POLITECNICO DI TORINO

Corso di Laurea Magistrale in:

INGEGNERIA GESTIONALE (ENGINEERING AND MANAGEMENT)

Tesi di Laurea Magistrale

EVALUATION OF AGILE PROJECT MANAGEMENT SUCCESS FACTORS IN LARGE SCALE INFORMATION TECHNOLOGY DEVELOPMENT PROCESSES

Supervisor	Candidate
Alberto de Marco	Carlos Eduardo Ramírez Negrete

Department of Management and Production Engineering

Collegio di Ingegneria Gestionale

Academic year 2020/2021

Evaluation of agile project management success factors in large scale information technology development processes

ABSTRACT

Successful agile implementations for IT development processes have revolutionized the way in which software is being delivered. During the last decade success factors of agile development research developed rapidly, and an increased attractiveness of the agile methodology for large scale set-ups has surged. This dissertation proposes a model consisting of six factors that that influence the success of large scale agile IT development project, qualifying success in terms of cost, time, scope and stakeholder contentment. A 5 point Likert-scale type survey was sent out to 85 agile practitioners. Statistical analysis carried out via SPSS v. 21.0. Suggest that personal characteristics, technical capabilities as well as a company´s management commitment play a crucial role in the success of large scale agile development processes, while the company's organizational culture seems to be unconnected. Identifying the importance of which factors bring the highest success helps setting out scalability mechanisms that correctly prioritize them, thus improving project outcomes. Relevant considerations such as exogenous variables that might affect survey respondents as well as research limitations and future work are addressed along the previous matter.

Keywords: Agile, Agile development, IT development, Large scale, Success factors, Project Success, Scalability.

ACKNOWLEDGEMENTS

I would like to thank the following people, without whom I would not have been able to complete this research, and without whom I would not have made it through my master's degree.

My supervisor, Professor Alberto de Marco, whose expertise was key in formulating the research survey items. Your insightful feedback pushed me to sharpen my thinking and brought my work to a higher level.

I am grateful to all of those with whom I have had the pleasure to work during this and other projects along during my Bachelor and Master studies. Especially, to the professors in the Engineering and Management department at *Politecnico di Torino* and the Bioengineering Department at Universidad de Antioquia, to whom I will always be deeply thankful for having shared their expertise in their courses.

I would like to acknowledge my colleagues at *Politecnico di Torino* and *Universidad de Antioquia* for their wonderful collaboration and company in and outside the classroom. Thanks for your stimulating discussions as well as happy distractions to rest my mind outside of my research.

And my biggest thanks go to my family for all the support you have shown me through this research, the culmination of seven years of learning. It was because of you that I took on this double degree opportunity and had the chance to write this thesis. To my father thanks for your advice and availability for any challenge that arose. To my mother, thanks for being the ear I could always talk to and without which I would have stopped these studies a long time ago, I promise we will see each other soon.

Table of contents

Introduction

The agile methodology has revolutionized information technology acting as a response to the ever-changing trends, the aim for operational excellence as a de-facto requirement and the need for a system that prioritizes adaptability and the optimization of efficiency as their core values. The introduction of agile has allowed companies to respond more quickly to change, increase R&D productivity and significantly impact their business growth; and even though the agile framework has been initially created for the software development industry its main principles have evolved beyond their origins to dive deeper into business workflow management, productivity, and customer responsiveness.

Agile software development methods were initially designed and work well for small, single, co-located development team settings employing iterative development cycles scattered by user feedback [1], a situation that favours flexibility and adaptability, which is, in fact, the main creator of value and the core of the attractiveness for the methodology. Agile can adapt to almost any conditions and processes of the organization. The charm of being agile has been increasingly found interesting across large and globally distributed organizations resulting in a considerate amount of effort being put into scaling the agile methodology across enterprises mainly because it has been proven to bring huge benefits to the small scale [2]. Numerous scaling frameworks and scaling practices have been proposed by consultants and practitioners to support globally distributed large scale agile transformations including Scrum-of-Scrums, Large Scale Scrum (LeSS), Scaled Agile Framework (SAFe), Disciplined Agile Delivery (DAD), Spotify Model, Nexus and Scrum at Scale [3] However, this trend has certainly not been challenge free and a considerate number of issues have become apparent, mainly: adaptability of the inter-team coordination routines resulting from the need to reach agreement on many decisions with experts, managers, stakeholders and other teams [4], and quality concerns with the need for frequent releases, which forces companies to govern, control and standardize multi-team development projects. Further, de-motivators such as traditional organizational culture, lack of agile experts, reluctance to adopt and lack of management and commitment support all play an important part as they pose as threats that could potentially damage project execution [5].

