SACJ 34(1) July 2022
Research Article

Critical success factors in software development projects

Moketo Ephraim Bogopa[a], Carl Marnewick[b]

[a] Department of Information Systems, College of Science, Engineering and Technology, University of South Africa, South Africa
[b] Department of Applied Information Systems, College of Business and Economics, University of Johannesburg, Johannesburg, South Africa

ABSTRACT
Irrespective of the software development methodology, projects still fail at any development stage. Some of the factors that may affect project success or failure include process-related factors, such as an unrealistic budget and schedule, or technical-related factors like the methodology. The study aims to determine the influence of the most critical success factors of each category (people, process, and technical) on the overall outcome of software projects in the South African context. This study surveyed members of software development teams from across South African organisations and gathered information about software development project success and factors that contribute to project outcomes. A structured questionnaire was used to collect the study data. According to the results, software development projects need: (1) a committed and motivated team; (2) the client must be involved all the time; (3) specifications/requirements must be clear; (4) leadership must be good; (5) the project's goals and objectives must be well-defined. The critical success factors are ranked by their strength in relation to success criteria. The study further found that most of the perceived factors are related to non-technical factors. This research contributes to the current software industry debate about critical success factors categories in the South African context.

Keywords: Software development team, critical success factors, agile software development

Categories: • **Software and its engineering** ∼ **software creation and management, software development process management**

Email:
Moketo Ephraim Bogopa bogopme@unisa.ac.za (CORRESPONDING),
Carl Marnewick cmarnewick@uj.ac.za

Article history:
Received: 9 April 2020
Accepted: 13 April 2021
Available online: 22 July 2022

1 INTRODUCTION

Software development is getting more and more popular as time goes by, and everywhere there is a piece of software that performs a simple or complex activity. The development of software products is taking place in new areas as compared to 10 years back (Ahmed, 2012). Software development projects influence modern business across sectors and industries (Dubey, 2011). Many aspects of society are now relying on software products and other related technologies.

Bogopa, M.E., and Marnewick, C. (2022). Critical success factors in software development projects. *South African Computer Journal 34*(1), 1–34. https://doi.org/10.18489/sacj.v34i1.820

SACJ is a publication of the South African Institute of Computer Scientists and Information Technologists. ISSN 1015-7999 (print) ISSN 2313-7835 (online).

Also, many organisations rely on projects they run in order to sustain business every day (Elmezain et al., 2021) – without projects some organisations will not operate. The information sources and statistics that travel faster to the media were developed by software developers following any software development method, including open source. This will also require several small-medium IT companies to support the initiation.

The software development team might develop a piece of software for a bank, medical, industrial, military, sports event, businesses, mobile phones, or computing applications. For the users of cell phones, there are continuously new enhancements that must be downloaded to their mobile phones to fix bugs and improve performance. The end-users or software users know only the final software product because they use the software or applications daily. Again, if a bank wants to deploy a new version of its cell phone banking application to its clients, the clients only receive notification of the new software once the software was installed successfully. However, if the software has failed for whatever reason, the clients will not know, and the same will happen to the software application which has failed at a certain stage of development. Regardless of the software development model or methodology used, a software development project is prone to fail at any stage as reported by Sommerville (2016) and Tsui et al. (2014). Large software organisations are using traditional software development methodologies like Waterfall, RUP model, etc. (Bitla & Veesamsetty, 2019). The characteristics of projects that are following the Waterfall or Agile approach are unique, each methodology has its virtues, and each has its supporters and critics (Vijayasarathy & Butler, 2015). The question is: If the organisation delivers 100% project success while following a traditional methodology why should the organisation adopt the Agile methodology?

Irrespective of so many proposed software development methodologies, software projects continue to fail (Gulec et al., 2017). According to Sandstø and Reme-Ness (2021), the creation of agile methods was encouraged by difficulties encountered in order to achieve success in software projects. Covid-19 gave the IT industry reasons to adopt Agile in many situations because software was always required faster than before to address challenges and other medical issues. Furthermore, the researchers and practitioners have identified factors that are perceived as the main critical project success factors (Camilleri, 2016; R. Kaur & Sengupta, 2013; Nasir & Sahibuddin, 2011). The failure rates of software development projects are posing serious challenges to the software project industry and academics (Mtsweni et al., 2016). As a result, the contributors to project failure need to be discussed in more detail and the software development community should be informed about the critical factors that contribute to the success of the project. According to Du Randt et al. (2014) the research about software development success factors started in the 1960s.

There are many factors that can affect the successful delivery of software development projects, such as process-related factors like an unrealistic budget and schedule, or technical-related factors like the chosen methodology (Nasir & Sahibuddin, 2011). Also, people-related factors can contribute to the success of the projects, which are the software development team's skills or the lack of support from top management. Top management is ranked as the most important critical success factor in project success at large (Alkraiji et al., 2022). In this article,

the factors that influence the outcome of software development projects are investigated. For businesses to perform properly, the critical success factors of software development projects need more attention (Yaghoobi, 2018).

The main objective of this article is to determine the influence of software success factors on the overall outcome of software projects. The following sub-objectives will assist to evaluate:

(i) the people factors influencing the outcome of software development projects,

(ii) the process factors influencing the outcome of software development projects, and

(iii) the technical factors influencing the outcome of software development projects.

The motive and intention of every software development team are to deliver software with minimum cost, within minimum time, and the exact quality as expected by the customer (Ahmed, 2012). If this does not happen, they should learn from the failed software project and try to deliver the next project. Now software is becoming a fundamental part of any business, the failure of software involves business operations, and such failure affects the development team and the organisation as a whole.

This article is organised as follows: Section one has highlighted the position of software projects in society and justified the reason why such projects should not fail. Section two presents the literature which is relevant to the objectives of the study, and the first part provides an overview of traditional and agile methods followed by a review of software project success and critical success factors. Section three describes the methodology adopted which covers the questionnaire development and the background of the chosen respondents and the selection method used. The results about software development factors and success rate are presented and analysed in detail in section four. Finally, section five summarises the contribution of the study and highlights the limitation exposed, then a conclusion is drawn.

2 LITERATURE REVIEW

According to Ahmed (2012), a software project could be regarded as any one of the following: software development, customisation of software, integration of software, maintaining software, or any phase of software development (requirements, design, construction, or testing of a software product). The focus of this article is on software development. Software development projects are complex in nature and if not managed properly, will not meet business needs (Vavpotic et al., 2022). Software development is one fundamental activity whereby the software is designed and programmed (Elmezain et al., 2021). The common tasks performed during software development are requirements gathering and specification, design, code or unit test, integration and test and user support, and problem-fix (Davis, 2013; Tsui et al., 2014). In the Waterfall model, the tasks are organised in sequence, whereas in the Agile software model they are interleaved (Elmezain et al., 2021). The Agile approach is contrary to the Waterfall approach since it incorporates changes of requirements at any time (Beerbaum,

2021). With the Waterfall or traditional method, each member of the software development team knows at which point should they start their software project tasks. The organisation that specialises in software development manages the tasks to ensure success (Spenser, 2010). Diverse types of software development projects need different development processes.

Shaydulin and Sybrandt (2017) have compared both agile and traditional methods to answer questions like "Why does the Waterfall persist?" and "What makes Agile methods so popular?" According to Shaydulin and Sybrandt (2017), the Waterfall model has changed a lot because of the experience gained by the software developers while using the model, also the role played by the iterative approach. Again in Shaydulin and Sybrandt (2017), when the authors find out why agile methods are so popular, they evaluated many agile methodologies as compared to traditional software development methodologies. They discovered that agile methodologies will remain popular because they address both quality and agility criteria of software development projects. As described in the Agile Manifesto, agile software methodologies are XP, Scrum, Lean, Kanban, and so on (Almeida, 2017; Himmat & Osman, 2020; Joseph et al., 2016; Shaydulin & Sybrandt, 2017), and the traditional software development methodologies are known as Waterfall, RUP, and Spiral model and the likes (Sommerville, 2016). Currently in 2020, Himmat and Osman (2020) also indicated that traditional software development models are still dominating the software industry. As suggested by Olteanu (2018), organisations that have tacit knowledge about traditional methods resist changing their development processes. A. L. Marnewick and Marnewick (2019) have emphasised the impact of Industry 4.0 in various other industries. Although the foundation of the 4th Industrial revolution (4IR) is implemented using agile methodologies (C. Marnewick & Marnewick, 2019), the study does not focus on 4IR.

Several scholars (Himmat & Osman, 2020; Khmelevsky et al., 2017; C. Marnewick et al., 2017; Raunak & Binkley, 2017; Vallon et al., 2018) have indicated that agile methods are the most popular methods in use and followed by the traditional Waterfall method. Over again some studies (Boeije, 2002; Digital.ai, 2019; Joseph et al., 2016; Mudarikwa & Grace, 2018; Theocharis et al., 2015) have all identified Scrum as the most used agile method in South Africa followed by XP and Lean. Dikert et al. (2016) found similar results in their systematic literature review study. That means the Scrum method is used largely by organisations in Africa (Joseph et al., 2016; Mkoba & Marnewick, 2020; Mudarikwa & Grace, 2018; Nakigudde, 2019). For example, Standard Bank's IT group is located in Johannesburg headquarters, and has changed from a Waterfall model to an agile approach (Johnston & Gill, 2017). The group manages the bank's technology and infrastructure. According to Mkoba and Marnewick (2020) 75% of South Africa organisations use the Scrum method to manage and implement their projects. Hobbs and Petit (2017), Naslund and Kale (2020), and Lindskog and Netz (2021) share the same sentiment that Scrum is the most dominating agile method in the industry of project management.

Organisations are prone to combine the Waterfall and Agile methods (C. Marnewick & Marnewick, 2019). Also other customers who are familiar with Waterfall plans may find it difficult to engage in an agile planning process (Elmezain et al., 2021). In 2015, Vijayasarathy

and Butler (2015) reported that organisations are using more than one methodology on their projects and the traditional methodologies are still popular. Research by Theocharis et al. (2015) emphasises mixing both Waterfall and Scrum to reap the benefits. The choice of software development methodology depends on what the customer prefers and the environment where the software will be used, and also depends on the type of software under development (Elmezain et al., 2021). The FBI had evolved into software development and quickly adopted agile software development methods (Fulgham et al., 2011). Well-known companies and organisations like Google, Microsoft, and IBM have adopted agile methods in their projects (Olteanu, 2018). Just like other IT organisations, Accenture is also promoting Agile (Naslund & Kale, 2020). Safety and health critical software require lots of documentation and analysis before the coding can start, therefore most are developed using a Waterfall process instead of Agile (Elmezain et al., 2021).

