Impact of Project & Team Characteristics Undertaking in Software Development on Small and Medium Organization (SMEs)

Mohammad Sadiq[1], Jibran Rasheed Khan[2] Farhan A. Siddiqui[3], Nadeem Mahmood[4]
[1]O'Reference Ptv. Ltd, Karachi Pakistan.
[2,3,4]Department of Computer Science – UBIT, University of Karachi, Pakistan, University Name
sadiq@oreference.com, jibran_rasheed@hotmail.com, farhan@uok.edu.pk, nmahmood@uok.edu.pk

Abstract: With the Information technology (IT) advancement, software development leveraging in people empowerment. It is the most rapidly growing industry in the IT sector around the globe. The ample demand and need of integrating software to reduce, automate, and accelerate the tasks, turn into increasingly determined in aim and scaling. Beyond the need, every year world is investing billions of dollars in software development but unfortunately, the project failure rate is uncontrollable. To cut the cost of project development and minimizing the failure risk, many organization outsource their project. But statistics showed a large number of projects still failed due to many reasons. This paper performed a state-of-the-art study and examine every aspect of the project and its associated factors including software process methodologies and team impact on project success rate. The study found key team characteristics, project factors, and project management methodologies necessary for the successful accomplishment of undertaken the project in micro to small-medium size organizations (SMEs).

Keywords: Project Management; Team characteristics; Software Process, Software Development; SME; Challenges

I. INTRODUCTION

With the technological advancement in Information Technology (IT) and Information System (IS), the software development profession leveraging in empowering people. During the past two decades, the surprising growth in the software development industry around the globe, not only enabling the possibilities but also ease in management, reducing human efforts, increase work efficiency, controlled over activities, and monitored environment fascinating its ample need and high demand. The software development industry captures a significant ratio in the global economy[1]. This emerged as one of the key economic drivers which is equally beneficial for the national and international impact of their country, especially for underdeveloped countries like Pakistan.

Despite the significant worth of the Software Development industry within a country or in a global society as a key economic driver. The survival key of Software products requires speedy development and marketing to sustain in a highly completive global industry[2]. Besides the global race, unlike traditional projects, there always a failure risk associated with software project development. The discipline was born in 1968 to address issues associated with software development and mitigating them by applying software process improvement (SPI) methodologies[3]. The principal objectives of SPI are to minimizing the project development cost, enhance quality, and reducing the time to market[4]. Unfortunately, now the term "Failure" associates with software as its synonyms[5], due to the high failure rate. In the past 40 years, it is a fact there is not much difference

has been observed in software project failure rate. The project failure rate keep capturing the large ratio, while the largest failure rate observed was 70%[6] and the latest failure rate is measured was 62% that includes projects completely failed and are not done in the scheduled time frame, or not accompanied all the requirements, functionalities and over budget[7]. In another study, it is stated that there were fifty thousand 50,000 undertaken during the year from 1992 to 2004, out of those only approximate 29% of them are termed as successful[8], [9]. The correlating uncertainty of project failure still alarming and troubling the practitioner and confusing in understanding what is negatively impacting project success[6]. The major project failure includes Nike's in 2000 loss in sales about 100$ million-dollar and 20% downfall in stock price, Hewlett-Packard (HP) project in 2004 which cost 160 million dollars, Hershey Foods failure impact in decrease stock by 8 %[10].

Among the various factors that influence the ultimate results of project development, company and team are one of the major and core factors. As the paper focuses on the small companies or organization's role to deliver and undertake projects and what are the factors that undesirably impacting of project success. Before drill down, to other factors, it is important to understand the different types of software organizations exist in our global society. The enterprise or organization is categorized by several indicators such as capital, balance, investment, employees, and employed. In general, when we say company or organization people idealized something having large infrastructure, lots of people power, heavy machinery. But, in IT and specifically the software development industry, only the size that is based

on the number of employees of the aforementioned indicators is a measure to categorize company type. Very Small Enterprise (VSE) or Micro Enterprise (ME) has a minimum of 1 or 2 people to 10 people in a group to carry out the project. Small Enterprise (SE) has headcount starting from 11 people group to 49 people team. *Medium Size Enterprise (MSE)* consists of more than 50 people to 249 people with several departments and sub-teams. Collectively, the VSE or ME or Startups, SE, and MSE are called as Small and Medium Size Enterprise (SME). Large Enterprise *(LE)* has a minimum of 250 or more employees with several departments, teams, distributed over the same or different geographical region. This type of organization having several projects under dedicated teams and some shared team (Online: EuroStat).