Still, agile transformation has been increasingly gaining popularity among experts, and each year new research themes concerning the large-scale start surging: -How do coordination mechanism change over time in large-scale agile? -What information systems infrastructure is needed in large

scale agile projects? - How to transform from control through the hierarchy, to control through autonomous teams and communities? [6] However, as mentioned before, available research on agile methods has long focused on single teams, and, in order to address the viability and effectiveness of the already identified success factors on the large scale further diving should be done, furthermore, there is a need to understand how the different and specific challenges of project portfolio management affect the agile transformation, how different scaling frameworks allow for a different focus, taking into account variabilities such as context-independent evaluation and how the product architecture is the main driver for how to drive a successful agile adaptation [7].

Background

The Agile methodology

The agile methodology was essentially born out of the necessity to satisfy the customer through early and continuous delivery of valuable results in the form of software releases [8]. It is a process in which the segmentation of a general problem allows for constant collaboration with stakeholders and add continuous improvement and iteration at every segment. After the initial client specifications (purpose and scope of the software) are received and clarified to the team the latter one cycles through a process that includes planning, executing and evaluating resulting even in a modification of the product to fit the customer needs better [9].

The agile principles were firstly discussed in the 1970s by William Royce who published a paper on the development of large software systems, but it wasn´t until later in 2001 when *the Agile Manifesto*, a "formal proclamation of four key values and 12 principles to guide an iterative and people-centric approach to software development" was published by 17 software developers. Agile revolves around four values that can be found on the Alliance´s manifesto [10]:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

Essentially, the agile methodology refers to these four principles as well as all the frameworks that implement them, the most popular ones including Scrum, Kanban, Extreme Programming (XP), Adaptive Project Framework (APF) Crystal, Dynamic Systems Development Method, and

Feature-Driven Development. In real-life practice, most software developers choose to take parts of the frameworks as they see fit and as they iterate on their own agile processes. Among all of the *agile* variations, Scrum is by far the most widely adopted [11].

The Scrum framework

Scrum itself is an agile framework that provides effective team collaboration on complex products. Created in 2010 by Ken Schwaber and Jeff Sutherland, all of Scrum's roles, events, artefacts, and the rules that bind them together were gathered in what is now called: *The Scrum Guide*, to explain Scrum clearly and succinctly [12].

According to scrum.org, scrum´s birthplace, "Scrum is not a methodology". It is based on the implementation of the scientific method of empiricism. Scrum replaces a programmed algorithmic approach with the possibility to enable a person to discover or learn something for themselves, with respect for people and self-organization to deal with unpredictability and solving complex problems [12]. Inside this framework, there are two main roles to fit: Scrum Master and Product Owner. The Scrum Master is essentially a team leader, they have the role of establishing responsibility for following the *agile* framework, providing guidance and education to the Scrum Team and removing unnecessary elements that keep the team from doing work. On the other hand, the product owner must be up to date with the projects' stakeholders' expectations and defines and gathers all possible requirements and resources that the Scrum Team needs. In addition, they also help set priorities straight by communicating their vision [13].

To maintain team meetings to the minimum and render work more effective Scrum includes some prescribed events, all being time-boxed. The Scrum Events are [12]:

- Sprint: A time-box of one month or less where a product increment (iteration goal) is created.
- Sprint planning: Setting the sprint goal.
- Daily Scrum: 15-minute time-boxed event for the Development Team to synchronize activities and create a plan for the next 24 hours.
- Sprint review: Inspection of the Increment and adaptation of the Product Backlog if needed.
- Sprint retrospective: Opportunity for the Scrum Team to inspect itself and create a plan for improvements.

