a software project they have recently worked on. The questionnaire addressed both technical and managerial features of software projects. The results of these evaluations were compared with results from logistic regression analysis, which serves as a comparative baseline.
(Pereira et al. 2008)	2005	To discover if there exist significant cultural differences among practitioners' definition of a successful software project.		Empirical investigation (via survey)		Results from a survey of Chilean software practitioners' perceptions of project success were compared with results of a previous study in the context of US practitioners (Procaccino et al. 2005).
(Drew Procaccino et al. 2002)	2002	To investigate the early risk factors and their effect on software project success		Empirical investigation (via survey)		42 software projects were analyzed. The survey consisted of the following headings: requirements, management, customers and users, estimation and scheduling, project manager, software development process, and development personnel.

Table 9 (continued)

Ref.	Year	Study goal	Contribution type			Summary of findings
			Proposed a set of CSFs	An empirical study on CSFs	General discussions on CSFs	
(Scott and Vessey 2002)	2002	Managing risks in enterprise systems implementations	Proposed a classification of risk factors			Based on several case –study projects in which the authors were involved
(Tom and Brian 2001)	2001	The relationship between user participation and the management of change surrounding the development of information systems: a European perspective	Proposed a classification of CSFs			CSFs were based on a process-based study of the relationship between user participation and organizational change in the development and implementation of information systems in a large organization.
(Reel 1999)	1999	Critical success factors in software projects	Based on author's experience.			"*At least seven of 10 signs of IS project failures are determined before a design is developed or a line of code is written*". Discussed five essential factors to managing a successful software project: (1) Start on the right foot (Set realistic objectives and expectations; Build the right team; Give the team what they think they need); (2) Maintain momentum (Management, quality); (3) Track progress; (4) Make smart decisions; (5) Institutionalize post-mortem analyses It also discussed 10 signs of project failures: (1) Project managers do not understand users' needs; (2) The project's scope is ill-defined (3) Project changes are managed poorly; (4) The chosen technology changes; (5) Business needs change; (6) Deadlines are unrealistic; (7) Users are resistant; (8) Sponsorship is lost; (9) The

Table 9 (continued)

Ref.	Year	Study goal	Contribution type			Summary of findings
			Proposed a set of CSFs	An empirical study on CSFs	General discussions on CSFs	
						project lacks people with appropriate skills; (10) Managers ignore best practices and lessons learned.
(Poulymenakou and Holmes 1996)	1996	A contingency framework for the investigation of information systems failure	Proposed a classification of CSFs			CSFs were under these groups: users, social interaction, project management, use of a standard method, management of change, organizational properties
(Lyytinen and Hirschheim 1987)	1987	Information systems failures: a survey and classification of the empirical literature	Proposed a classification of CSFs based on classification of the empirical literature			Failure notions are classified into: correspondence failure, process failure, interaction failure, and expectation failure. 12 classes of reasons for IS failure are draw from the literature, in four groups: IS, IS environment, IS development, and IS development environment.

Appendix 2

Table 10 List of the questions in the survey. The list of the questions in the survey and types of their answers are shown in table below. The full details about the possible answers can be found in an online resource (Garousi and Tarhan 2018)

Aspect	Survey Questions (and Metrics)	Type of Answers			
		Single answer from a list	Multiple answers could be chosen from a list	Likert scale	Integer
Profiles and demographics of practitioners	1. Please choose your gender.	x			
	2. What is (are) your position(s) at the time when you were involved in the project?		x		
	3. Please choose the city that you work(ed) in at the time when you were involved in the project?	x			
	4. How many years of work experience do you have in IT and the software development industry?				x
	5. Are you the holder of any of the following certifications?		x		
Profiles and demographics of companies	6. What is the target sector of the products developed (services offered) by your company?		x		
	7. What is the size of your company/organization (number of employees participating in software development)?)?	x			
	8. Is your company/team certified under any of the following process improvement models?		x		
	9. If your company/team has been certified under CMMI or ISO/IEC 15504, what is its process maturity certification level?	x			
Profiles and demographics of projects	10. What was the nature of the software engineering project (in terms of the software engineering phases/activities)?		x		
	11. What was the development type of the project?		x		
	12. What was the independence level of the software engineering project?	x			
Organizational factors	13. What was the support level of top-level management?			x	
	14. What was the type of organizational culture and management style when the project under study was conducted?			x	
	15. How well defined is/was the organizational and project vision and mission?			x	