According to Gulec et al. (2017), both the Waterfall and Agile approaches are usable although are not cost-efficient. However, the Waterfall model is more manageable than Agile, although Agile is more adjustable than Waterfall. Additionally, regarding agile methods, organisations have difficulty in deciding their appropriate agile method (Gandomani et al., 2013). Practitioners who support and follow the traditional approach believe that agile approaches are more chaotic and their lack of formal procedures affects project success (Lalic et al., 2022). Since each methodology has its disadvantages and advantages, each organisation should carefully choose an appropriate methodology for its projects (Chiyangwa & Mnkandla, 2018; Dikert et al., 2016; Gulec et al., 2017; Nakigudde, 2019). Large organisations have experienced similar challenges with agile adoptions (Dikert et al., 2016; Mudarikwa & Grace, 2018), which is recommended for small and medium projects while the traditional approach is recommended for large projects and larger teams (Gandomani et al., 2013). Furthermore, Agile projects include a small team of developers who are expected to deliver working software in a short period (Fulgham et al., 2011). The other industry that has experienced some challenges with agile methods adoption is the banking industry. The banks are highly regulated institutions, hence they need highly anticipated documentation, so by default, the industry uses a traditional approach (Beerbaum, 2021; Dewantari et al., 2022). According to Kamath (2019), agile methods make no sense in the banking industry because in this industry as a developer you need methodology documents that you need to check regularly and logs of meetings are always required, which is contrary to the Agile approach. The author has spent two decades in the banking industry; hence he recommends the Waterfall methodology of PRINCE2. According to Lalic et al. (2022), both agile and traditional project management approaches have their own challenges which are based on project characteristics, hence a hybrid project management approach was introduced to the software industry. This article does not focus on why some organisations are using which methodology. However, some researchers (C. Marnewick & Marnewick, 2019) have emphasised the significant role played by agile methods in software development projects. According to Nakigudde (2019), there is no specific model that warrants successful software development projects, since all have weaknesses and strengths. There are settings where agile methods do not show their strength

(Theocharis et al., 2015), therefore there is no single software development methodology that can replace Waterfall entirely (Shaydulin & Sybrandt, 2017). This article is not based on any specific software development method, as C. Marnewick et al. (2017) articulated that project success does not depend on the chosen method.

2.1 Software development project success

According to Burke (2011), if the question we ask about project success is "Was the project delivered on time, on budget, and to the required quality?" and if the answer is "yes" to all of these questions, then that means the project was a success from the project manager's perception and definition of the word project success. When project success is defined by the project sponsor's perspective, the connotation changes. The project sponsor defines project success in favour and the benefits for the company. The benefits are measured in terms of maintaining competitive advantage, increasing profits, an increase in sales, venture to an opportunity, or enhancing a brand image. Besides the three traditional constraints of software success called time, cost, and quality, few authors consider other constraints like risks (Burke, 2011; Lehtinen et al., 2014; Poranen, 2014). Comparatively, both Albert et al. (2017) and Lalic et al. (2022) emphasise that time, cost, and performance have already been criticised by a large number of researchers and practitioners. Regardless of the project approach, projects need planning, scheduling, and budgeting in order to succeed (Gandomani et al., 2013).

One of the reasons projects are declared failed is that the definition of "success" for software projects is different for different people and stakeholders (Albert et al., 2017; C. Marnewick et al., 2013; Poranen, 2014). Software development projects have three potential outcomes i.e. the project is successful, a failure, or it is regarded as challenged. According to Pretorius et al. (2012) the definition of "project success" depends on a person's perspective.

Therefore, the definition of project success will always be defined from different perspectives (Pozzi et al., 2021). According to the Prosperus Report (C. Marnewick et al., 2013), there is no dominating criterion when it comes to the definition of the word project success. That is possible considering that project success criteria differ from one project to another because of the diversity of projects. The authors of this article have adopted software project success as meeting the three traditional constraints of time, budget, and quality (scope), regarding scope as part of quality. The dimension of scope was incorporated in the dimension quality, the reason being that the required objectives and functionality are regarded as scope, but that is also part of a good quality product (quality dimension). The traditional success criteria using quadruple constraint are significant and that was assured by 45% of the respondents of the C. Marnewick et al. (2013) study. The project manager defines the project based on the traditional project success criteria and the sponsor looks at the benefits that the project can deliver. The focus of project managers is on delivering projects on time and within budget (Badewi, 2016). The customers want functional software that meets their software project metrics which they have agreed upon with the supplier with regards to cost, quality, and time. According to Hedeman et al. (2005), project success is when the stakeholders are satisfied with the results

of the delivered project. The criteria for software development projects success is perceived differently by several researchers and practitioners, also Khoza and Marnewick (2020) agrees with the opinion.

According to Camilleri (2016) the word success is misleading when applied to software projects because even the current successful project can be declared a failure in the future. The project scope can change at any time or new features can be added at a later stage. Projects that are completed on time, within budget, and according to the scope as defined by stakeholders of the organisation, are regarded as being successful (Hedeman et al., 2005; B. P. Kaur & Aggrawal, 2013). According to Humble and Russell (2009) the definition of the phrase "successful projects" is not enough if it does not include the concepts of completing the project on time and within the budget.

A study of the success rate of IT projects in South Africa was initially undertaken by Sonnekus and Labuschagne (2003) and thereafter taken over by C. Marnewick (2013). The study reports on IT projects in general, not restricted to software development projects. The success rates of IT projects in South Africa in 2003 (43%) and 2008 (37%) were not as impressive as that of 2011 (59%). The results of 2011 by C. Marnewick (2013) almost doubled the results of 2008 (from 37% to 59%). The success rate of 59%, and 29% were calculated as challenged, and the failure rate was 12%. The results generated from the study by C. Marnewick (2013) were derived from projects that were undertaken in both South Africa and the African continent at large.

A comparative analysis of the results of project success published by The Standish Group (the CHAOS Report) and those generated from the South African study have revealed that the local success rates are reporting more favourable results than those reported by The Standish Group. Another cross-analysis of project size and performance reported by Joseph et al. (2016) has reported a success rate of 69% for medium projects and 75% for small projects. They compared the 2015 CHAOS Report and Agile project performance, which reported 58% project success by Agile and only 39% success rate for the CHAOS Report. In this article, the software project success measures were used from the traditional three-constraint definition because most authors defined them as they wish. Thus the reason some practitioners like C. Marnewick et al. (2017) do not want to define the word project success anymore is that no one is sure about the definition of project success.

2.2 Critical Success Factors in Software development projects

This subsection explains the reasons why projects fail and analyses critical success factors across industries as compared to software development projects success factors. Each industry has its critical success factors that affect the delivery of their projects, although there are some which are common to all industries (Ahimbisibwe et al., 2015). Radujković and Sjekavica (2017) consider project success factors as enablers of project success. According to Nakigudde (2019) the critical success factors are the aspects that can lead to the successful completion of a project.

D. S. Nguyen (2016) in his research study has categorised agile software development project success factors into the process, people, technical, technologies and development tools. Dikert et al. (2016) are some of those who indicated the most number of the agile project critical success factors; their study has reported 35 factors and grouped them into 11 categories. Another study conducted by Kateb et al. (2015) has classified the critical success factors into the team, project management, organisation, and environmental factors. Thereafter, they categorised the critical success factors into the most important critical success factors according to CEOs and project managers. The top critical success factor of CEOs is clear, well-defined, and measurable project objectives, and according to the software project manager perspective, the top critical success factor was regarded as direct interaction with the client. According to Lehtinen et al. (2014) the common causes of software project failures include the environment, tasks, methods, and people. The project success and critical success factors need to be well defined to enable the project outcome to be fairly judged. Mcleod and Macdonell (2011) have monitored how software critical success factors that influence software development are perceived over a period of thirty years. The top ten factors that contribute to project success that has been reported by the Standish Group during the period 2012-2015 are listed in order of importance in Table 1 (Hastie & Wojewoda, 2015; Standish Group, 2013; Standish Group International, 2012).

Table 1: Success Factors from 2012 to 2015 [a]

Factors	**2012**	**2013**	**2015**
Executive support	1	1	1
User involvement	2	2	3
Clear business objectives	3	7	10
Emotional maturity	4	8	2
Optimisation	5	3	4
Agile process	6	6	7
Project management expertise	7	5	9
Skilled resources	8	4	5
Execution	9	9	8
Tools and infrastructure	10	10	–
Modest execution	–	–	6

[a]Hastie and Wojewoda (2015) and Standish Group (2013)

Since 2012, executive support tops the list as the most important factor contributing to the success of a project. The executive sponsor is the most important person involved with a

project and should have the skills to lead and guide a project to resolution (Standish Group, 2013). Also, it appears that successful projects have strong non-technical factors in terms of executive support and user involvement that may lead to clearly defined requirements and project objectives. Technology, tools, and methods play an important part but less influential role (Marchewka, 2016). Software developers and system analysts have acknowledged that user involvement is one of the crucial factors in the success of software projects (Ruhe & Wohlin, 2016).

An analysis of software project failures by Lehtinen et al. (2014) revealed that there is no single cause of software project failure. The analysis also revealed that a lack of software testing plays a central role in software project failure. The sequence of factors contributing to the success of a software development project has not changed dramatically in the past five years. Similarly, the executive sponsor factor is still on top of the list as the most important factor contributing to the success of projects. Other factors moved up and down their positions on the list as illustrated in Table 1.

The Standish Group's CHAOS Reports seem to be widely recognised in the industry because the Group's research started about 25 years ago and has reported more than 80,000 completed IT and software development projects (Standish Group International, 2012). Similar studies on IT and software development projects confirm the same critical success factors of The Standish Group International (Joseph & Marnewick, 2014; Nasir & Sahibuddin, 2011; Standish Group International, 2012).