Hereafter, in the rest of the paper, we refer SME as our focus entities in the software industry and evaluates the project and team characteristics in such an organization. Whereas the large enterprises dealing with high-end projects in terms of duration, budget, and complexity capturing the cooperate sector of the industry. The SME plays an important and significant role where LE neglected or rejected the medium or lower end clients and their needs[11]. SMEs take advantage of producing products and services to a market of existing products, maintenance, and extension services[12]. Sometimes LEs also outsource projects of new product extension, add-ons, and other services to reduce the production cost and save the in house human resources that could be adjusted to other projects of the company. Globally, SME is the most predominant form of enterprise that capture about 92.2% of European enterprises, 98% of enterprise in the USA, and more specifically VSE covered about 70% to 90% in Organization for Economic Co-operation and Development (OECD) countries and about 57 % in the USA where employees are less than 10[13]. SMEs are usually the sub-contractor that comply with standards and are the main driver of innovative ideas and applications.

The rest of the paper is design as in the next section we will look at the research methodologies adopted for identifying the facts also few case studies results considered in the study. Then, it will strictly focus on Factors influencing the project for SME, under this head paper will discuss core influencing factors that have directly or indirectly impacted the project. It will discuss the prominent SDLC methodologies used in almost every SME organization, compare their success and failure rate and methodologies adoptability ratio. This will be followed by the team characteristics, which will focus on the type of team, its features, and need to have a high impact on project success rate. After studying team types, will look at the factors that causing the project failure, including soft skills and technical skills, and will discuss the most important team factors and skills necessary to deliver the successful project in the discussion and recommendation section. Finally, the paper ends with the conclusion section.

II. METHODOLOGY

The research is based on an exploratory study technique, in which it used a constructive and inductive approach as an epistemological viewpoint. The work is primarily taken from the numerous recent paper from reputable sources like Elsevier, ScienceDirect, IEEE, Springer, and ResearchGate to gain knowledge and subject insight. The study found 110 papers using Google Scholar and Elsevier platform for article searching using the keywords; Factor influencing project, Software project improvement, why a project fails, project methodologies in VSE and SME, project team characteristics, people factor, team factor, factors influencing projects, software methodologies, most adopted methodologies, suitable methodologies for SME, VSE and SME role, including some renowned organization research report for software success/failure rate and their findings such as Project management Survey Report 2017 by PMI, and Standish Group 2015 report. The work also includes three books; Software Engineering by Roger Pressman, Software Engineering by Ian Sommerville, and PMBOK by PMI. After filtering the content, among 110 papers only 57 are taken. After removing duplications only 37 out 57 are taken as part of this work based on objectives and subject relevancy.

The objective of the study is to identify the issues and factors that influencing directly or indirectly on a project, its success ratio, their importance, identify failure causes.

O1: To identify factors influencing the project and causes failure.

O2: To determine which SDLC methodology should SMEs adopt to ensure project success.

O4: To identify the factors that should be considered in building the project team.

To achieve the study objectives, the following 5 research review questions are built to validate and recognize root causes of project failure in organizations especially in SMEs.

RQ1: Why team and selection has impact on project success ratio? What should be considering in team selection and what really matters?

RQ2: Why should adopt SDLC methodology and which methodology should adopt to run project? Does SDLC plays role in project success?

RQ3: Why project Team and SDLC selection determine the pre-success ratio of project? How Team and SDLC selection ensure project success?

RQ4: Does software fails only due to the small team size or in small size organization? Is this really concern to care of while outsourcing project to SMEs?

RQ5: Why project fails? What are the factors and mistakes that organization made?

III. FACTORS INFLUENCING PROJECT

A. *Project Characteristics*

Many factors influence project success and completeness which includes few core aforementioned factors; adopting software process methodology, selecting

the right team for the project, having a good leader, and team equips with the right technical and soft skills. Besides these factors, other core factors that apply to any size of SME are project complexity, scope, and requirement identification, financial budget, planning, and execution, controlling, and management of projects.