Scrum also includes artefacts, which are designed to maximize the sharing of key information so that everybody has the same understanding of the processes being done. The Scrum Artifacts are [12]:

- Product Backlog: List of requirements in the product.
- Sprint Backlog: Set of Product Backlog items selected for the actual Sprint, plus a plan for delivering the product Increment and realizing the Sprint Goal.
- Increment: Sum of all the Product Backlog items completed during a Sprint. Increment values are accumulated throughout the whole project.

Figure 1 represents how a single Scrum team works in order to deliver a successful final product, showcasing the different events and artefacts that take place.

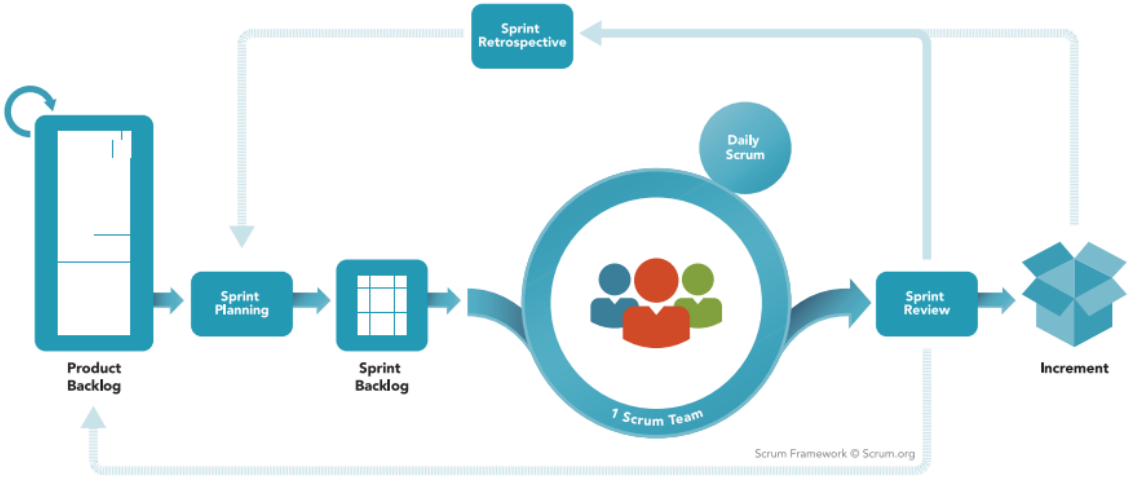


Figure 1: The Scrum framework.

Available online at Scrum.org

The Kanban framework

Kanban is a highly visual workflow management method that is popular among Lean teams. Like Scrum, Kanban is a process designed to help teams work together more effectively [14].

Kanban is based on 3 basic principles [15]:

- Visualize what you'll do today (workflow automation): Seeing all the items within the context of each other can be very informative.

- Limit the amount of work in progress (WIP): This helps balance the flow-based approach so teams don‘t start and commit to too much work at once.
- Enhance flow: When something is finished, the next highest priority item from the backlog is pulled into play.

Kanban promotes continuous collaboration and encourages active, ongoing learning and improvement by defining the best possible team workflow. Kanban can be used in any knowledge work setting and is particularly applicable in situations where work arrives unpredictably and/or when you want to deploy work as soon as it is ready, rather than waiting for other work items [14].

Given Kanban’s approach to start with your existing process and evolve it, there are no roles explicitly called for when adopting Kanban. However, two roles have emerged in practice that serve particular purposes [14].

Service Request Manager: Understands the needs and expectations of customers, and facilitates the selection and ordering of work items at the Replenishment Meeting. This function is often filled by a product manager, product owner, or service manager [14].

Service Delivery Manager: Responsible for the flow of work to deliver select items to customers. Facilitates the Kanban Meeting and Delivery Planning. Other names for this function include flow manager, delivery manager, or flow master [14].