Software projects are closing prematurely because of many different reasons. Factors behind the software project outcome results are reported every year. A review and classification of these factors can potentially contribute to or lead to a successful release of the desired software product. As shown in Table 2, the top four success factors listed by Joseph and Marnewick (2014) are completely different from those of The Standish Group (CHAOS Report). The consistent top four success factors reported by The Standish Group Report are executive support, user involvement, and clear business objectives as represented in Table 2 (Hastie & Wojewoda, 2015; Joseph & Marnewick, 2014).

Table 2: Comparison of Top 4 Success Factors

	CHAOS Report success factors (2010 to 2015) [a]	**ICT factors (2014)**[b]
1	Executive management support	Requirements definition clarity
2	User involvement	Communication between team and customers
3	Clear business objectives	Communication between project team members
4	Emotional maturity	Business objectives clarity

[a]Hastie and Wojewoda (2015)
[b]Joseph and Marnewick (2014)

The success factors provided by R. Kaur and Sengupta (2013) are also based on software development projects and are related to the factors provided by The Standish Group. The factor that is common to both studies is clear business objectives, although Nasir and Sahibuddin (2011) used the phrase "unclear goals" to emphasise failure cause. Both Nasir and Sahibuddin (2011) and Joseph and Marnewick (2014) have found defining requirements as their top critical success factor for the delivery of software development projects. This means that both clear requirements and executive management support are essential for projects to succeed.

It is clear from the literature review that the success factors that are reported by most researchers and practitioners have also been reported in The Standish Group Reports, although there are minor differences with the ranking of these factors. Kateb et al. (2015) also explained that some of these factors truly affect the delivery of successful projects. The project success and failure factors apply to all project environments, irrespective of whether a project is being conducted in a multi-project environment or an international setting (Camilleri, 2016).

The industry practitioners report different categories of project factors and a different number of critical success factors. Other factors have emerged during a recent case study conducted by Dakkak et al. (2021) about the continuous deployment of cloud-based software development organisations. The authors have divided the critical success factors categories into Customer, Technical, and Organisation. Authors like Zahran (1999) have recommended only three main categories of software development project success critical factors, which are people, process, and technology, hence the study has grouped the critical success factors into three. Critical success factors identified in other types of projects are not valid critical success factors for software development projects (Nasir & Sahibuddin, 2011).

Software development factors influence the project's outcome. If those factors are managed adequately, then the software projects are unlikely to fail. From the above discussed critical success factors, the study recommends the categories of critical success factors by Nasir and Sahibuddin (2011) outlined in the following subsections.

2.2.1 People factors

Human factors were linked to the success of a project and hence the literature about people-related factors where reviewed. People aspects describe the human-related factors that influence the project outcome such as team members' skills. According to Dakkak et al. (2021) the factors like top management support, skilled staff, and user involvement are organisational related factors. The role of top management members such as chief executives were periodically involved in the project review activities (Pozzi et al., 2021). The factors which will be addressed consist of:

(a) Effective project management skills/methodologies (project manager)

(b) User/client involvement

(c) Support from top management

(d) Good leadership

(e) Committed and motivated team

(f) Good performance by vendors/contractors/consultants

(g) Skilled and sufficient staff

2.2.2 Process factors

According to D. Nguyen (2016), process-related factors are aspects that are linked with the functions or processes of the project, e.g. the review of the code of the software or testing the software code. Project process describes how well the software development process has been undertaken, it measures how a project is delivered on time, within budget, and following agreed scope (Ahimbisibwe et al., 2015). The process factors are the factors that link human-related factors and technical factors, if these processes are not taken care of, then human weaknesses will be exposed. Nasir and Sahibuddin (2011) from the literature of IT and software projects found that process factors consist of:

(a) Clear requirements and specifications

(b) Clear objectives and goals

(c) Realistic schedule

(d) Realistic budget

(e) Frozen requirement

(f) Effective communication and feedback

(g) Proper planning

(h) Appropriate development processes/methodologies (process)

(i) End-user training provision

(j) Up-to-date progress reporting

(k) Adequate resources

(l) Effective monitoring and control

(m) Risk management

(n) Clear assignment of roles and responsibilities

(o) Effective change and configuration management

(p) Good quality management

2.2.3 Technical factors

The addition of the technical factors category to software development success was recommended by Sudhakar (2016) because software projects are affected by response times, speed of the device, troubleshooting, etc. Technical factors are those aspects that have an effect on how a project functions and are mostly linked to the software, hardware, and technology employed within the project development process (D. Nguyen, 2016). Technical factors mostly address problems that are associated with technology, software quality, and tools. Nasir and Sahibuddin (2011) found that technical success factors consist of:

(a) Supporting tools and good infrastructure

(b) Familiar with technology/development methodology

(c) Complexity, project size, duration, and number of organisations involved

2.2.4 Summary of critical success factors

The reason different studies have variations of critical success factors is because different stakeholders have their own priorities and interests that may impact project success (Ahimbisibwe et al., 2015). In this article, the critical success factors used were adopted from the comparative study of Nasir and Sahibuddin (2011), because both researchers have covered the most possible list of software development industry factors. Nevertheless, the software project success criteria adopted were not defined by Nasir and Sahibuddin (2011)'s comparative study. The three categories of factors used in this study to determine the influence on software development project success were as identified by Nasir and Sahibuddin (2011).

Nasir and Sahibuddin have covered almost every possible critical success factor of the software development project. The critical success factors of Nasir and Sahibuddin are relevant to software development projects and they have a good frequency of appearing in the literature. Most of their critical success factors are identified in Sudhakar (2016) as top factors of each category. Nasir and Sahibuddin (2011) found twenty-six factors that are critical to the success of software projects and classified seven of these factors as people-related factors in their comparative study. Of the twenty-six factors, Nasir and Sahibuddin (2011) further classified 16 factors as process-related factors and only three are technical related factors. Nasir and Sahibuddin (2011) have categorised the critical success factors specifically for any software development team and software organisation. Since then, several other studies (Da Silva & Dos Santos, 2017; Fayaz et al., 2017; Ibrahim et al., 2013; Kabir & Rusu, 2016; Kateb et al., 2015; Sudhakar, 2016) on software critical success factors have indicated most of Nasir and Sahibuddin's 26 critical success factors in their investigations. The factors for each category sourced from Nasir and Sahibuddin (2011) are presented in Table 3 (People (with seven factors), Process (with 16 factors), and Technical (with three factors)).

Those 26 are not the only critical success factors that affect software development projects, however, they appear at the top of many other studies like in Chiyangwa and Mnkandla (2017).

Table 3: Success factors for software development projects [a]

Category name		Factor intention
People related factors	1	Committed and motivated team
	2	User/client involvement
	3	Good leadership
	4	Skilled and sufficient staff
	5	Support from top management
	6	Effective project management skills/methodologies (project manager)
	7	Good performance by vendors/contractors/consultants
Process related factors	1	Clear requirements and specifications
	2	Clear objectives and goals
	3	Proper planning
	4	Effective communication and feedback
	5	Clear assignment of roles and responsibilities
	6	Good quality management
	7	Appropriate development processes/methodologies (process)
	8	Adequate resources
	9	Realistic budget
	10	Effective change and configuration management
	11	Realistic schedule
	12	Up-to-date progress reporting
	13	Risk management
	14	Effective monitoring and control
	15	End-user training provision
	16	Frozen requirement
Technical related factors	1	Supporting tools and good infrastructure
	2	Familiar with technology/development methodology
	3	Complexity, project size, duration, and number of organisations involved

[a]Nasir and Sahibuddin (2011)

The critical success factors used in this article are not management or organisational issues, but only the process, people, and technical. In the study of software development by Guzmán et al. (2011) the critical success factors are categorised into human, process, and technology factors. That is the same for this study because they have used the word human and the study has used the word people. The 26 critical success factors are classified into the same three categories as da Silva and Santos (Da Silva & Dos Santos, 2017; Zahran, 1999).

The scope of the study is within South Africa. The main objective of this research article is to determine the perfect success factors which are critical to the delivery of a successful software development project.

A successful project can be realised if there is more concern for people, processes, and technical factors within the software development organisation. Project success was defined using three success dimensions namely budget, time, and quality. The study proposed that project success is directly or indirectly influenced by the perceived people, technical and process factors. The constructs used for this study are people with seven factors, process with sixteen factors, and technical with only three factors as represented in the below Figure 1. The figure represents the proposed conceptual framework which on the left-hand present three independent variables and on the right hand the research uses software development project success as a dependent variable.

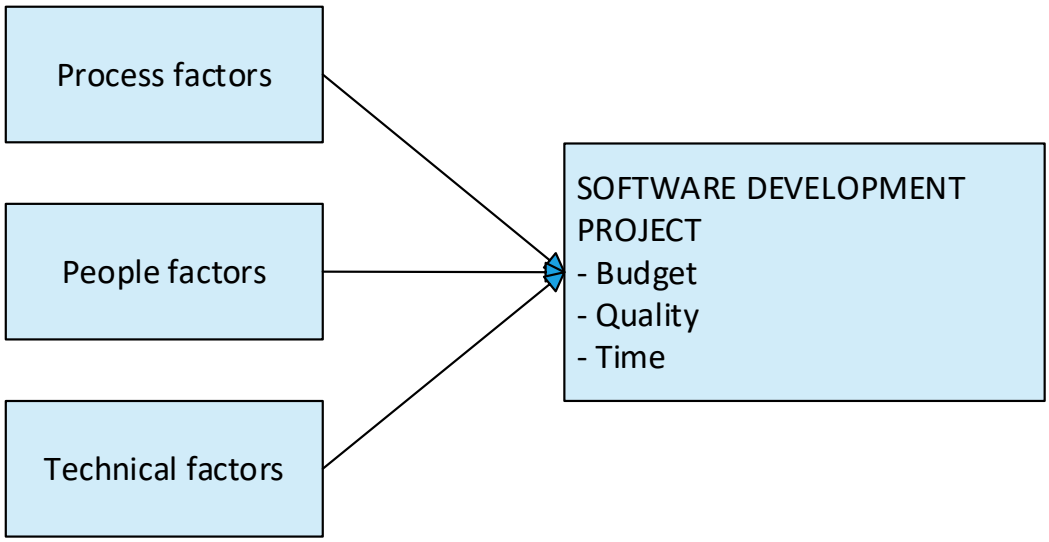

Figure 1: The research model

The three categories of factors used in Figure 1 represent 26 critical success factors, and the perceived project outcome is defined by the budget allocated, the quality or scope of the end product, and the time allotted.

https://doi.org/10.18489/sacj.v34i1.820

3 RESEARCH METHODOLOGY

The approach used in this study is positivist and the quantitative method was adopted, although different researchers have different ideas about research approaches. Sekaran and Bougie (2016) believe that the researcher should be guided by the research scientific approach to getting the truth about the subject under investigation. The study employs the positivism research approach, which means the results of this research article will be generalised to a larger population (Gordon, 2019). In applying the positivism research approach, deductive reasoning will be employed; the study will start with ideas in the form of theories and obtain data to test the theories designed (Gordon, 2019). The goal of the research is to describe the observable claim and test the predetermined research design. According to Gordon (2019), a survey study is centred on positivism methodology, which means if the procedures for sampling, data collection, and analysing can be performed by another researcher, then the same results will be achieved. Most researchers substitute the word positivism for quantitative research (Biggam, 2015). Positivism bases its facts on available evidence (Naoum, 2013; Saunders et al., 2012), hence this article will collect data to support the claims made.