- *Project scope* helps in identifying the project complexity, boundaries, features, functionalities, and goals which also involving in project planning and documenting activates.
- *Project complexity* identification assists in determining the project's critical and complex task, logical and technical complexness, the necessity to pick new technology, analyzing risk[14], [15].
- *Financial and Budget*, after getting a complete understanding of the project it is the main task to estimate the project hidden cost and direct cost, the financial risk of currency conversion possess (in case of dealing with other currency) which directly associating with the project life and completeness[16].
- *Planning*, no doubt core activity, without proceeding with this no one identifies how much time, cost, resource, and the team should be required to deliver the product.
- *Controlling and Management* allows us to keep track of each project activity, process, task, and code controlling using version controlling tools, application, and some mechanism to monitor individual contribution and thus measure overall team performance and project health[17].

B. *SDLC Methodologies*

Software Development Life Cycle termed as SDCL, which states the cyclic processes that occur throughout the software life from project initialization to deployment and maintenance of software. SDLC is a descriptive and detailed plan which describe project need, scope, feasibility, requirements, project nature, and type, how to develop, maintain, alter or enhance the software product while keeping the quality and completeness[18], [19]. Table 1 exhibit the various software process methodologies that one can follow to proceed software project. Whereas, work will only discuss the most adopted methodologies (Agile and Waterfall model) by the practitioner and target entities i.e. SME (includes VSE/ME, Startup, SE, and ME in this context).

TABLE I   TOP 12 SOFTWARE METHODOLOGIES

Waterfall Model	Feature Driven Development
Rapid Application Development	Agile software development
Extreme Programming	Rational Unified Process
Lean Development	Spiral Model
Prototype Model	Scrum Development
Dynamic Systems Development Model	Joint Application Development

The waterfall process model is also known as the classic life cycle model, the oldest paradigm in software deployment. The process itself suggests a systemic approach which tells that the activities of each stage or the prior stage is well defined and clear [13]. The processes in this model begin with the client specification of project, requirement gathering, planning, designing, modeling, and finally deployment. The linear flow of Waterfall sometimes restrict and bound other team members and participant to wait until the prior work complete which called as Block State. This might result in surpass time added to the following stages and effects on the project timeline and budget[19]. This model is most suitable for the project where the requirement is crystal clear and definite, the task is well defined and a shared team can be integrated to save time and minimize cost factor.

Agile, unlike the waterfall model, more focusing on development rather design and documentation. It allows the team to quickly respond to the uncertainties and changes. It is also adaptable and good to use where project requirements keep changes by time and more concern with functionalities[18], [19]. Agile uses the incremental approach to specification, development, and delivery intended to deliver the project quickly and promptly by controlling and reflecting all the changes to the project. Agile methodology phenomena are mainly based on two aspects, philosophy and development guidelines[19]. The philosophy keeps customer concern and satisfaction by accommodating incremental delivery in the early stage that keep interacting customer and drive changes more appropriately. On the other hand, development guidelines allow design and activity stress release by active communication between the development team and the client. This approach highly suitable and adaptable for all size of time that allows coherent bond between intended parties. Table 2 listed all the agile methodologies [7], [10], [16]–[24] while Table 3 illustrates the most adaptable agile flavor among all the methodologies their ratios under the agile umbrella [24][29].

TABLE II.   METHODOLOGIES UNDER AGILE UMBRELLA

Lightweight Approaches	Full Approach
Scrum	Scrum Of Scrum
Continues development	Scrum at scale
Kanban	Large Scale Scrum
Extreme programming	Scaled agile framework
Continues integration	Disciplined agile delivery
Lean software development	Dynamic system development methods
Feature-driven development	Agile project management
Test-driven development	Agile unified process
Crystal clear	Open unified process

TABLE III MOST ADAPTABLE AGILE METHODOLOGIES & USABILITY RATIO

Agile Methodology	Usability Ratio
Scrum	56%
Hybrid	14%
ScrumBan	8%
Scrum/XP hybrid	6%
Others (Multiple)	16%

The hybrid model, by definition hybrid model, could be the combination of any model that collectively carry out in executing software project development, refine the software process, and reflecting the uncertainty. In this context, the hybrid model refers to the Waterfall and Agile.

IV. SDLC Methodology Impact on SMEs

This section focus on the most frequently reported factors where SME usually bear the loss or cause project failure. The study found that SME mostly face challenges in identifying the project scope, selecting or lacking the right number of people and skills set, wrongly adopt methodologies or not incorporate them in project appropriately, not analyze project complexity and need, management which resulted in the overestimated budget, planning delays, elevating failure probability and produce low-quality products. Performed the analysis based on the study, case studies results and findings performed the analysis and found the reason that could mitigate or reduce the project failure ratio and enhance the probability of successful delivery.