The extreme programming (XP) framework

Extreme Programming (XP) is a disciplined approach for high-quality agile software development focused on speed and continuous delivery. It promotes high customer involvement, rapid feedback loops, continuous testing, continuous planning, and close teamwork to deliver working software at very frequent intervals, typically every 1-3 weeks [15].

The methodology takes its name from the idea that the beneficial elements of traditional software engineering practices are taken to “extreme” levels. As an example, code reviews are considered a beneficial practice. Taken to the extreme, code can be reviewed continuously through the practice of pair programming [15].

The original XP method is based on four simple values: simplicity, communication, feedback, and courage.

XP involves twelve main supporting practices [15]:

- Planning Game
- Small Releases
- Customer Acceptance Tests
- Simple Design
- Pair Programming
- Test-Driven Development
- Refactoring
- Continuous Integration
- Collective Code Ownership
- Coding Standards
- Metaphor
- Sustainable Pace

The Crystal framework

The Crystal methodology is one of the most lightweight, adaptable approaches to software development. Crystal is comprised of a family of agile process models, including Crystal Clear, Crystal Yellow, Crystal Orange and others. Each has unique characteristics driven by several factors, such as team size, system criticality, and project priorities. This Crystal family addresses the realization that each project may require a slightly tailored set of policies, practices, and processes to meet the product‘s unique characteristics [15].

Introduced by Alistair Cockburn, Crystal focuses primarily on people and the interaction among them while they work on an agile software development project. There is also a focus on business-criticality and business-priority of the system under development [16].

Unlike traditional development methods, Crystal doesn’t try to fix the tools and techniques of development but keeps people and processes at the core of the process. However, it is not only the people or the processes that are important, rather the interaction between them that is most important [15].

Several key tenets of Crystal include teamwork, communication, and simplicity, as well as a reflection to frequently adjust and improve the process. Like other agile frameworks, Crystal

promotes early, frequent delivery of working software, high user involvement, adaptability, and the removal of bureaucracy or distractions [16].

The Dynamic Systems Development Method (DSDM) framework

The Dynamic Systems Development Method (DSDM) is an agile approach that grew out of the need to provide a common industry framework for rapid software delivery. Since 1994, the DSDM methodology has evolved to provide a comprehensive foundation for planning, managing, executing, and scaling agile process and iterative software development projects. [15]

DSDM is based on eight key principles that direct the team and create a mindset to deliver on time and within budget. These agile principles primarily revolve around business needs/value, active user involvement, empowered teams, frequent delivery, integrated testing, and stakeholder collaboration. DSDM specifically calls out “fitness for business purpose” as the primary criteria for delivery and acceptance of a system, focusing on the useful 80% of the system that can be deployed 20% of the time [17].

Compromising any of the following principles undermines the philosophy of DSDM and introduces risk to the successful outcome of the project [15].

DSDM’s includes 8 Key Principles:

- Focus on the business need
- Deliver on time
- Collaborate
- Never compromise quality
- Build incrementally from firm foundations
- Develop iteratively
- Communicate continuously and clearly
- Demonstrate control

Business Requirements are baselined at a high level early on in the project. Rework is built into the process, and all development changes must be reversible. System requirements are planned and delivered in short, fixed-length time-boxes – also known as sprints or iterations – and prioritized using MSCW Rules [17].

M – Must have requirements

S – Should have if at all possible

C – Could have but not critical

W – Won‘t have this time, but potentially later

All critical work must be completed in a DSDM project’s defined time-box. It is also important that not every requirement in a project or time-box is considered critical. Within each time-box, less critical items are also included so that they can be removed to keep from impacting higher priority requirements on the schedule [17].

The Feature Driven Development (FDD) framework

Feature Driven Development is a model-driven, short-iteration process that was built around software engineering best practices such as domain object modelling, developing by feature, and code ownership. The blending of these practices that resulted in a cohesive whole is the best characteristic of FDD [15].