A structured questionnaire was designed to collect the study data. The study adhered to ethical values prescribed by our university to defend the integrity of the research. The University research ethics committee assessed the research instrument used and provided ethical clearance before the questionnaire was distributed to the respondents. According to Walliman (2017) the research has value if it is carried out honestly. No incentives were promised to complete and submit the questionnaire.

The opinions of the software development team members are on a five-point Likert scale as designed by the positivist researchers (Remenyi & Bannister, 2012). The target sample was the members of a software development team. The software industry of South Africa was chosen as a sample. A questionnaire constructed was used to discover the success rate and critical success factors perceived by members of the software development team. The first section of the questionnaire was about collecting personal information, and the second section was about the factors critical to a project's success, software development project, outcome and factors that determine the outcome of the software project. The second section of the questionnaire was divided into two scales.

The questionnaire used in this study used two Likert scales. The first scale used frequency, where the respondents were asked to rate the outcome of their projects from 1–5; 1 is never while 5 is every time. The scale was designed to define the level of project outcome, denoting the software development project success. Respondents were asked to: *Please rate the following in terms of frequency (Rate from 1–5; 1 is Never, while 5 is Every time): On average, please rate your projects in the last 6 months.* For example, the first question about project outcome was about the budget/cost, and the rest of the questions were asked as in Table 4.

Questions related to the duration of the project were under the time dimension which was linked to question 2 of the project outcome. Similarly, question 3 to 5 was about quality, and the question about customer satisfaction was part of the quality questions.

Table 4: How questions were asked in this study

Budget	1	The project was completed within or below budget
Time	2	The project was completed on time or earlier
Quality	3	The product met the customer's specifications
	4	The customer is using the delivered product
	5	The customer was satisfied with the project

Then on the second scale, the respondents were asked Likert-type questions, where they rate their agreement with the factors critical to the project's success. In the same section, 26 ordinal questions were divided into three themes which are process, people, and technical factors. The questions about critical success factors were adopted from Nasir and Sahibuddin (2011). The Likert scale questions were used to measure the level of criticality of factors towards software development projects. The scale was from 1–5; 1 represents strongly disagree, while 5 represents strongly agree. The items used for determining the people, process, and technical factors were collected in the literature review study of Nasir and Sahibuddin (2011).

Table 5 represents a sample of how software development critical success factors were asked. The question was: *To what extent do you agree that the following project success factors are critical for the success of your software development projects? (Rate from 1–5; 1 is Strongly Disagree, while 5 is Strongly Agree)*.

Before data was collected from the entire sample of IT professionals, the questionnaire was tested by 10 participants from the software development industry. The changes were noted, and the questionnaire was updated accordingly. Thereafter, the email was sent to respondents of the study, who were selected based on their positions e.g. data scientists, business analysts, project managers, etc.

A total of 750 respondents were invited and reminded after a week to complete the questionnaire. Nevertheless, only 480 agreed to participate in the study. About 55% of the respondents were contacted via social media (i.e. LinkedIn). The sample size for the study was therefore 212 in total. An overall response rate of 28% was recorded meaning that out of 750 invited respondents, only 212 members of a software development project team completed the questionnaire. Regarding the population of the study, the target population is represented in Table 6. The respondents are assigned different roles. Just to highlight some insights, 78 out of 212 study respondents have more than 10 years of experience. 52% of the respondents are software developers; then 31 which is 14% have chosen option other, business analysts are on 8.5% followed by project managers (7.1%) and data scientists (6.1%). Out of the 31 respondents who chose option other, only three specified their specific positions with the software development team, and 28 respondents have not named their other positions, but they have chosen length of experience. IT industry is one of the fields that have more types of

Table 5: How software development critical success factors were asked

	Category	Software Development Factors	Strongly disagree	Disagree	Neutral	Agree	Strongly agree
1	People factors	User client involvement	☐	☐	☐	☐	☐
		Support from top management	☐	☐	☐	☐	☐
		Good leadership	☐	☐	☐	☐	☐
2	Process factors	Clear requirements and specifications	☐	☐	☐	☐	☐
		Clear objectives and goals	☐	☐	☐	☐	☐
		Realistic schedule	☐	☐	☐	☐	☐
3	Technical factors	Supporting tools and good infrastructure	☐	☐	☐	☐	☐
		Familiar with technology/development methodology	☐	☐	☐	☐	☐

employment positions, e.g. Scrum Master.

3.1 Reliability and validity

The possible objections to the validity of the instrument used are discussed in this section. According to Kumar (2011) the meaning of validity is the ability of a research instrument to measure what it is designed to measure. Reliability is a measure that indicates that the survey questionnaire used is error-free when used by different researchers and ensures consistent measurement throughout the research time.

3.1.1 Content validity

The questionnaire was designed to collect data. The researchers had to rely on expert advice from the software industry to guide them and determine if the designed questionnaire tests what it is supposed to test (Goddard & Melville, 2001). The metrics used to define the project success and the critical success factors chosen to characterise the data set were valid. We defined success as a combination of budget, time, and quality. The researchers further validated the research instrument by distributing it to 10 members of the software development team who participated in the pilot study. The respondents were asked to specify the time required to complete the survey and provide any comments, and the recommendations were implemented.

Table 6: Cross-tabulation of position and work experience

Position	**N**	**%**
Senior manager	11	5.2
Software Developer	111	52.4
Project Manager	15	7.1
Software Architect	4	1.9
Business Analyst	18	8.5
Quality Assurer	5	2.4
Project Administer	4	1.9
Data Scientist	13	6.1
Other:	28	13.2
Senior DBA	1	0.005
Programme Manager	1	0.005
Database Administrator	1	0.005
Length of experience in number of years		
< 1	18	8.5
1–5	63	29.7
5–10	53	25.0
10–15	45	21.2
15–20	14	6.6
> 20	16	7.55
prefer not to say	3	1.4
Total	**212**	**100**

3.1.2 Reliability

The reliability test ensures that when the research instrument is used by other researchers, it will produce the same results (Sekaran & Bougie, 2016). Currently, there is no complete research instrument that can be used to determine the critical success factors that determine the software development project outcome. Though the researchers adopted the list of critical success factors from the literature study of Nasir and Sahibuddin (2011), that was not a complete questionnaire that is ready-made. The most popular used data analysis is descriptive statistics, which cover frequency, descriptive, and cross tabs functions (Devlin, 2006).

The closer the Cronbach's alpha is to 1, the higher the internal consistency reliability (Sekaran & Bougie, 2016). The Cronbach's alpha values of less than 0.6 are considered very poor; any reliability value that is higher than 0.75 is good. The reliability value of 0.8 is acceptable in most cases, but reliability values of less than 0.6 are not acceptable (D. Nguyen, 2016). In this article, the measurement scales of validity and reliability were determined by calculating the values of Cronbach's alpha using SPSS 25.

The Cronbach's alpha for all four items was calculated, and the scales are reliable and

rational. The Cronbach's alpha coefficient for all four constructs is represented in Table 7.

Table 7: Reliability statistics

Name of variable	Cronbach's alpha	Number of items
Project outcome	.840	5
People related critical factors	.872	7
Process related critical factors	.941	16
Technical related critical factors	.747	3

The Cronbach's alpha for all four constructs were close to 1, implying that the questionnaire has internal consistency. Cronbach's alpha of project outcome is 0.840 and for the software development project critical success factors of people, process, and technical were 0.872, 0.941, and 0.747. The total number of items used was 31. This shows and confirms reasonable consistency reliability.

3.2 Data analysis

This study was descriptive and simple techniques of exploratory data analysis were used (Oates, 2012). The right statistic for ordinal values was chosen for the study. Statistical techniques such as correlation analysis were used to explore the relationships among variables and manipulate data (Kerr et al., 2009). SPSS 25 was used for quantitative data analysis together with Microsoft Excel 2016. This section presents the findings of quantitative research. The data set was processed to achieve the research objectives. The questionnaire was composed of a total of 26 software development factors. The structure of factors was divided into three categories namely: people, processes, and technical.

The selected factors of this research study apply to any software development project. The software development critical success factors by order of criticality for each category are presented in Table 8, and the top 10 factors which have the highest means have been highlighted for each category.

Our findings discovered that several factors belong to the people factor category, although process-related factors dominate more than 50% of the overall list of critical success factors. The top 10 critical success factors that influence the project's success are discussed in more detail in Section 4. Out of a total of 212 respondents, only 190 completed the section on critical success factors in the questionnaire. Certain critical success factors received a low rating. Frozen requirements are a process-related factor that has received a low rating of (mean = 3.38; SD = 1.035) out of 26 as shown in Table 8. The respective mean values obtained for each construct are indicated in parentheses as follows: people (4.34); process (4.10); technical (4.16) and the software development project outcome was 3.83. Only 3 of the 26 factors received a rating of less than 4, namely: effective monitoring and control, end-user training provision, and frozen requirements, as shown in Table 8.