It observed that most of the time adopted SDLC methodology not properly accommodating with its needs. Usually, the methodology adopted in the beginning but when there is any delay or any mishandling occurs, use the non-standard approach to power the rapid result. Other issues observed is a change in scope or project scope not identified. Referring to the most useable methodologies discussed before, it found that the waterfall technique is not much flexible and does not allow any sort of the change in project scope and its requirement during the project life cycle. Although, it allows flexible time and resource management. Whereas, the Agile technique very much flexible and suitable to most of the techies' nature of updating and incorporating the changes to project time to time but it will cost in resource engagement which inflexible to accommodate easily and has a direct impact on the project life and delivery. This is how the statistics reflecting the fact during the past 10 years[9], [28], [29] shown in Tables 4 and 5.

TABLE IV AGILE VS WATERFALL PROJECT SUCCESS RATE

Method	Agile		Waterfall	
Year	2009-2012	2013-2017	2009-2012	2013-2017
Successful	39	42	11	26
Challenged	52	50	60	53
Failed	9	8	29	21

TABLE V AGILE VS WATERFALL FAILURE RATIO BY PROJECT SIZE

Project Size	Small	Medium	Large
Agile	4%	10%	16%
Waterfall	10%	20%	35%

From the above tables, there were about 39% of success ratio recorded during the term 2009-2012 that will slightly increase in the next five year to 42% of Agile users during 2013-2017 span[]. Whereas, SMEs are failed to successfully deliver the 16% large projects which might be the cause in the small and unsuitable team of Agile users. On the contrary, Waterfall seems much straightforward to adopt and easy but many are failed to complete the projects. The figures reflecting that only 11% was the success ratio during 2009-2012, which significantly improved in the next term 2013-2017 to 26% [Standish Chose Report]. But, still losing or not completing the project within the specified project timeframe.

V. Project Team Characteristics

The Team is the core element of any kind of project that idealizes and predict how project ultimate result in terms of success and failure regardless of project type and nature. There are many perspective and definition of what is team refer to? The simplest definition of a team is the group of people responsible to follow the practices to fulfill project requirements[30]. Another version is, Team is a small group of people committed to a common goal with relative skills and interdependent roles[31]. Another version stated is team refers to a small group with suitable corresponding skills set, committed to achieving a common goal set and approach for which they are equally accountable[32].

Software project development teams are very different than any other traditional project teams, they are intangible, innovative, and yet most complex than other kinds[5]. As the field evolving rapidly, the ideal team requirements also vary by time. There is no such phenomenon or standard to fix the team selection criteria. Thus, the software project team can be characterized by only people's personality, interpersonal relationship, leadership, people soft skills, and technical skills that best suites to specific project requirements [33]. Although the study founds more than 50 team characteristics, this work focus and covers only core important factors that every software project team must have.

- Leadership is one of the key team characteristics that could lead the project successfully and raise the possibility of achieving goals [20], [29], [30]. A leader helps in resolving the inter-team conflict, social or personal heat among the team members [23]. Also, engaging members actively and encouraging team morale and keep motivated even hard deadline to achieve the goals[2], [35].

- Team Member *and Size*, it is also an important factor to consider that directly impact on several project factors such as cost and management. Team size is very critical and can only be determined by the understanding of project nature, type, and need. If one hired and below the number required team members, it will decrease the leadership performance because a leader cannot get the advantage of what member does not know or not skilled. This might reduce the project cost but highly resulted in delays. Likewise, if one hired more people to achieve the project

goals, it will only be beneficial to add extra aid to team skills and experience, but outreach the project budget[36].

• Technical Skills, another top concerning capability in software team selection. It the most commonly reported complex problem what are the skills people should have? What technical skills should determine? How many skills should one member have? These questions are confusing and challenging in picking team members. This also distributed under numerous subfactor but will cover the only basic need that one should have. It is very critical to know what you are hiring and whom you are hiring. If the member you are choosing for the project has required experience and knowledge i.e. specific programming language, application, and framework familiarity is relevant to project and development requirement[30][37]. Here, technical skills not only concern with core programming language skills but in general it refers to a technical person either front-end developer, Back-end, Architecture, design, or modeler. How many such high skills programmer project needs. Secondly, with the experience of people how many people comparatively less experience but relevant to the project should assist the team collectively in producing a quality product within the projected time. This is the key stage where one should cut off the project cost with the right mixture of high and average skills members [38].