Springer

Table 10 (continued)

Aspect	Survey Questions (and Metrics)	Type of Answers			
		Single answer from a list	Multiple answers could be chosen from a list	Likert scale	Integer
	16. What was the level of organizational maturity in terms project planning in the project under study?			x	
	17. What was the level of project monitoring and controlling?			x	
	18. What was the level of Organization-wide change management practices?			x	
Team factors	19. What was the level of project team commitment in the project under study?			x	
	20. What was the extent of internal project communication during the project under study?			x	
	21. What was the level of project team empowerment in the project under study?			x	
	22. How well was the project team's composition made?			x	
	23. What was the average level of team members' general expertise and competency in general software engineering concepts?			x	
	24. What was the average level of team members' expertise and competency with the tasks assigned within the project?			x	
	25. What was the average level of team members' experience with the software development methodologies used in the project?			x	
Customer factors	26. What was the level of User (client) participation/involvement in the project?			x	
	27. What was the level of User (client) support in the project?			x	
	28. What was the level of customer skills, training and education in IT in general?			x	
	29. What was the level of customer (client) experience in their own business domain?			x	
Contingency (project) factors	30. What was the level of technological uncertainty in the project?			x	
	31. What was the level of project complexity?			x	
	32. What was the level of urgency during the project overall?			x	
	33. What was the project size in terms of number of team members?			x	
	34. What was the level of requirement specification changes during the project?			x	
	35. What was the level of project criticality?			x	
	36. What type of software development methodologies was used in the project?			x	
Characteristics of the process	37. To what extent did the project finish on budget?			x	
	38. To what extent did the project finish on schedule/time?			x	
	39. To what extent did the project conform to the initially-defined scope (features)?			x	

Table 10 (continued)

Aspect	Survey Questions (and Metrics)	Type of Answers			
		Single answer from a list	Multiple answers could be chosen from a list	Likert scale	Integer
	40. To what extent has the project been influential in positive team building, team dynamics, and enhancing team members' opinions about each other?			x	
Characteristics of the resulting product/service	41. What was the overall quality of software product/service delivered?			x	
	42. To what extent has the software product/service been able to generate revenues for the organization?			x	
	43. What was the functional suitability of the software/service delivered?			x	
	44. What was the reliability of software/service delivered?			x	
	45. What was the performance efficiency of software/service delivered?			x	
	46. What was the usability of software/service delivered?			x	
	47. What was the security-related quality of software/service delivered?			x	
	48. What was the compatibility of software/service delivered?			x	
	49. What was the maintainability of software/service delivered?			x	
	50. What was the transferability of software/service delivered?			x	
	51. What was the user satisfaction with the software/service delivered?			x	
	52. What was the team satisfaction with the software/service delivered?			x	
	53. What was the top-management satisfaction with the software/service delivered?			x	

Appendix 3

Table 11 Factor and variable codes matrix. For brevity and to prevent repeating the long names of factors, we use codes in this study. The following table presents the codes representing the factors

Group	Factor or variable	Code
Organizational factors (OF)	Top management support	OF.01
	Organizational culture and management style	OF.02
	Vision and mission	OF.03
	Project planning	OF.04
	Project monitoring and controlling	OF.05
	Organization-wide change management practices	OF.06
Team factors (TF)	Team commitment	TF.01
	Internal team communication	TF.02
	Team empowerment	TF.03
	Team composition	TF.04
	Team's general expertise	TF.05
	Team's expertise in the task and domain	TF.06
	Team's experience with the development methodologies	TF.07
Customer factors (CF)	Customer participation	CF.01
	Customer support	CF.02
	Customer skill, training and education in IT	CF.03
	Customer experience in their own business domain	CF.04
Project factors (PF)	Technological uncertainty	PF.01
	Technical complexity	PF.02
	Urgency	PF.03
	Relative project size	PF.04
	Specification changes	PF.05
	Project criticality	PF.06
	Software development methodology	PF.07
Process factors (ProcF)	Budget	ProcF.01
	Schedule	ProcF.02
	Scope	ProcF.03
	Team building and team dynamics	ProcF.04
Product-quality factors (ProdF)	Overall quality	ProdF.01
	Business and revenue generation	ProdF.02
	Functional suitability	ProdF.03
	Reliability	ProdF.04
	Performance efficiency	ProdF.05
	Usability	ProdF.06
	Security	ProdF.07
	Compatibility	ProdF.08
	Maintainability	ProdF.09
	Transferability	ProdF.10
Satisfaction factors (SF)	User satisfaction	SF.01
	Team satisfaction	SF.02
	Top management satisfaction	SF.03