Table 8: Critical success factors for software development projects

Category	Rank	Critical success factor	Mean	Std Dev	N
People related factors	1	Committed and motivated team	4.52	0.732	191
	2	User/client involvement	4.41	0.816	190
	3	Good leadership	4.35	0.789	189
	4	Skilled and sufficient staff	4.33	0.793	188
	5	Support from top management	4.31	0.811	190
	6	Effective project management skills/methodologies (project manager)	4.25	0.827	191
	7	Good performance by vendors/contractors/consultants	4.22	0.834	189
Process related factors	1	Clear requirements and specifications	4.37	0.868	190
	2	Clear objectives and goals	4.36	0.808	191
	3	Proper planning	4.29	0.939	189
	4	Effective communication and feedback	4.28	0.874	190
	5	Clear assignment of roles and responsibilities	4.19	0.820	190
	6	Good quality management	4.18	0.787	188
	7	Appropriate development processes/methodologies (process)	4.15	0.818	190
	8	Adequate resources	4.14	0.876	189
	9	Realistic budget	4.10	0.887	187
	10	Effective change and configuration management	4.09	0.830	190
	11	Realistic schedule	4.07	0.884	191
	12	Up-to-date progress reporting	4.04	0.857	191
	13	Risk management	4.03	0.925	190
	14	Effective monitoring and control	3.99	0.805	191
	15	End-user training provision	3.96	0.913	189
	16	Frozen requirement	3.38	1.035	188
Technical related factors	1	Supporting tools and good infrastructure	4.27	0.746	190
	2	Familiar with technology/development methodology	4.16	0.799	189
	3	Complexity, project size, duration, and number of organisations involved	4.06	0.770	187

3.2.1 The correlation analysis

Correlation analysis is a measure of both the strength and direction of the relationship between two variables (Field, 2013; Pallant, 2010). The correlation analysis measures the size and direction of the relationship between the software project outcome and the critical success factors. The correlation analysis will help to determine whether the independent variable is associated with the dependent variable or not (Longest, 2019). The coefficient is known as the Pearson correlation coefficient (or Pearson product-moment correlation coefficient), and according to Field (2013) the coefficient was discovered by Karl Pearson. According to Lipschutz and Schiller (2012) Pearson's correlation coefficient r is always between minus 1.00 and plus 1.00.

According to Pallant (2010), several authors have suggested their interpretation of the

correlation coefficient, hence he suggested that correlation coefficient values close to zero indicate a weak relationship and values close to 1 indicate a strong association. The correlation coefficient of minus 1 signifies a perfect negative relationship, a correlation coefficient of plus 1 represents a perfect positive relationship, while a correlation coefficient near 0 indicates no relationship at all (Field, 2013; Levine, 2010). The strength of the correlation between +0.3 and -0.3 is the same, the difference is only the direction (Pallant, 2010).

A correlation was performed between project outcome and critical success factors. The constructs of three categories of factors (people, process, and technical) were used to calculate the Pearson correlation coefficient against project outcome. The project outcome (success) constructs were made out of a total of five items, namely budget, time, and quality (quality consisted of three measures, which are customer specification, delivered product, and satisfaction with the project). The constructs used for correlation analysis between critical success factors and project outcomes for this article are illustrated in Table 9.

Table 9: Correlation between critical success factor constructs and project outcome constructs

		Project outcome		
	Construct	**Pearson Correlation**	**Sig (2-tailed)**	**N**
Software Development Factors (Critical success factors)	People	.235[a]	0.002	173
	Process	.275[a]	0.000	158
	Technical	.219[a]	0.004	175

[a]Correlation is significant at the 0.01 level (2-tailed)

The Pearson correlation coefficients for people, process, and technical constructs are 0.235, 0.275, and 0.219 respectively. Since according to Field (2013) the values of ± 0.1 represent a small effect, and values closer to ± 0.3 represent a medium effect, similarly the values between ± 0.2 and ± 0.3 will be categorised as a small-medium effect. Then according to Pallant (2010), this represents a small strength of the relationship. Table 9 figures represent that there is no significant correlation to determine that either variable influences the project outcome, although process-related factors are better than the two other variables.

Now let us interpret the significance level as listed as Sig (2-tailed) in Table 9, which represents the level of confidence we should have with our results (Pallant, 2010). The significance is strongly influenced by the population size, with a large size sample of N > 100, very small correlations of $r = 0.2$ may reach statistical significance at a $p < 0.05$ (Pallant, 2010). The correlation results revealed that all three constructs have a significant correlation $(p < 0.01)$, people (0.002), process (0.000) and technical (0,004). As indicated in Table 9, there is a significant correlation between people, process, technical, and project outcome factors since their significant values are less than 1 percent (0.05). Overall, the Pearson correlation coefficient of all three constructs shows a weak positive correlation and that was confirmed by a p-value of $p < 0.05$.

4 DISCUSSION OF RESULTS

The results of this study are now presented and discussed in this section.

4.1 Perceived influence of project success factors

The top 10 critical success factors that influence project success are illustrated in Figure 2, by the way of a 100% stacked column chart, which shows the relative contribution percentage of each critical success factor in stacked columns against the total or cumulative stacked columns of 100%. The contribution percentage of the first two options (strongly disagree and disagree)

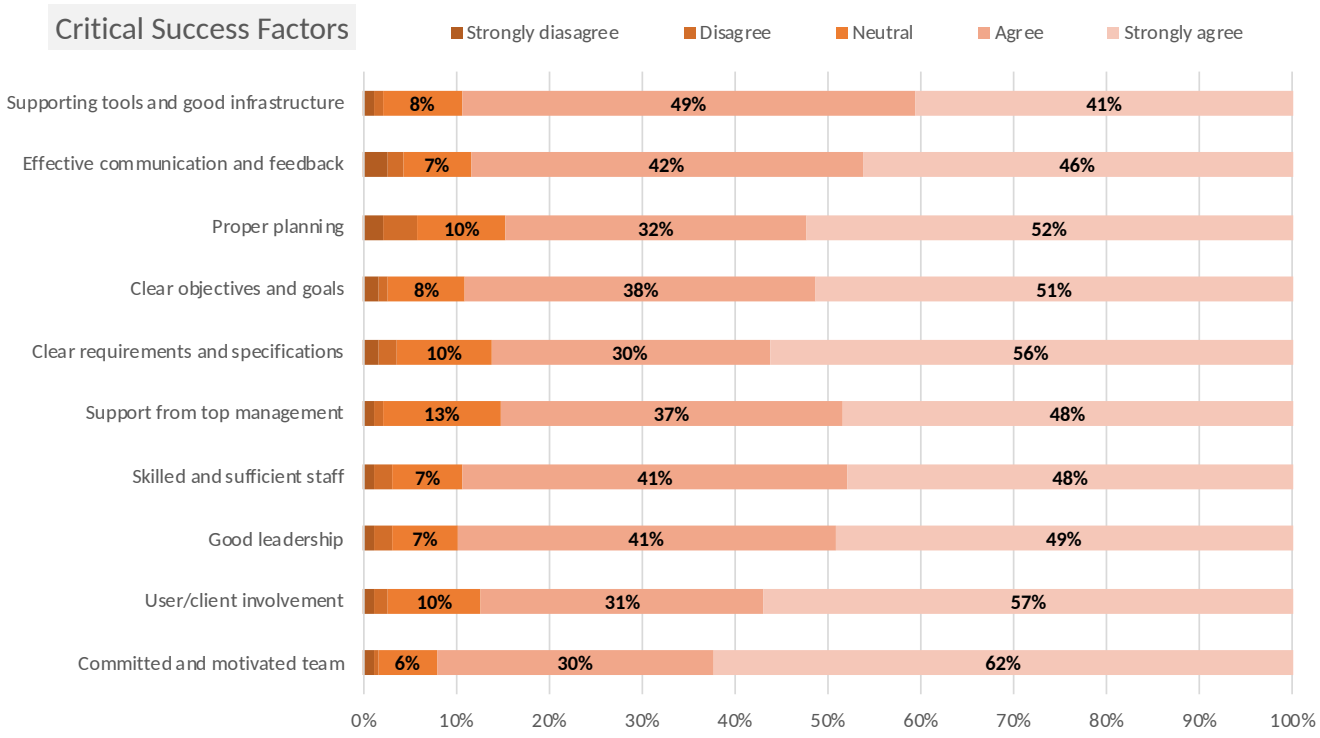

Figure 2: Breakdown of the top ten critical success factors

is not shown on the 100% stacked bar chart in Figure 2, since both options contribute less than 6% on each factor and are displayed on the left side of the chart. Out of a total of 212 respondents, only 190 completed the section on critical success factors in the questionnaire. Other critical success factors received a low rating, e.g. frozen requirements (mean = 3.38; SD = 1.035). Only 3 of the 26 factors received a rating of less than four, namely: effective monitoring and control, end-user training provision, and frozen requirements, as shown in Figure 2.

The top five factors that are regarded as more critical to the success of software development projects are: committed and motivated team; user/client involvement; clear require-

ments and specifications; clear objectives and goals; and good leadership. The highest mean score obtained for the success factor (committed and motivated team) was 4.52; this means that the respondents agree that the committed and motivated team factor is critical for the success of the software development project.

Table 10 details the comparison between the study's top 10 critical success factors and other dominating studies. The critical success factors reported in Table 10 from the study, also appear at the top of the list of other studies (Hastie & Wojewoda, 2015; Joseph & Marnewick, 2014; C. Marnewick et al., 2013; Nasir & Sahibuddin, 2011).