• Softs Skills, also known as non-technical skills are the core element that reflects people nature and personal capability to deal with uncertainty in professional and social life, inter-personal relationship, behavior, communication and managing people are few examples of soft skills[39]. Soft skills are something that not only critical for a team but equally vital for the project stakeholders, management, leaders irrespective of high skills, or experience to inexperience persons [3], [30]. These allow people to communicate and collaborate among the team members and within the organization, to understand and effectively deliver concerns to leaders and managers. Another important soft skills which are on the top necessity specifically and especially for any software engineer is learning skills[33]. Learning is a top-notch requirement in a software engineering discipline and for a software engineer to have learning capability either the person is highly skilled or fresher. If the software engineer does possess this soft skill s/he might be out phase from the market, devalues its worth, such person experience has no meaning to organization or project and it became the threat for project success and right choice for a team[8], [37].

C. Impact of Team on Project

The 50% of project success can be estimated with the right choice of people and the right team size[34], [35]. A good team or featured team should have relevant competitive

skills in their domain. With an experience skilled domain, it is good that team members have knowledge and understanding of co-related areas [30]. The features team also defined as the pentagon team. The word pentagon is referred to as core 5 areas of the software development team that are 1-Frontend, 2-backend and database, 3-Architecture, 4-UI/UX, and 5-Software Testing[33] because a team performance never is evaluated by an individual member efforts[30]. Nor this only based on only subjective or objective nature, but it measured on both parameters collectively[30], [33], [37]. It is also important to realize that not every person belongs to an organization is part of a team[32].

VI. RESULT AND DISCUSSION

The study found more than 50 factors reported by various researchers in different scenarios and project types which are either directly indirectly associated or dependent factors. These factors are grouped into three categories; project factors, software process methodologies, project team factors, and only selected core factors among all reported are considered in this work as mentioned in the above section.

From the project factor, [O1] the most common and identified factor that directly impact on project success is an ambiguous scope which troubles in defining project boundaries and directly impact on project success. The scope is the first element in a project life which declares what will be end product or service will deliver to end-user. Furthermore, it also helps in identifying project complexity, budgeting, and feasibility. The second most identified issue is lack of proper planning directly impact on project success. Besides this, it also responsible for the smoothness process, budgeting, and resources management and time allocation that influencing on overall project success rate. This is mostly found in SMEs where the group of inexperienced techies carries out the project or where less experience SMEs lacking planning activity or concern resources.

From the second category, software process methodology, the most highlighted factor is the selection of right and appropriate methodology which aid in handling software process and some earlier stage neglected or unidentified issues such as ambiguous project scope and process management. Among the numerous approaches, not every methodology capable of dealing with any prior or occurring issues during the software development process. Each process approach should be chosen according to the project nature and its complexity or ambiguity. To achieve the [O2] of the study, [RQ2] founds that among all the process methodologies Agile turns to be the most flexible and successful approach which is much capable of handling changes and ambiguity in project statements, meeting milestones and ensure smooth development process in SDLC. Adoption of suitable process methodology defines working criteria, a procedure to allow the inclusion of updating and changing without effecting other process components [RQ3].

From the third category, the project team, selection of project team members directly influence project success rate and indirectly impact many other factors as well. It is found that about 50% of success is determining the right project team selection [RQ3]. There are many factors to choose the right assets for the project, among all the most reported factors are the right mix of technical skills which soft skills and team size [O3]. Technical skills are hard to underline as these vary project to project and need of required skills level should be matched. Whereas, soft skills most frequently reported are management, communication, interpersonal relationship, and learning skills. These help in building and maintaining a healthy environment, capable of learning new things in a challenging period and handling tight timeline. Besides this, the most considerable part of this category is size. Team size directly impacts on project success rate, indirectly and associating with other factors such as project budget and management. Like technical skills, project size cannot be hardline to large, minimum, or small as this varies by project and its requirements.