Appendix 4

Table 12 Details of cross-correlations analysis of CSFs

		OF Organizational factors					TF Team factors							CF Customer factors				PF Contingency (project) factors						
		02	03	04	05	06	01	02	03	04	05	06	07	01	02	03	04	01	02	03	PF.04	PF.05	PF.06	PF.07
Cross-correlations values among CSFs – original dataset																								
OF	01	0.13	0.49	0.16	0.39	0.22	0.33	0.14	0.26	0.23	0.04	0.14	0.26	0.32	0.20	0.23	0.15	0.12	0.04	0.06	0.17	0.07	0.22	0.05
	02		−0.02	−0.11	−0.07	−0.16	0.09	0.11	0.27	0.13	0.11	0.03	−0.04	0.15	0.13	0.26	0.07	0.26	−0.01	−0.27	−0.20	−0.15	−0.28	−0.02
	03			0.27	0.41	0.34	0.33	0.26	0.15	0.38	0.07	0.17	0.36	0.36	0.28	0.17	0.36	−0.06	0.07	0.07	0.11	−0.12	0.21	−0.03
	04				0.70	0.43	0.25	0.26	0.22	0.35	0.30	0.23	0.25	0.17	0.26	0.23	0.12	−0.05	0.13	−0.03	0.33	0.00	0.18	−0.16
	05					0.48	0.40	0.27	0.21	0.46	0.34	0.38	0.29	0.29	0.20	0.31	0.12	−0.09	0.14	−0.03	0.28	0.04	0.15	−0.02
	06						0.25	0.21	0.16	0.37	0.19	0.31	0.28	0.18	0.10	0.14	0.20	−0.02	0.21	0.03	0.18	−0.05	0.17	0.02
TF	01							0.49	0.29	0.47	0.24	0.24	0.42	0.30	0.15	0.25	0.20	0.04	0.04	−0.06	0.08	0.05	0.01	−0.03
	02								0.39	0.44	0.27	0.16	0.40	0.18	0.13	0.12	0.13	−0.02	0.11	−0.08	0.02	−0.20	0.06	−0.18
	03									0.30	0.38	0.18	0.25	0.06	0.07	0.15	0.01	0.14	0.09	−0.11	−0.08	−0.05	−0.03	−0.18
	04										0.44	0.36	0.46	0.18	0.22	0.34	0.13	0.20	0.14	0.06	0.18	0.08	0.19	−0.02
	05											0.51	0.51	0.07	0.16	0.17	−0.02	0.19	0.21	−0.03	0.09	0.01	0.15	−0.17
	06												0.33	0.20	0.17	0.26	0.27	0.00	0.16	−0.17	0.03	−0.01	0.09	0.00
	07													0.19	0.13	0.17	0.23	0.08	0.09	0.12	0.01	−0.03	0.09	−0.13
CF	01														0.75	0.48	0.48	0.00	0.08	0.10	0.09	−0.02	0.22	−0.25
	02															0.45	0.34	−0.02	0.00	0.04	0.19	0.07	0.25	−0.25
	03																0.38	0.18	−0.07	−0.09	−0.07	0.03	0.04	0.04
	04																	−0.10	0.08	0.03	0.10	−0.25	0.05	−0.14
PF	01																		0.19	0.05	−0.01	0.05	0.00	0.07
	02																			0.11	0.42	0.25	0.45	−0.07
	03																				0.29	0.10	0.32	−0.13
	04																					0.23	0.39	−0.17
	05																						0.18	−0.08
	06																							−0.04
	07																							

Table 12 (continued)