Table 10: Ranking of Factors contributing to Software Development projects success

	Researcher observations	**Nasir & Sahibuddin**[a]	**CHAOS Report**[b]	**Prosperus Report**[c]
1	Committed & motivated team	Clear requirements & specifications	Executive support	Requirements definition clarity
2	User/client involvement	Clear objectives & goals	Emotional maturity	Communication between team & customers
3	Clear requirements & specifications	Realistic schedule	User involvement	Communication between project team members
4	Clear objectives & goals	Effective project management skills/methodologies	Optimisation	Business objectives clarity
5	Good leadership	Support from top management	Skilled resources	Understanding of users' needs
6	Skilled & sufficient staff	User/client involvement	Standard architecture	Project manager competency
7	Support from top management	Effective communication & feedback	Agile Process	Executive support
8	Proper planning	Realistic budget	Modest execution	Handling change
9	Effective communication & feedback	Skilled & sufficient staff	Project management expertise	User involvement
10	Supporting tools & good infrastructure	Frozen requirement	Clear business objectives	Change control processes

[a]Nasir and Sahibuddin (2011)
[b]Hastie and Wojewoda (2015)
[c]Joseph and Marnewick (2014) and C. Marnewick et al. (2013)

Most of the top 10 critical success factors are found in the people category, which means people are more important in software development projects. The findings indicated that for the software development project to be successful, the software development team should

be committed to their project and stay motivated. The CHAOS Report also listed some of the critical success factors that fall under the people category as executive management, user involvement, skilled resources, and emotional maturity. Similarly, Joseph and Marnewick (2014) have close to five critical success factors that fall under the people's category as user involvement, executive support, project manager, communication of team, customers, and project team. Our study suggests that communication is not the success factor that has a strong influence as compared to Hastie and Wojewoda (2015), Joseph and Marnewick (2014), C. Marnewick et al. (2013) and Nasir and Sahibuddin (2011), where communication is regarded as the factor with a strong influence on the success of software development projects.

Software project managers are obligated to lead and direct a software development project team to produce a successful software development project, hence they must be aware of the most critical success factors of their industry because those factors need more focus and should be addressed before the project start. Furthermore, software development projects have several stakeholders who are also decision-makers at some point and they are keen to know the factors that lead their project to fail since it is well known that IT-related projects are failing at a high rate.

4.2 State of software project success

Software development project performance is measured differently as compared to the projects of other industries. The performance results generated in this quantitative study were compared with the results of other longitudinal and cross-sectional research in the academic environment. The Cronbach alpha coefficient for this subset is 0.956, therefore the analysis is valid. Table 11 presents the performance figures observed by this study.

Table 11: Project performance results

Project Performance Metrics	**Measures**	**Mean**	**Results**
Budget	The project was completed within or below budget	3.57	71.4%
Quality/scope	The product met the customer's specifications	3.94	78.8%
	The customer is using the delivered product	4.23	84.6%
	The customer was satisfied with the project	4.07	81%
Time	The project was completed on time or earlier	3.33	66%

Although Agarwal and Rathod (2006) are certain that success is very rare in software projects, the software development organisations have managed to improve the project's success in this article. The interpretation of project success in this article was based on three main traditional project success measures, namely: budget, quality, and time taken to deliver the

software project (see Table 11). Furthermore, the quality metric was divided into three measures, which are product specifications, customer satisfaction, and the use of the delivered software by the customer. Time was used to measure the duration to develop and deliver software developed and correlates with other measures such as software size. Therefore, to measure the success rate of the software development projects, the respondents were asked to rate the previous project outcomes on average based on three success dimensions from 1(one) to 5 (five), whereby 1 is never and 5 represents every time. The type of Likert scale used was measured in frequency of use. Therefore, if the same success dimensions were rated never or almost never, then that project is regarded as a failure. The projects must not exceed the time or budget to complete, and must be accepted by the user; that is our definition of software project success in this article. According to the results displayed in Table 11, the projects are occasionally completed within their original budget and they met the quality specified before the start of the project almost every time. This means that the organisations are measuring the cost of developing software. The projects are sometimes completed in time since the time success metric was rated 3.33 out of 5. This good quality contributes positively to the success of software development projects. The Likert scale of 1 and 2 obtained for success dimension time, needs more attention because such projects often experience challenges.

The quality project performance metrics measures of "is a customer using the delivered product", "the customer was satisfied with the project", and "the product met the customer's specification", were assessed and the respective mean values of 4.23, 4.07, and 3.94 were obtained against a 5 point Likert scale. The average mean of quality metric was found to be 4.08 out of 5 after combining its 3 measures. Using the same Likert scale, the project was completed within or below budget (mean is 3.57) and on time (mean is 3.33). However, it is noteworthy that the organisations performed poorly on the "the project was completed on time or earlier" indicator. Therefore, when the traditional project performance metrics budget, quality, and time were used to define the project success, the respective average mean values of 3.57, 4.08, and 3.33 were obtained against a 5-point Likert scale. Hence the overall perceived performance of 3.886, represents a significant improvement of 77.7%.

The project success rate of IT projects in Africa is sitting at 59% (C. Marnewick et al., 2013). The metric that has lowered the overall performance of the study is the time performance measure, which has recorded 66%, and budget has recorded 71%, then similarly quality-scope has recorded 81.5%. A 77% success rate reported in the current study constitutes a significant increase in comparison to the 69% reported by PMI (2017) when the same metrics (budget, quality, and time) were used to define the project's success. Table 12 is displaying the figures of other related studies as compared to this study.

Table 12: Overall software development projects success rates

PMI (2017)	C. Marnewick et al. (2017)	Khoza and Marnewick (2020)	This study
69%	63%	64%	77.7%

The freshly released study of Tam et al. (2020) also uses iron triangle metrics of time, budget, and quality to measure the project success, which are the same metrics as PMI (2017). The results of the study are above all three figures used in Table 12, but overall, the software development project success is steadily improving as compared to 10 years back of the figures of the CHAOS Report and the Prosperus Report. The study registered the highest success rate, but that is not more than 10% of the figures reported by the PMI which has recorded 69%.

5 CONCLUSION

The main aim of this research paper was to identify the critical success factors that have a direct influence on the delivery of software development projects. The researcher has used only three variables (budget, cost, and quality) to define project success, although there are several ways to define project success. The researchers have discovered that the project success rate has improved and can be further improved by the organisations developing software projects if they concentrate on the identified software project factors. The improved project outcome means that the organisations developing software in South Africa are delivering the project on time and within the estimated initial cost and with expected quality as compared to other previous figures. Therefore, there is no need to outsource and offshore from South Africa to European IT companies. The project team members are responsible for the success and failure of the project. Project success must be defined before the start of the project. This article has discovered that software projects require highly committed and motivated team members to be a success.

When the software development organisation delivers good quality of work to its customers, the word gets around and shortly more potential clients become interested to do business with them. Then the business grows. The major factors that are important to the outcome of the projects are a committed and motivated team, and user/client involvement, both belonging to people-related factors categories. The user or customer involvement was found by the studies of agile software projects to be the top critical success factor and was also found to be the 2nd most important critical success factor for 2012 and 2013 in the CHAOS reports. Further work will be done by repeating the study in other neighbouring countries of South Africa. Also, the research instrument can be refined and improved to cover other definitions of the word software project success and critical success factors of other categories like organisational.

5.1 Contribution of the research study

This research contributes to the current debate about the processes that may lead a project to fail. The software industry will now focus on the most critical success factors identified by the study. The aim of understanding the major critical success factors that contribute to the delivery of successful software was achieved. The study has made a significant contribution that will assist organisations to control the top 10 critical factors during the software development phase. The literature has revealed that the definition of project success will always

differ. The members of the software development team can benefit from the most critical success factors discovered by this article and the categorisation of factors by concentrating on them. This article contributes to the world and industry debate on critical success factors that contribute toward the delivery of software development projects. Furthermore, software project managers will now know that it is very important to have or assemble a committed software development team and they must make sure that the team remains motivated for the success of the software development project.

This article has revealed the software development project factors that need more attention from IT project managers. It also helps other stakeholders like IT organisation management and project funders, and customers who are familiar with the software development process to focus on the discovered factors when embarking on software development projects. Other scholars who are doing comparative analysis studies of critical success factors of the software development project can use the findings and compare them with longitudinal studies of the Standish and Prosperus Reports used by international and local communities of research in IT.

5.2 Limitation of the study

The study was conducted in South Africa only. The findings are relevant to the countries that develop software projects, not the countries that outsource or offshore software projects. Other limitations could be the way practitioners define the software development project's success. The researchers differ with some of the practitioners when it comes to the definition of the word project success in software development. The data set was more full of software developers than any other positions like software testers and project managers. But that was not the case since most members of software development projects are programmers/ software developers. The reporting was not based on either traditional software or agile development methods. This article has addressed the software development projects' critical success factors only, not other industry projects. Furthermore, the categories of critical factors were limited to people, process, and technical factors.

References

Agarwal, N. & Rathod, U. (2006). Defining 'success' for software projects: An exploratory revelation. *International Journal of Project Management*, *24*(4), 358–370. https://doi.org/10.1016/j.ijproman.2005.11.009

Ahimbisibwe, A., Cavana, R. Y. & Daellenbach, U. (2015). A contingency fit model of critical success factors for software development projects: A comparison of agile and traditional plan-based methodologies. *Journal of Enterprise Information Management*, *28*(1), 7–33. https://doi.org/10.1108/JEIM-08-2013-0060

Ahmed, A. (2012). Software Project Templates. *Software Project Management* (pp. 315–317). Boca Raton: CRC Press. https://www.taylorfrancis.com/books/9781439846568

Albert, M., Balve, P. & Spang, K. (2017). Evaluation of project success: A structured literature review. *International Journal of Managing Projects in Business*, *10*(4), 796–821. https://doi.org/10.1108/IJMPB-01-2017-0004

Alkraiji, A. I., Jayawickrama, U., Olan, F., Asaduzzaman, M., Subasinghage, M. & Gallage, S. (2022). The perspective of national ERP vendors in achieving ERP project success in government organisations: A case of Saudi Arabia. *Enterprise Information Systems*, *16*(1), 71–104. https://doi.org/10.1080/17517575.2020.1845811

Almeida, F. (2017). Challenges in Migration from Waterfall to Agile Environments. *World Journal of Computer Application and Technology*, *5*(3), 39–49. https://doi.org/10.13189/wjcat.2017.050302

Badewi, A. (2016). The impact of project management (PM) and benefits management (BM) practices on project success: Towards developing a project benefits governance framework. *International Journal of Project Management*, *34*(4), 761–778. https://doi.org/10.1016/j.ijproman.2015.05.005

Beerbaum, D. O. (2021). Applying Agile Methodology to Regulatory Compliance Projects in the Financial Industry: A Case Study Research. https://dx.doi.org/10.2139/ssrn.3834205

Biggam, J. (2015). *Succeeding with your master's dissertation : A step-by-step handbook*. McGraw Hill/Open University Press.