On the contrary, the secondary objectives of the study are to identify that software only fails in SMEs or not [RQ4]. Software not only fails due that are undertaking in SMEs, but renowned large enterprises failed in past decades as Nike's project fails in 2000 bear cost of 100 million dollars, Hewlett-Packard (HP) project fails in 2004 cost 160 million dollars and Hershey Foods failure cause 8% decrease in its stock[10]. Whereas, the core reason behind software failures either in SME or large enterprises are the same as mentioned above. In the past two decades, the largest failure observed was 70% and the latest failure rate was 62%[6]. The study found interesting facts that collectively (direct and outsourced projects) SMEs covers about 92.2% of the European market and 98% of the US market[13]. About 75% of US companies outsource the project [30]. Whereas, from Table 2 and 3 SMEs more handily capable of delivering small and medium-size projects where the failure rate it comparatively minimal i.e. 4% and 10% respectively as compared to a large project. Thus, this can infer that with inappropriate require team size impact project success ratio and increases failure probability[RQ4].

VII. CONCLUSION

This paper performed the exploratory review to find the team characteristics and project factors that influencing and negatively impacting on the project delivery, success, and product quality. The study highlighted the most important core factors which lead the project failure or resultant dissolving SME based on the analysis of the reports and findings. Paper founds mostly SME fails in incorporating SDLC mythologies, identifying project complexity, building the right team with effective leadership and management. Also, many SMEs are not giving much attention to the soft skills set. Majority SMEs are the combination of low or inexperienced college and university techies. Also, large companies outsource projects to control or reduce the budget but not ensure whom to award. These lacking could be overcome by adopting soft skills such that learning, communicating, and leadership skills. Whereas, the other factors that need more attention are proper understanding of project scope, planning, management, and expectation controlling.

REFERENCES

[1] P. M. Clarke, R. V. O. Connor, T. Rout, and A. Dorling, *Software Process Improvement and Capability Determination*. 2016.

[2] S. Sarin and C. McDermott, "The Effect of Team Leader Characteristics on Learning, Knowledge Application, and Performance of Cross-Functional New Product Development Teams," *Decis. Sci.*, vol. 34, no. 4, p. 108, 2003.

[3] A. Medina and A. J. Francis, "What Are the Characteristics that Software Development Project Team Members Associate with a Good Project Manager?," *Proj. Manag. J.*, vol. 46, no. 5, pp. 81–93, Oct. 2015.

[4] T. O. A. Lehtinen, M. V. Mäntylä, J. Vanhanen, J. Itkonen, and C. Lassenius, "Perceived causes of software project failures - An analysis of their relationships," *Inf. Softw. Technol.*, vol. 56, no. 6, pp. 623–643, 2014.

[5] W. Al-Ahmad, K. Al-Fagih, K. Khanfar, K. Alsamara, S. Abuleil, and H. Abu-Salem, "A Taxonomy of an IT Project Failure: Root Causes," *Int. Manag. Rev.*, vol. 5, no. 1, 2009.

[6] D. Cecez-Kecmanovic, K. Kautz, and R. Abrahall, "Reframing Success and Failure of Information Systems: A Performative Perspective," *Misq*, vol. 38, no. 2, pp. 561–588, 2014.

[7] C. Geambaşu, I. Jianu, I. Jianu, and A. Gavrila, "INFLUENCE FACTORS FOR THE CHOICE OF A SOFTWARE DEVELOPMENT METHODOLOGY.," *Account. Manag. Inf. Syst.*, vol. 10, no. 4, pp. 479–494, 2011.

[8] K. M. Whitney and C. B. Daniels, "The root cause of failure in complex IT projects: Complexity itself," *Procedia Comput. Sci.*, vol. 20, pp. 325–330, 2013.

[9] T. Resolution and F. O. R. All, "CHAOS REPORT 2015," 2015.

[10] Y. K. Dwivedi *et al.*, "Research on information systems failures and successes: Status update and future directions," *Inf. Syst. Front.*, vol. 17, no. 1, pp. 143–157, 2014.

[11] N. Tripathi, E. Annanperä, M. Oivo, and K. Liukkunen, "Exploring Processes in Small Software Companies: A Systematic Review," 2016, pp. 150–165.

[12] M. Muñoz, J. Mejia, B. Corona, J. A. Calvo-Manzano, T. San Feliu, and J. Miramontes, "Analysis of Tools for Assessing the Implementation and Use of Agile Methodologies in SMEs," 2016, pp. 123–134.

[13] L. Ibrahim, E. Wallmüller, and W. Daschner, "Using Enterprise SPICE in Very Small Entities," 2016, pp. 423–430.

[14] P. Fitsilis, "Measuring the Complexity of Software Projects," in *2009 WRI World Congress on Computer Science and*

Information Engineering, 2009, vol. 7, no. May 2009, pp. 644–648.