		OF Organizational factors					TF Team factors							CF Customer factors				PF Contingency (project) factors						
		02	03	04	05	06	01	02	03	04	05	06	07	01	02	03	04	01	02	03	PF.04	PF.05	PF.06	PF.07
Cross-correlations values among CSFs – balanced dataset																								
OF	01	−0.03	0.58	0.19	0.44	0.31	0.49	0.38	0.24	0.21	−0.17	0.26	0.29	0.35	0.17	0.17	0.27	−0.21	−0.10	0.20	0.10	−0.11	0.18	0.24
	02		−0.07	0.00	−0.13	−0.26	0.03	0.30	0.23	0.13	−0.11	−0.06	−0.16	0.07	0.09	0.24	0.22	0.11	−0.07	−0.37	−0.34	−0.23	−0.26	−0.08
	03			0.43	0.51	0.60	0.58	0.37	0.12	0.38	−0.14	0.10	0.29	0.49	0.36	0.28	0.49	−0.16	0.19	0.08	0.15	−0.24	0.21	0.07
	04				0.83	0.52	0.26	0.26	0.07	0.45	0.28	0.24	0.17	0.27	0.43	0.38	0.22	−0.17	0.05	0.08	0.44	0.10	0.28	−0.19
	05					0.58	0.45	0.32	0.15	0.39	0.28	0.46	0.23	0.41	0.35	0.45	0.30	−0.29	0.03	0.09	0.37	0.06	0.25	0.05
	06						0.31	0.06	0.18	0.19	−0.02	0.40	0.02	0.14	0.04	0.16	0.21	−0.36	0.12	0.01	0.21	−0.07	0.10	0.27
TF	01							0.57	0.32	0.52	0.08	0.36	0.37	0.34	0.20	0.38	0.49	−0.08	0.03	−0.12	0.07	−0.02	0.13	0.01
	02								0.28	0.57	−0.01	0.19	0.18	0.22	0.23	0.20	0.32	−0.03	−0.04	−0.24	0.10	−0.07	−0.10	−0.13
	03									0.27	0.32	0.35	0.18	−0.19	−0.21	0.18	0.14	0.03	0.00	−0.26	−0.08	−0.14	−0.20	0.13
	04										0.17	0.15	0.31	0.16	0.22	0.51	0.21	0.02	0.00	0.01	0.11	0.16	0.16	−0.05
	05											0.55	0.42	0.05	0.19	0.10	−0.08	0.05	0.14	−0.03	0.10	0.18	0.23	−0.07
	06												0.33	0.21	0.14	0.09	0.19	−0.20	0.20	−0.06	0.13	0.09	0.13	0.10
	07													0.18	0.09	0.04	0.24	−0.01	0.13	−0.02	0.05	0.14	0.10	−0.15
CF	01														0.73	0.48	0.48	−0.09	0.13	0.21	0.07	−0.11	0.39	−0.05
	02															0.44	0.28	−0.07	0.02	0.19	0.27	0.05	0.40	−0.14
	03																0.37	0.07	−0.04	−0.11	−0.04	0.04	0.16	0.19
	04																	0.01	0.23	−0.17	0.13	−0.39	−0.02	−0.04
PF	01																		0.29	0.01	0.01	0.02	0.01	0.21
	02																			0.17	0.39	0.26	0.36	−0.10
	03																				0.42	0.09	0.41	−0.12
	04																					0.21	0.29	−0.34
	05																						0.21	−0.14
	06																							0.13
	07																							

References

Agarwal, N., & Rathod, U. (2006). Defining 'success' for software projects: an exploratory revelation. *International Journal of Project Management, 24*(4), 358–370.

Ahimbisibwe, A., Cavana, R. Y., & Daellenbach, U. (2015). A contingency fit model of critical success factors for software development projects. *Journal of Enterprise Information Management, 28*(1), 7–33.

Akdur, D., Garousi, V., Demirörs, O. (2015). A survey on software modeling and model-driven techniques in embedded systems engineering: Results from Turkey. In *Proceedings of the Turkish National Software Engineering Symposium "Ulusal Yazılım Mühendisliği Sempozyumu" (UYMS)*.

Akkaya, M. U. et al. (2012). The software sector in Turkey (Turkish: Türkiye'de Yazilim Sektörü). http://www.sde.org.tr/userfiles/file/TURKIYEDE_YAZILIM_%20SEKTORU.pdf.

Belle, G. V. (2002). *Statistical rules of thumb*. John Wiley & Sons.

Berntsson-Svensson, R., & Aurum, A. (2006). Successful software project and products: An empirical investigation. *Presented at the proceedings of acm/ieee international symposium on empirical software engineering*.