Bitla, K. S. & Veesamsetty, S. S. (2019). *Measuring Process Flow using Metrics in Agile Software Development* (Master's thesis). Blekinge Institute of Technology, Faculty of Computing, Department of Software Engineering. https://www.diva-portal.org/smash/record.jsf?pid=diva2:1334809

Boeije, H. (2002). A Purposeful Approach to the Constant Comparative Method in the Analysis of Qualitative Interviews. *Quality and Quantity*, *36*, 391–409. https://doi.org/10.1023/A:1020909529486

Burke, R. (2011). *Advanced Project Management: Fusion Method Xyz: A Project Methodology Systems Approach for the Project Sponsor to Implement Corporate Strategy*. Burke Publishing.

Camilleri, E. (2016). *Project success : Critical factors and behaviours* (1st). Routledge. https://www.worldcat.org/title/project-success-critical-factors-and-behaviours/oclc/1100062466

Chiyangwa, T. B. & Mnkandla, E. (2017). Modelling the critical success factors of agile software development projects in South Africa. *SA Journal of Information Management*, *19*(1), 1–8. https://doi.org/10.4102/sajim.v19i1.838

Chiyangwa, T. B. & Mnkandla, E. (2018). Agile methodology perceived success and its use: The moderating effect of perceived compatibility. *South African Computer Journal*, *30*(2), 1–16. https://doi.org/10.18489/sacj.v30i2.554

Da Silva, K. M. B. & Dos Santos, S. C. (2017). Critical Factors in Agile Software Projects according to People, Process and Technology Perspective. *Proceedings - 6th Brazilian Workshop on Agile Methods, WBMA 2015*, 48–54. https://doi.org/10.1109/WBMA.2015.19

Dakkak, A., Mattos, D. I. & Bosch, J. (2021). Success Factors when Transitioning to Continuous Deployment in Software-Intensive Embedded Systems. *Proceedings - 2021 47th Eur-*

omicro Conference on Software Engineering and Advanced Applications, SEAA 2021, 129–137. https://doi.org/10.1109/SEAA53835.2021.00025

Davis, B. (2013). *Mastering Software Project Requirements: A Framework for Successful Planning, Development & Alignment*. J. Ross Publishing. https://books.google.co.za/books?hl=en&lr=&id=4NNpAgAAQBAJ

Devlin, A. S. (2006). *Research methods : Planning, conducting and presenting research*. Belmont CA: Thomson/Wadsworth.

Dewantari, D., Raharjo, T., Hardian, B., Wahbi, A. & Alaydrus, F. (2022). Challenges of Agile Adoption in Banking Industry: A Systematic Literature Review, 357–362. https://doi.org/10.1109/ICSEC53205.2021.9684622

Digital.ai. (2019). 15th Annual State Of Agile Report. https://digital.ai/resource-center/analyst-reports/state-of-agile-report

Dikert, K., Paasivaara, M. & Lassenius, C. (2016). Challenges and Success Factors for Large-Scale Agile Transformations: A Systematic Literature Review. *Journal of Systems and Software*, *119*, 87–108. https://doi.org/10.1016/j.jss.2016.06.013

Du Randt, F. J., Van Waveren, C. C. & Chan, K.-Y. (2014). An Empirical Study on the Critical Success Factors of Small To Medium Sized Projects in a South African Mining Company. *The South African Journal of Industrial Engineering*, *25*(2), 13–13. https://doi.org/10.7166/25-2-832

Dubey, S. J. (2011). Key Success Factors in Software Development Projects. *SSRN Electronic Journal*, 1–11. https://doi.org/10.2139/SSRN.1952935

Elmezain, M., Baduruzzaman, W. H. W. & Khoiry, M. A. (2021). The impact of project manager's skills and age on project success. *Brazilian Journal of Operations & Production Management*, *18*(4). https://doi.org/10.14488/BJOPM.2021.017

Fayaz, A., Kamal, Y., Ul Amin, S. & Khan, S. (2017). Critical success factors in information technology projects. *Management Science Letters*, *7*, 73–80. https://doi.org/10.5267/j.msl.2016.11.012

Field, A. (2013). *Discovering statistics using IBM SPSS statistics*. Sage. https://books.google.co.za/books?hl=en&lr=&id=c0Wk9IuBmAoC

Fulgham, C., Johnson, J., Crandall, M., Jackson, L. & Burrows, N. (2011). The FBI gets Agile. *IT Professional*, *13*(5), 57–59. https://doi.org/10.1109/MITP.2011.88

Gandomani, T. J., Ghani, A., Zulzalil, H., Azim, A., Bakar, A., Sultan, M. & Nafchi, M. Z. (2013). Obstacles in moving to agile software development methods: At a glance. *Journal of Computer Science*, *9*(5), 620–625. https://doi.org/10.3844/jcssp.2013.620.625

Goddard, W. & Melville, S. (2001). *Research methodology : An introduction* (2nd). Kenwyn South Africa: Juta. https://www.worldcat.org/title/research-methodology/oclc/953628816

Gordon, L. (2019). *Real research: Research methods sociology students can use* (2nd). https://www.worldcat.org/title/real-research-research-methods-sociology-students-can-use/oclc/1055570026

Gulec, U., Yilmaz, M. & Isler, V. (2017). A Literature Survey: Is it Necessary to Develop a New Software Development Methodology for Virtual Reality Projects? *Article in Journal of Universal Computer Science*, *23*(8), 725–754. https://doi.org/10.3217/jucs-023-08-0725

Guzmán, J. G., Saldaña, J., Ramos, A. A. S. & Esteban, A. S. (2011). Success factors for the management of global virtual teams for software development. *igi-global.com*. https://doi.org/10.4018/jhcitp.2011040105

Hastie, S. & Wojewoda, S. (2015). Standish Group 2015 CHAOS Report - Q&A with Jennifer Lynch. https://www.infoq.com/articles/standish-chaos-2015/

Hedeman, B., Heemst, G. & Fredriks, H. (2005). *Best practice: Project management based on Prince2* (3rd). Van Haren Publishing.

Himmat, M. & Osman, A. S. A. (2020). Agile software development methods and current trends. *Journal of Software Engineering & Intelligent Systems*, *5*(1), 25–30. www.jseis.org

Hobbs, B. & Petit, Y. (2017). Agile Methods on Large Projects in Large Organizations. *Project Management Journal*, *48*(3), 3–19. https://doi.org/10.1177/875697281704800301

Humble, J. & Russell, R. (2009). The agile maturity model applied to building and releasing software. *ThoughtWorks White paper*. https://info.thoughtworks.com/rs/thoughtworks2/images/agile_maturity_model.pdf

Ibrahim, R., Ayazi, E., Nasrmalek, S. & Nakhat, S. (2013). An Investigation of Critical Failure Factors In Information Technology Projects. *10*(3), 87–92. https://doi.org/10.9790/487X-1038792

Johnston, K. & Gill, G. (2017). Standard Bank: The Agile Transformation. *Journal of Information Technology Education: Discussion Cases*, *6*(1), 1–31. https://doi.org/10.28945/3923

Joseph, N. & Marnewick, C. (2014). Structured Equation Modeling for Determining ICT Project Success Factors. *Project Management Institute Research and Education Conference*. https://www.pmi.org/learning/library/structured-equation-modeling-ict-project-success-10616

Joseph, N., Marnewick, C. & Santana, M. J. (2016). Agile software development and IT project performance in South Africa: A positive relationship? *Proceedings of International Association for Management of Technology IAMOT 2016 Conference*, 335–355. https://www.researchgate.net/publication/318206568

Kabir, M. & Rusu, L. (2016). IT project development using capability maturity model. *Information Resources Management Journal*, *29*(4), 35–48. https://doi.org/10.4018/IRMJ.2016100103

Kamath, R. (2019). The Agile methodology makes no sense in banking. *eFinancialCareers*. https://www.efinancialcareers.co.uk/news/2019/08/agile-in-banking

Kateb, G. A., Sweis, R., Obeidat, B. & Maqableh, M. (2015). An investigation on the critical factors of information system implementation in Jordanian information technology companies. *European Journal of Business and Management*, *7*(36), 11–28. https://core.ac.uk/download/pdf/234626994.pdf

Kaur, B. P. & Aggrawal, H. (2013). Critical failure factors in information system: An exploratory review. *Journal of global research in computer science*, *4*(1), 76–82. www.jgrcs.info

Kaur, R. & Sengupta, J. (2013). Software process models and analysis on failure of software development projects. *arxiv.org*, *2*(2), 1–4. https://arxiv.org/abs/1306.1068

Kerr, A. W., Hall, H. K. & Kozub, S. A. (2009). *Doing Statistics with SPSS*. SAGE Publishing. https://uk.sagepub.com/en-gb/eur/doing-statistics-with-spss/book219358

Khmelevsky, Y., Li, X. & Madnick, S. (2017). Software development using Agile and Scrum in distributed teams. *2017 Annual IEEE International Systems Conference (SysCon)*, 1–4. https://doi.org/10.1109/SYSCON.2017.7934766

Khoza, L. & Marnewick, C. (2020). Waterfall and Agile information system project success rates – A South African perspective. *South African Computer Journal*, *32*(1), 43–73. https://doi.org/10.18489/sacj.v32i1.683

Kumar, R. (2011). *Research methodology: A step-by-step guide for beginners.* Sage Publications Limited.