Feature Driven Development consists of five basic activities [18]:

1) Development of an overall model
2) Building a feature list
3) Planning by feature
4) Designing by feature
5) Building by feature

FDD begins by establishing an overall model shape, which will result in a feature list. It then continues with a series of two-week “plan by feature, design by feature, and build by feature” iterations. The features are small, “useful in the eyes of the client” results. If they will take more than two weeks to build, then they will have to be broken down into smaller features [18].

FDD’s main purpose is to deliver tangible, working software promptly, repeatedly. The advantage of using FDD is that it is scalable even to large teams due to the concept of ‘just enough design initially’ (JEDI). Because of its feature-centric process, FDD is a great solution to maintain control for incremental and inherently complex agile project management [15].

Main success factors of agile project management in IT

Several attempts at pointing out specific and individual success factors of agile project management in IT development have been made due to the increasing popularity of the methodology. Different propositions of proper implementation of an agile framework have surfaced over the last decade depending on the context in which each study was carried out, however, similar findings point towards the fact that a consensus can be traced.

Abdullah Aldahmash et al. identified 8 main players based on empirical studies and used a classical project management approach to categorize them into 4 factors as follows [19]:

- Technical factors: Delivery strategy, agile development techniques.
- Organizational factors: Organizational culture, communication.
- Process factors: The project management process
- People factors: Team capability and training, customer involvement, top management support.

Abhishek Srivastava et al. Decided to investigate the most important agile success factors influencing quality in the software industry using the analytic network process (ANP). Their conclusion shows that nine criteria, namely: Root cause Analysis, Mutation testing, Cycle time/Test and lean approach, Continuous Integration, Process action, IT governance strategy, organizational change, Effective risk mitigation, feedback/tangible outcomes were chosen that significantly impact the agile testing process. They deepen on their results singling out organizational change and mutation testing as the most critical factor for an enterprise (client) to make sure that they are given preferences while working on a project [20]. Jeff Totten conducted a research question asking to what extent the organizational, human resource, and technical factors predict the perceived level of success when using agile project management methods, and his results show that the variables with the highest score were those that affected the team members and the dynamics of the process itself, such as the ability to react to change, improved project visibility, improved employee engagement, and improved team dynamics/morale. The objectives that scored the lowest were considered as those providing business performance benefits, such as on-time delivery, improved managerial effectiveness, staying within the project budget, and increasing business revenue. The four significant independent variables shown to predict project success in his research study were (1) commitment by management with a clear vision, (2) holding daily stand-up meetings, (3) keeping task sizes small, and (4) using visual management [21]. Carlos Tam et al. carried out a more specific

task and examined the people factors that contribute to agile software project success, creating a model consisting of five people-factors that influence the success based on the classical approach in terms of cost, time, and customer satisfaction. As independent variables, the five people-factors that are proven to be related to agile software development project success are personal characteristics, training and learning, societal culture by Misra et al. (2009), team capability, and customer involvement by Chow and Cao (2008). Results confirm the full mediation relationship between personal characteristics and societal culture to project success. Their Findings also demonstrate how important it is to maximise team capability, as it should be the priority for an appropriate project management implementation, followed closely by customer involvement [22]. Chamika Perera et al. examined client involvement in Sri Lankan software companies to evaluate its contribution to the success of agile projects. Their research concluded that organizational culture is the main foundation for the success of an agile project regarding client involvement as it determines the type and quality of the client-employee professional relationship. The further proposition includes steps to build up a good organizational culture such as creating a respectful workspace, proper communication tools, and the creation of unconventional thinking strategies [23].

This research work evaluates success factors based on the classical project management approach which weights immediate project performance against its main design parameters—schedule (time), budget (cost), scope, and/or quality adding both internal and external stakeholder satisfaction as success measurements to be analysed. As attested upon the aforementioned literature research, the chosen success factors will be people related as all directions point towards them as being the most influential criteria regarding project success. Organizational culture (OC), team capability (TC), personal characteristics (PC), management commitment (MC) and customer involvement (CI) are hypothesized as main contributors towards IT large scale agile development. Moreover, inter-team coordination (IC) is added as an independent variable as large scale agile needs to respond to different coordination mechanisms that do not exist on its common scale counterpart.