Boehm, B. (2006). A view of 20th and 21st century software engineering. *Presented at the proceedings of the 28th international conference on software engineering*, Shanghai, China.

Caralli, R. (2004). The critical success factor method: Establishing a Foundation for Enterprise Security Management. *Technical Report (CMU/SEI-2004-TR-010)*. Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.

Cerpa, N., Bardeen, M., Kitchenham, B., & Verner, J. (2010). Evaluating logistic regression models to estimate software project outcomes. *Information and Software Technology, 52*(9), 934–944.

Chandrasekaran, V. (2011). Correlation or Causation? https://www.bloomberg.com/news/articles/2011-12-01/correlation-or-causation. Last accessed: Oct. 2017.

Chow, T., & Cao, D.-B. (2008). A survey study of critical success factors in agile software projects. *Journal of Systems and Software, 81*(6), 961–971.

Cohen, J. (1988). *Statistical Power Analysis for the Behavioral Sciences*, 2nd ed. Lawrence Erlbaum Associates.

Cruzes, D. S., Dyb, T. (2010). And #229, Synthesizing evidence in software engineering research. Presented at the proceedings of the 2010 ACM-IEEE international symposium on empirical software engineering and measurement, Bolzano-Bozen, Italy.

Davis, K. (2014). Different stakeholder groups and their perceptions of project success. *International Journal of Project Management, 32*(2), 189–201.

Drew Procaccino, J., Verner, J. M., Overmyer, S. P., & Darter, M. E. (2002). Case study: factors for early prediction of software development success. *Information and Software Technology, 44*(1), 53–62.

Ellis, C. (2005). Why customer satisfaction is your most important project metric. http://www.cio.com.au/article/570800/why-customer-satisfaction-your-most-important-project-metric/, Last accessed: Oct. 2017.

Fortune, J., & White, D. (2006). Framing of project critical success factors by a systems model. *International Journal of Project Management, 24*(1), 53–65.

Garousi, V., & Felderer, M. (2017). Worlds apart: Industrial and academic focus areas in software testing. *IEEE Software, 34*(5), 38–45.

Garousi, V., & Herkiloğlu, K. (2016). Selecting the right topics for industry-academia collaborations in software testing: An experience report. In *IEEE International Conference on Software Testing, Verification, and Validation*, pp. 213–222.

Garousi, V., & Tarhan, A. (2018). Investigating the impact of team formation by introversion/extraversion in software projects. *Balkan Journal of Electrical and Computer Engineering, 6*(2), 64–72.

Garousi, V., & Varma, T. (2010). A replicated survey of software testing practices in the Canadian province of Alberta: What has changed from 2004 to 2009? *Journal of Systems and Software, 83*(11), 2251–2262.

Garousi, V., & Zhi, J. (2013). A survey of software testing practices in Canada. *Journal of Systems and Software, 86*(5), 1354–1376.

Garousi, G., Garousi, V., Ruhe, G., Zhi, J., Moussavi, M., & Smith, B. (2015a). Usage and usefulness of technical software documentation: An industrial case study. *Elsevier Journal of Information and Software Technology, 57*, 664–682.

Garousi, V., Coşkunçay, A., Can, A. B., & Demirörs, O. (2015b). A survey of software engineering practices in Turkey. *Journal of Systems and Software, 108*, 148–177.

Garousi, V., Eskandar, M. M., Herkiloğlu, K. (2016a). Industry-academia collaborations in software testing: Experience and success stories from Canada and Turkey. *Software Quality Journal, special issue on Industry Academia Collaborations in Software Testing,* pp. 1–53.

Garousi, V., Ergezer, E. G., Herkiloğlu, K. (2016b). Usage, usefulness and quality of defect reports: An industrial case study in the defence domain. In *International Conference on Evaluation and Assessment in Software Engineering (EASE)*, pp. 277–282.

Garousi, V., Tarhan, A., Pfahl, D., Coşkunçay, A., Demirörs, O. (2016c). Survey of project characteristics in software engineering (Questionnaire). https://doi.org/10.6084/m9.figshare.4009692, Last accessed: Oct. 2017.