Lalic, D. C., Lalic, B., Deli, M., Gracanin, D. & Stefanovic, D. (2022). How project management approach impact project success? From traditional to agile. *IJMPB*. https://doi.org/10.1108/IJMPB-04-2021-0108

Lehtinen, T., Mäntylä, M., 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. https://doi.org/10.1016/j.infsof.2014.01.015

Levine, D. M. (2010). *Business statistics : A first course (Computer file, 2010) [WorldCat.org]*. Pearson Education India. https://www.worldcat.org/title/business-statistics-a-first-course/oclc/779115893

Lindskog, C. & Netz, J. (2021). Balancing between stability and change in Agile teams. *International Journal of Managing Projects in Business*, *14*(7), 1529–1554. https://doi.org/10.1108/IJMPB-12-2020-0366

Lipschutz, S. & Schiller, J. J. (2012). *Introduction to probability and statistics*. New York: McGraw-Hill. https://www.worldcat.org/title/schaums-outlines-introduction-to-probability-and-statistics/oclc/979659514

Longest, K. C. (2019). *Using Stata for quantitative analysis*. https://books.google.co.za/books?hl=en&lr=&id=ztaIDwAAQBAJ

Marchewka, J. T. (2016). *Information technology project management: Providing measurable organizational value*. John Wiley & Sons. https://books.google.co.za/books?hl=en&lr=&id=rsLlBQAAQBAJ

Marnewick, A. L. & Marnewick, C. (2019). The Ability of Project Managers to Implement Industry 4.0-Related Projects. *IEEE Access*, *8*, 314–324. https://doi.org/10.1109/ACCESS.2019.2961678

Marnewick, C. (2013). To Mature or not to Mature: The Information Systems Conundrum. *South African Computer Journal*, *51*(1), 10–21. https://doi.org/10.18489/sacj.v51i0.186

Marnewick, C., Erasmus, W. & Joseph, N. (2017). *The symbiosis between information system project complexity and information system project success*. AOSIS Publishing. https://doi.org/10.4102/aosis.2017.itpsc45

Marnewick, C., Labuschagne, L., Eloff, M., Steyn, H. & Tobin, P. (2013). *Prosperus Report , The African Edition* (C. Marnewick, Ed.). https://cdn.ymaws.com/www.projectmanagement.org.za/resource/resmgr/Publications/Prosperus_African_Edition.pdf

Marnewick, C. & Marnewick, A. L. (2019). The demands of industry 4.0 on project teams. *IEEE Transactions on Engineering Management*, *67*(3), 941–950. https://doi.org/10.1109/TEM.2019.2899350

Mcleod, L. & Macdonell, S. G. (2011). Factors that Affect Software Systems Development Project Outcomes : A Survey of Research. *ACM Computing Surveys*, *43*(4), 24–56. https://doi.org/10.1145/1978802.1978803

Mkoba, E. & Marnewick, C. (2020). Conceptual Framework for Auditing Agile Projects. *IEEE Access*, *8*, 126460–126476. https://doi.org/10.1109/ACCESS.2020.3007874

Mtsweni, E. S., Horne, T. & Van der Poll, J. A. (2016). Soft Skills for Software Team Members. *International Journal of Computer Theory and Engineering*, *8*(2). https://doi.org/10.7763/IJCTE.2016.V8.1035

Mudarikwa, G. & Grace, T. D. (2018). Agile system development methodologies usage and acceptance in South African banking firms: An exploratory analysis. *Proceedings of the Annual Conference of the South African Institute of Computer Scientists and Information Technologists*, 248–257. https://doi.org/10.1145/3278681.3278711

Nakigudde, S. (2019). *Project Management Models and Software Development Project Success* (Master's thesis). Makerere University. https://doi.org/10.13140/RG.2.2.36203.08482

Naoum, S. G. (2013). *Dissertation research and writing for construction students*. London, New York: Routledge. https://www.worldcat.org/title/dissertation-research-writing-for-construction-students/oclc/823726457

Nasir, M. H. N. & Sahibuddin, S. (2011). Critical success factors for software projects: A comparative study. *researchgate.net*, *6*(10), 2174–2186. https://doi.org/10.5897/SRE10.1171

Naslund, D. & Kale, R. (2020). Is Agile the latest management fad? A review of success factors of agile transformations. *International Journal of Quality and Service Sciences*, *12*(4), 489–504. https://doi.org/10.1108/IJQSS-12-2019-0142

Nguyen, D. S. (2016). Success factors that influence Agile software development project success. *American Scientific Research Journal for Engineering, Technology, and Sciences*, *17*(1), 172–222. http://asrjetsjournal.org/

Nguyen, D. (2016). Success factors for building and managing high performance agile software development teams. *International Journal of Computer*, *20*(1), 51–82. https://ijcjournal.org/index.php/InternationalJournalOfComputer/article/view/537

Oates, B. J. (2012). *Researching information systems and computing* (1st, repr.). Los Angeles: Sage. https://www.worldcat.org/title/researching-information-systems-and-computing/oclc/857891518

Olteanu, C. G. (2018). IT Agile Transformation. *Economy Informatics*, *18*(1), 23–31.

Pallant, J. (2010). *SPSS survival manual : A step by step guide to data analysis using SPSS* (4th edition). Maidenhead: Open University Press/McGraw-Hill. https://www.worldcat.org/title/spss-survival-manual-a-step-by-step-guide-to-data-analysis-using-spss-4th ed/oclc/1031523646

PMI. (2017). *Success Rates Rise 2017 9th Global Project Management Survey* (tech. rep.). https://www.pmi.org/-/media/pmi/documents/public/pdf/learning/thought-leadership/pulse/pulse-of-the-profession-2017.pdf

Poranen, T. (2014). Software Project Management Summaries 2014. https://trepo.tuni.fi/bitstream/handle/10024/96057/software_project_management_2014.pdf

Pozzi, R., Rossi, T. & Secchi, R. (2021). Industry 4.0 technologies: Critical success factors for implementation and improvements in manufacturing companies. *Production Planning & Control*. https://doi.org/10.1080/09537287.2021.1891481

Pretorius, S., Steyn, H. & Jordaan, J. (2012). Project Management Maturity and Project Management Success in the Engineering and Construction Industries in Southern Africa. *The South African Journal of Industrial Engineering*, *23*(3), 1–12. https://doi.org/10.7166/23-3-507

Radujković, M. & Sjekavica, M. (2017). Project Management Success Factors. *Procedia Engineering*, *196*, 607–615. https://doi.org/10.1016/j.proeng.2017.08.048

Raunak, M. & Binkley, D. (2017). Agile and other trends in software engineering. *IEEE 28th Annual Software Technology Conference*, 1–7. https://ieeexplore.ieee.org/abstract/document/8234457/

Remenyi, D. & Bannister, F. (2012). *Writing up your research: The quick guide series*. Academic Conferences; Publishing International Limited.

Ruhe, G. & Wohlin, C. (2016). *Software Project Management in a Changing World*. Heidelberg: Springer. https://link.springer.com/book/10.1007/978-3-642-55035-5

Sandstø, R. & Reme-Ness, C. (2021). Agile Practices and Impacts on Project Success Integration of Project and Production Management. *Journal of Engineering, Project, and Production Management*, *2021*(3), 255–262. https://doi.org/10.2478/jeppm-2021-0024

Saunders, M., Lewis, P. & Thornhill, A. (2012). *Research methods for business students* (6th). Prentice Hall. https://www.worldcat.org/title/research-methods-for-business-students/oclc/669693976

Sekaran, U. & Bougie, R. (2016). *Research methods for business : A skill-building approach*. John Wiley & Sons. https://www.worldcat.org/title/research-methods-for-business-a-skill-building-approach-seventh-edition/oclc/943211712

Shaydulin, R. & Sybrandt, J. (2017). To Agile, or not to Agile: A Comparison of Software Development Methodologies. *arxiv.org*. https://arxiv.org/abs/1704.07469

Sommerville, I. (2016). *Software Engineering* (10th Global Edition). UK: Pearson.

Sonnekus, R. & Labuschagne, L. (2003). *IT project management maturity versus project success in South Africa. The Prosperus Report* (tech. rep.).

Spenser, M. (2010). Understanding the Software Development Process, 1–2.

Standish Group. (2013). *CHAOS MANIFESTO 2013 Think Big, Act Small* (tech. rep.). Boston. https://www.standishgroup.com

Standish Group International. (2012). *CHAOS Manifesto 2012: The Year of the Executive Sponsor* (tech. rep.). Boston. https://www.standishgroup.com/

Sudhakar, G. P. (2016). Understanding the Meaning of "Project Success". *Binus Business Review*, *7*(2), 163–169. https://doi.org/10.21512/BBR.V7I2.1586

Tam, C., Jóia Da Costa Moura, E., Oliveira, T. & Varajão, J. (2020). The factors influencing the success of on-going agile software development projects. *International Journal of Project Management*, *38*, 165–176. https://doi.org/10.1016/j.ijproman.2020.02.001

Theocharis, G., Kuhrmann, M., Münch, J. & Diebold, P. (2015). Is Water-Scrum-Fall Reality? On the Use of Agile and Traditional Development Practices. *Lecture Notes in Computer Science*, *9459*, 149–166. https://doi.org/10.1007/978-3-319-26844-6_11

Tsui, F., Karam, O. & Bernal, B. (2014). *Essentials of Software Engineering* (3rd). USA: Jones and Bartlett Publishers, Inc.

Vallon, R., José, B., Estácio, S., Prikladnicki, R. & Grechenig, T. (2018). Systematic literature review on agile practices in global software development. *Information and Software Technology*, 161–180. https://doi.org/10.1016/j.infsof.2017.12.004

Vavpotic, D., Kalibatiene, D., Vasilecas, O. & Hovelja, T. (2022). Identifying Key Characteristics of Business Rules That Affect Software Project Success. *Applied Sciences*, *12*(2). https://doi.org/10.3390/app12020762

Vijayasarathy, L. R. & Butler, C. W. (2015). Choice of software development methodologies: Do organizational, project, and team characteristics matter? *IEEE Software*, *33*(5), 86–94. https://doi.org/10.1109/MS.2015.26

Walliman, N. (2017). *Research methods: The basics* (2nd). London, New York: Routledge. https://doi.org/10.4324/9781315529011

Yaghoobi, T. (2018). Prioritizing key success factors of software projects using fuzzy AHP. *Journal of Software: Evolution and Process*, *30*(1), 1–11. https://doi.org/10.1002/smr.1891

Zahran, S. (1999). Book Review Software Process Improvement: Practical Guidelines for Business Success. *Maint: Res. Pract*, *11*, 285–291. https://doi.org/10.1002/(SICI)1096-908X(199907/08)11:4