Large-scale agile development

Atlassian, the online guide for agile development defines agile at scale as “the ability to drive agile at the team level while applying the same sustainable principles, practices, and outcomes at other layers of the organization” [24]. Large scale agile has been interpreted through different lenses across the literature and even though size is the common denominator that properly defines what

large scale means discrepancies still originate in terms of the number of people, teams, project budget, codebase size, and project duration [25]. Workshop participants at XP2014 provided different definitions for large scale agile development. According to the participants, what is to be considered as large-scale varies depending on the person defining it and on the context, ranging from broad concepts such as multiple teams working together to deliver software artefacts to specific characteristics, like over 50 developers OR 1/2 million lines of code OR more than 3 sites/time zones [26]. Paasivaara et al. Considered a project with seven development teams and 40 people as large scale [25] Berger and Beynon-Davies added a project cost approach, arguing that a project which costs over 10 million GBP along with a team size of 50 people counts as large scale [27]. On the other hand, Petersen and Wohlin suggest that a project having over 5 million lines of code classifies as large scale one [28]. A project duration of two years with a scope of 60 to 80 features is deemed as large-scaled by Bjarnason et al [29]. The large scale was also measured by the number of collaborating and coordinating teams, as done by Dingsøyr et al. who categorized a project as large scale with ranges of two to nine collaborating teams, introducing the concept of very large scale for over 10 collaborating teams [30]. Several additional studies discussing large-scale agile software development and their interpretations of large-scale were also taken into account. All of these referring to the number of people involved. Initial agile propositions showcase that standard agile procedures work great for lesser groups of ten to fifteen individuals [31] who share a physical dependency. However, Fowler considers the Crystal methodology to be suitable for up to 50 people [32], a number that has been reported as seen by practitioners and researchers as the size of the largest organization suitable for agile.

To simplify operations in this research paper, large scale agile development will be defined as any process that involves more than 50 people or at least 5 different teams developing together the same project using an agile method.

Large-Scale agile frameworks

Some organizations have successfully applied agile methods to their escalation attempts to allow themselves to create, react to, embrace, and learn from change while improving their value proposition concerning their customer. These endeavours resulted in the creation of different frameworks for scaling agile. Scrum has long been the leader and most prominent bodywork for team-level agile, adaptations such as Large Scale Scrum (LeSS) and Scrum at Scale have been used [33], however particular challenges opened up space quickly filled by other methodologies, like the

Scaled Agile Framework (SAFe) and Disciplined Agile Delivery (DAD) [34]. A new phenomenon has also taken place, digital-native organizations have been born agile and thus turned their performing process into established large scale agile frameworks named after the companies themselves, mainly Spotify and Netflix [35] Each of the previously mentioned frameworks incorporates predefined workflow patterns and routines and is supported by an ever-increasing set of tools:

LeSS: Essentially regular scrum applied to large-scale development. LeSS is based on the idea that scaling frameworks should be minimalistic (i.e. include fewer rules, roles, and artefacts) to drive success. LeSS provides two different large-scale Scrum frameworks. LeSS, up to eight teams (of eight people each), and LeSS Huge, up to a few thousand people on one product. In LeSS all Teams are in a common Sprint to deliver a common shippable product, every Sprint [36].

Scrum at Scale: an extension of the Scrum framework. Scrum@Scale is generally adopted by organizations that have already implemented Scrum successfully at the team level and are looking to spread it throughout the organization. The main goal is to align growing organizations around one common and shared set of goals. Coordination is managed through a Scrum of Scrums, which is comprised of Scrum Masters from each team, and a MetaScrum made up of product owners [37].

SAFe: a set of organization and workflow patterns for implementing agile practices at enterprise scale. It was formed around three primary bodies of knowledge: agile software development, lean product development, and systems thinking. SAFe promotes alignment, collaboration, and delivery across large numbers of agile teams [38].

DAD: a learning-oriented process decision framework for IT solution delivery. It provides a solid foundation from which to scale agile solution delivery within enterprise-class organizations. DA utilizes Scrum and Kanban, along with transformation knowledge in areas like HR and finance, governance, DevOps, portfolio management and more [37].