Garousi, V., Tarhan, A., Pfahl, D., Coşkunçay, A., Demirörs, O. (2016d) Survey of project characteristics in software engineering (dataset). https://doi.org/10.6084/m9.figshare.4012779, Last accessed: Oct. 2017.

Garousi, V., Coşkunçay, A., Demirörs, O., & Yazici, A. (2016e). Cross-factor analysis of software engineering practices versus practitioner demographics: An exploratory study in Turkey. *Journal of Systems and Software, 111*, 49–73.

Garvin, D. A. (1988). *Managing quality: The strategic and competitive edge*. Simon and Schuster.

Gates, L. P. (2010). Strategic planning with critical success factors and future scenarios: An integrated strategic planning framework. *Technical Report (CMU/SEI-2010-TR-037),* Pittsburgh, PA: Software Engineering Institute, Carnegie Mellon University.

Groves, R. M., Fowler, F. J., Couper, M. P., Lepkowski, J. M., Singer, E., Tourangeau, R. (2009) *Survey Methodology*, 2nd edition. Wiley.

Hashim, R., Abbas, M., Hashim, M. (2013). Critical success factors assessment in software projects. In *Science and Information Conference (SAI)*, 2013, pp. 282–287.

Hatfield, J., Faunce, G., Job, R. (2006). Avoiding confusion surrounding the phrase "correlation does not imply causation. *Teaching of Psychology.*

Hoegl, M., & Gemuenden, H. G. (2001). Teamwork quality and the success of innovative projects: a theoretical concept and empirical evidence. *Organization Science, 12*(4), 435–449.

Howell, D. C. (2013). *Fundamental statistics for the behavioral sciences*. Cengage Learning.

Karapıçak, Ç., & Demirörs, O. (2013). A case study on the need to consider personality types for software team formation. In T. Woronowicz, T. Rout, R. O'Connor, & A. Dorling (Eds.), *Software process improvement and capability determination* (Vol. 349, pp. 120–129). Berlin: Communications in Computer and Information Science: Springer.

Lehtinen, T. O. A., Mäntylä, M. V., Vanhanen, J., Itkonen, J., & Lassenius, C. (2014). Perceived causes of software project failures – An analysis of their relationships. *Information and Software Technology, 56*(6), 623–643.

Linåker, J., Sulaman, S. M., Maiani de Mello, R., Höst, M., Runeson, P. (2015). Guidelines for conducting surveys in software engineering. *Technical report #5366801*, Sweden: Lund University, https://lup.lub.lu.se/search/publication/5366801.

Lunsford, T. R., & Lunsford, B. R. (1995). The research sample, part I: sampling. *Journal of Prosthetics and Orthotics, 7*, 105–112.

Lyytinen, K., Hirschheim, R. (1987). Information systems failures: a survey and classification of the empirical literature. In *Oxford Surveys in Information Technology*: Oxford University Press, Inc., pp. 257–309.

Malhotra, R. (2016). *Empirical research in software engineering: Concepts, analysis, and applications*. CRC Press.

McLeod, L., & MacDonell, S. G. (2011). Factors that affect software systems development project outcomes: a survey of research. *ACM Computing Surveys, 43*(4), 1–56.

Nasir, M. H. N., & Sahibuddin, S. (2011). Critical success factors for software projects: a comparative study. *Scientific Research and Essays, 6*(10), 2174–2186.

Nielsen, M. (2012). If correlation doesn't imply causation, then what does? http://www.michaelnielsen.org/ddi/if-correlation-doesnt-imply-causation-then-what-does/. Last accessed: Oct. 2017.

Novella, S. (2009). Evidence in medicine: Correlation and causation. http://sciencebasedmedicine.org/evidence-in-medicine-correlation-and-causation/. Last accessed: Oct. 2017.

Pereira, J., Cerpa, N., Verner, J., Rivas, M., & Procaccino, J. D. (2008). What do software practitioners really think about project success: a cross-cultural comparison. *Journal of Systems and Software, 81*(6), 897–907.

Pinto, J. K., Rouhiainen, P., & Trailer, J. W. (2000). Project success and customer satisfaction: Toward a formalized linkage mechanism*. In R. A. Lundin & F. Hartman (Eds.), *Projects as business constituents and guiding motives* (pp. 103–115). Boston: Springer US.

Poulymenakou, A., & Holmes, A. (1996). A contingency framework for the investigation of information systems failure. *European Journal of Information Systems, 5*(1), 34–46.