[15] P. Fitsilis, A. Kameas, and L. Anthopoulos, "Classification of Software Projects' Complexity," in *Information Systems Development*, no. April 2014, New York, NY: Springer New York, 2011, pp. 149–159.

[16] J. M. Verner and L. M. Abdullah, "Exploratory case study research: Outsourced project failure," *Inf. Softw. Technol.*, vol. 54, no. 8, pp. 866–886, 2012.

[17] X. Larrucea, R. V. O. Connor, R. Colomo-palacios, and C. Y. Laporte, "Software Process Improvement in Very Small Organizations," *IEEE Softw.*, vol. 33, no. April, pp. 85–89, 2016.

[18] I. Sommerville, *Software Engineering*, 9th ed. Addison-Wesley, 2011.

[19] R. Pressman, *software engineering : A practicioner's Approach*, 7th ed. McGraw Hill, 2010.

[20] L. R. Vijayasarathy and C. W. Butler, "Choice of Software Development Methodologies: Do Organizational, Project, and Team Characteristics Matter?," *IEEE Softw.*, vol. 33, no. 5, pp. 86–94, 2016.

[21] A. Martin, C. Anslow, and D. Johnson, "Teaching Agile Methods to Software Engineering Professionals: 10 Years, 1000 Release Plans," 2017, pp. 151–166.

[22] J. A. Livermore, "Factors that significantly impact the implementation of an agile software development methodology," *J. Softw.*, vol. 3, no. 4, pp. 31–36, 2008.

[23] E.-M. Schön, D. Winter, M. J. Escalona, and J. Thomaschewski, "Key Challenges in Agile Requirements Engineering," 2017, pp. 37–51.

[24] S. A. Hossain, "Comparative Study on Agile software development methodologies," *Glob. J. Comput. Sci. Technol.*, vol. 13, no. 7, p. 25, 2013.

[25] A. Alami, "Why Do Information Technology Projects Fail?," *Procedia Comput. Sci.*, vol. 100, pp. 62–71, 2016.

[26] K. Siau, X. Tan, and H. Sheng, "Important characteristics of software development team members: an empirical investigation using Repertory Grid," *Inf. Syst. J.*, vol. 20, no. 6, pp. 563–580, 2010.

[27] R. Joslin and R. Müller, "The impact of project methodologies on project success in different project environments," *Int. J. Manag. Proj. Bus.*, vol. 9, no. 2, pp. 364–388, 2016.

[28] KPMG, "Proect management Survey Report 2017," 2017.

[29] VersionOne Inc., "COLLAB.NET VERSIONONE," 2017.

[30] G. Purna Sudhakar, A. Farooq, and S. Patnaik, "Soft factors affecting the performance of software development teams," *Team Perform. Manag. An Int. J.*, vol. 17, no. 3/4, pp. 187–205, Jun. 2011.

[31] A. M. Gondal and A. Khan, "Impact of team empowerment on team performance: case of the telecommunications industry in Islamabad," *Int. Rev. Ofbus. Res. Pap.*, vol. 4, no. 5, pp. 138–46, 2008.

[32] D. K. Smith and J. R. Katzenbach, "The discipline of teams," *Harv. Bus. Rev.*, vol. 83, no. 7, p. 162, 2005.

[33] M. Gidlund, "Measuring feature team characteristics of software development teams," 2016.

[34] M. Pyhäjärvi, "SPICE – International Standard for Software Process Assessment," 2004.

[35] PMI, "Success Rates Rise 2017 9th Global Project Management Survey," 2017.

[36] R. S. Huckman, B. R. Staats, and D. M. Upton, "Team Familiarity, Role Experience, and Performance: Evidence from Indian Software Services," *Manage. Sci.*, vol. 55, no. 1, pp. 85–100, 2009.

[37] V. Lalsing, S. Kishnah, and S. Pudaruth, "PEOPLE FACTORS IN AGILE SOFTWARE DEVELOPMENT AND PROJECT MANAGEMENT," *Int. J. Softw. Eng. Appl.*, vol. 3, no. 1, pp. 117–137, 2012.

[38] A. Trendowicz and J. Münch, "Factors Influencing Software Development Productivity-State-of-the-Art and Industrial Experiences," *Adv. Comput.*, vol. 77, no. 09, pp. 185–241, 2009.

[39] M. Niazi *et al.*, "Challenges of project management in global software development: A client-vendor analysis," *Inf. Softw. Technol.*, vol. 80, pp. 1–19, 2016.