Spotify: is a people-driven, autonomous framework for scaling agile. It stresses the importance of culture and networks and provides an example for dealing with multiple teams in a product development organization [39].

Nexus: is a framework that drives to the heart of scaling by minimizing cross-team dependencies and integration issues. It uses Scrum as its building block. Nexus is a framework consisting of roles, events, artefacts, and rules that bind and weave together the work of approximately three to nine

Scrum Teams working on a single Product Backlog to build an Integrated Increment that meets a goal [40].

The level of success varies between enterprises and models chosen, some organizations were able to become industry leaders while being globally distributed with agile large-scale development whereas some others failed at the initial framework choice and tried follow-up alternative combinations, or abandoned them completely [34]. Failures are mainly due to the misunderstanding of framework concepts and routines due to the specificity of the context in which they were created, causing companies to run out of proper guidance for outside utilization [35].

Research methodology

Research process

After careful literature research, the main success drivers of IT agile development were selected as seen in figure 2. To evaluate their effectiveness when it comes to scalability and large scale adaptation these proven success factors are then turned into a survey sent to agile practitioners in large scale set-ups for them to answer. Collected data is then studied via a series of statistical and qualitative analysis further explained in the next section. Finally, conclusions and propositions on how IT companies should approach large scale agile development are stated which serve as first base recommendations that may help new ventures and diminish the existing research gap on large scale agile development/implementation.

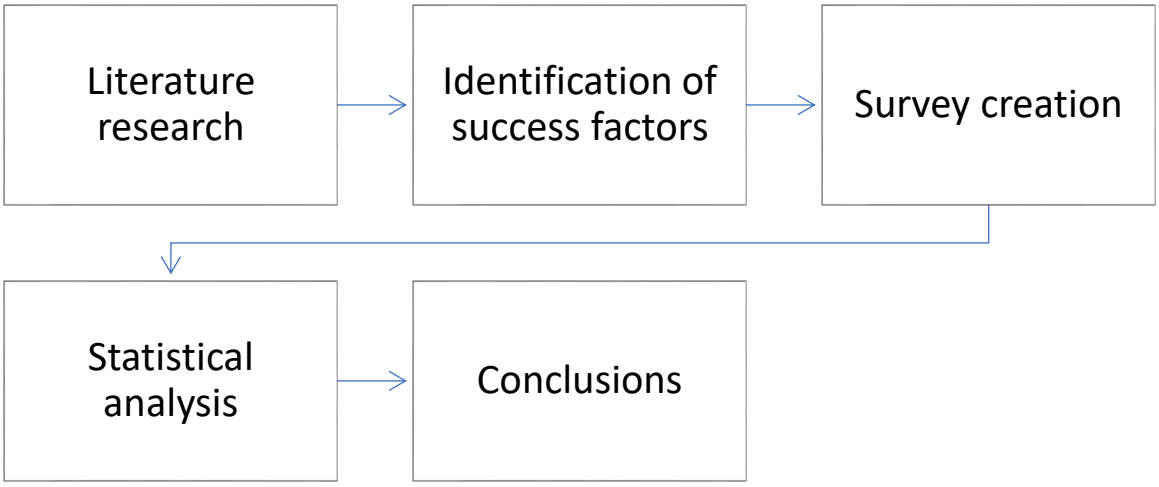


Figure 2: Research steps/process.

Research model

A proposition based on the research model carried out by Carlos Tam et al. [22] resulted in seven theoretically well-grounded variables grouped into the proposed research model as can be seen in figure 2. Large scale agile IT development project success, which is this study's dependent variable, is defined in terms of time (i.e. on-time delivery), cost (i.e. on or under budget), customer satisfaction (i.e. overcoming customer's expectations regarding quality and/or scope of the project) and stakeholder contentment (i.e. assuring the well-being and optimum relationship of both internal and external project stakeholders). As independent variables, six people-factors that are proven to be related to agile software development project success were selected. The factors: organizational culture, team capability, personal characteristics, management commitment and customer involvement were taken from already proven studies that assure their criticality regarding agile development, whereas inter-team coordination is added as a way to respond to the uncertainty and high degree of interdependence resulting from project scalability. The combination of both common scale success factors adding factors that account for new scalability challenges is useful for understanding the impact of project success.