Procaccino, J. D., Verner, J. M., Shelfer, K. M., & Gefen, D. (2005). What do software practitioners really think about project success: an exploratory study. *Journal of Systems and Software, 78*(2), 194–203.

Punter, T., Ciolkowski, M., Freimut, B., John, I. (2003) Conducting on-line surveys in software engineering. In *Proceedings of International Symposium on Empirical Software Engineering*, pp. 80–88.

Reel, J. S. (1999). Critical success factors in software projects. *IEEE Software, 16*(3), 18–23.

Rockart, J. (1979). Chief executives define their own information needs. *Harvard Business Review*, 81–92.

Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Emprircal Software Engineering, 14*(2), 131–164.

Savolainen, P., Ahonen, J. J., & Richardson, I. (2012). Software development project success and failure from the supplier's perspective: A systematic literature review. *International Journal of Project Management, 30*(4), 458–469.

Scott, J. E., & Vessey, I. (2002). Managing risks in enterprise systems implementations. *Communications of the ACM, 45*(4), 74–81.

Shadish, W. R., Cook, T. D., Campbell, D. T. (2001) .*Experimental and quasi-experimental designs for generalized causal inference,* 2nd edition. Cengage Learning.

Sjøberg, D. I. K., Dybå, T., Anda, B. C. D., & Hannay, J. E. (2008). Building theories in software engineering. In F. Shull, J. Singer, & D. I. K. Sjøberg (Eds.), *Guide to advanced empirical software engineering* (pp. 312–336). London: Springer London.

Sjoeberg, D. I. K., Hannay, J. E., Hansen, O., Kampenes, V. B., Karahasanovic, A., Liborg, N. K., & Rekdal, A. C. (2005). A survey of controlled experiments in software engineering. *IEEE Transactions on Software Engineering, 31*(9), 733–753.

Stankovic, D., Nikolic, V., Djordjevic, M., & Cao, D.-B. (2013). A survey study of critical success factors in agile software projects in former Yugoslavia IT companies. *Journal of Systems and Software, 86*(6), 1663–1678.

Subiyakto, A., & bin Ahlan, A. R. (2013). A coherent framework for understanding critical success factors of ICT project environment. In *Research and Innovation in Information Systems (ICRIIS), 2013 International Conference on*, pp. 342–347.

Sudhakar, G. P. (2012). A model of critical success factors for software projects. *Journal of Enterprise Information Management, 25*(6), 537–558.

Tessem, B., Maurer, F. (2007). Job satisfaction and motivation in a large agile team. In *Proceedings of International Conference on Agile Processes in Software Engineering and Extreme Programming*, pp. 54–61.

Thompson, T. L. (2008). The relationship between job satisfaction and project success: A quantitative study of project managers in Houston, Texas. School of Business, Capella University.

Tom, B., & Brian, F. (2001). The relationship between user participation and the management of change surrounding the development of information systems: a European perspective. *Journal of Organizational and End User Computing (JOEUC), 13*(1), 12–25.

Torchiano, M., Fernández, D. M., Travassos, G. H., Mello, R. M. D. (2017). Lessons learnt in conducting survey research. In *IEEE/ACM International Workshop on Conducting Empirical Studies in Industry*, pp. 33–39.

Weber, R., Waller, M., Verner, J., & Evanco, W. (2003). Predicting software development project outcomes. In K. Ashley & D. Bridge (Eds.), *Case-based reasoning research and development* (Vol. 2689, pp. 595–609). Berlin: Lecture Notes in Computer Science: Springer.

Whitney, K. M., & Daniels, C. B. (2013). The root cause of failure in complex IT projects: Complexity itself. *Procedia Computer Science, 20*, 325–330.

Wieringa, R., & Daneva, M. (2015). Six strategies for generalizing software engineering theories. *Science of Computer Programming, 101*, 136–152.

Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., Wesslén, A. (2012). *Experimentation in software engineering: An introduction*. Springer.

Vahid Garousi is an Associate Professor of Software Engineering in the Information Technology Group (INF) of Wageningen University, the Netherlands. Previously, he was an Associate Professor of Software Engineering in Hacettepe University in Ankara, Turkey (2015-2017) and an Associate Professor of Software Engineering in the University of Calgary, Canada (2006-2014). He is the Manager of Bahar Software Engineering Consulting BV and has also provided software engineering consulting to many companies in several countries since 2003. He received his PhD in Software Engineering in Carleton University, Canada, in 2006. His research interests in software engineering include: software testing, empirical studies, action-research, and industry-academia collaborations. Vahid was an IEEE Computer Society Distinguished Visitor from 2012 to 2015. He is a member of the IEEE and the IEEE Computer Society, and is also a licensed professional engineer (PEng) in the Canadian province of Alberta.

Ayça Tarhan works as a researcher and practitioner in the area of software engineering for fifteen years. She has led or been involved in projects originating from industry-academia collaborations on software quality evaluation, business process modeling, and system requirements elicitation. She worked as an adjunct faculty at Informatics Institute of Middle East Technical University from 2002 to 2006, and was a visiting researcher at Department of Industrial Engineering and Innovation Sciences of Eindhoven University of Technology from 2013 to 2015. She currently works as an Assistant Professor at Computer Engineering Department of Hacettepe University (H.U.) where she also leads Software Engineering Research Group (HUSE). Her research interests include internal and external software quality, software development methodologies, software measurement, process maturity, and process mining.

Dietmar Pfahl earned his Master (1986) and PhD (2001) degrees in Germany, at the Universities of Ulm and Kaiserslautern, respectively. Before joining the University of Tartu in 2013 as an Associate Professor, he worked eight years in industry (Siemens AG, Germany), and held appointments with the University of Calgary, Canada, the Lund University, Sweden, and the University of Oslo, Norway. He also worked for the German Aerospace Research Establishment near Munich, Germany, the Fraunhofer IESE in Kaiserslautern, Germany, and the Simula Research Laboratory near Oslo, Norway. He was the founder and director of Pika Research, Inc., Claresholm, Canada, from 2009 to 2015. Since 2008, he is Adjunct Professor with the Schulich School of Engineering at the University of Calgary, Canada. In spring 2017, he was appointed Professor of Software Engineering at the University of Tartu, Estonia. Dietmar Pfahl's research interests focus on data-driven decision support in software engineering and management. Previously, he published articles on software process simulation and improvement, software release planning, and software testing. He has more than 100 publications in top-ranked software engineering journals and conference proceedings. He is a senior member of the IEEE and the ACM.

Ahmet Coşkunçay is an assistant professor in Computer Engineering Department of Ataturk University, Turkey. As a PhD holder in Information Systems from Middle East Technical University, he has been actively participating in private sector projects as a software engineering researcher and consultant. He is also the technology development zone vice general manager of AtaTeknokent. His research interests include business process management, software process improvement, software requirements engineering, and ontology development.

Onur Demirors is a Professor of Computer Engineering at the Izmir Institute of Technology and the strategy director of Bilgi Grubu Ltd.. His current research focuses on decentralized modelling, organizational change, software measurement, and management. He has leaded major research and application projects on developing improvement and modelling techniques, on establishing and implementing modelling approaches for organizations and on establishing measurement infrastructures for software organizations. He has leaded application projects for dozens of companies to improve their processes, to establish their measurement infrastructures, to create organizational knowledge structures and to identify their software needs. He continues to teach on decentralized modelling, event based systems, software project and quality management, software measurement and innovative software development approaches.

Affiliations

Vahid Garousi[1] • **Ayça Tarhan**[2] • **Dietmar Pfahl**[3,4] • **Ahmet Coşkunçay**[5] • **Onur Demirörs**[6,7]

Ayça Tarhan
atarhan@hacettepe.edu.tr

Dietmar Pfahl
dietmar.pfahl@ut.ee

Ahmet Coşkunçay
ahmet.coskuncay@bg.com.tr

Onur Demirörs
onurdemirors@iyte.edu.tr

1 Information Technology Group, Wageningen University, Wageningen, Netherlands

2 Software Engineering Research Group, Department of Computer Engineering, Hacettepe University, Ankara, Turkey

3 Institute of Computer Science, University of Tartu, Tartu, Estonia

4 Department of Electrical and Computer Engineering, University of Calgary, Calgary, Canada

5 Bilgi Grubu Ltd., Ankara, Turkey

6 Izmir Institute of Technology, Izmir, Turkey

7 University of New South Wales, Sydney, Australia