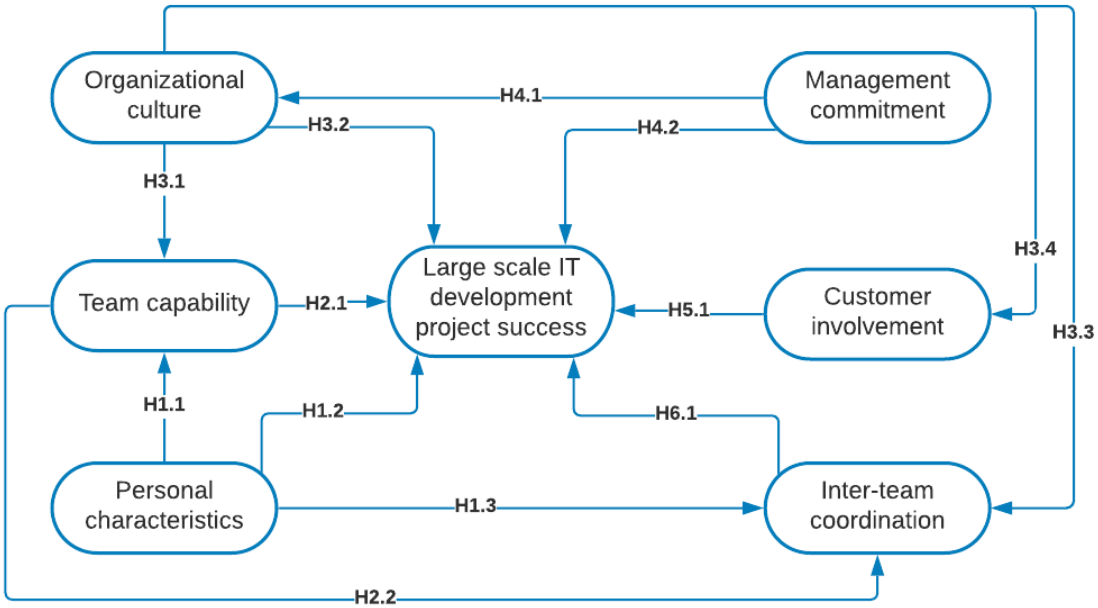


Figure 3: Research model.

Success factors

Personal characteristics

Shipper et al. stated that personal characteristics comprise many qualities that are not cognitive, such as communication skills, empathy, and resiliency [41]. Wouter Aghina et al. researched to identify specific personality traits and values that help agile teams bloom, getting to the conclusion that emotional stability, agreeableness, extroversion, and the ability to handle ambiguity are principal key players in the success of an agile implementation and team efficiency [42]. According to the International Project Management Association, communication and interpersonal skills, honesty, collaborative attitude, and working with others are part of the necessary set of personal and interpersonal competencies for an individual to achieve good performance in a team project, programme or portfolio, leading to its success [43], and thus the next hypotheses are presented:

H1.1: Personal characteristics positively impact team capability on large scale agile IT development.

H1.2: Personal characteristics positively influence a large scale agile IT development project's success.

H1.3: Personal characteristics directly affect inter-team coordination on large scale agile IT development.

Team capability

McKinsey & Company states that much of the road to success regarding agile development depends on the talent, whether developed or recruited [42]. A specific set of capabilities in terms of skill, technical competence and expertise are related to the effectiveness of agile process implementation. How knowledge is used, along with the conditions that allow teams to accomplish their tasks all refer to the capability of the team itself [44]. Chow and Cao added some extra attributes, such as team members' motivation and commitment, agile knowledgeable managers with an adaptive management style, and proper vision of technical training to the project team [45]. Based on the aforementioned succeeding hypothesis are brought up:

H2.1: Team capability directly impacts the success of large scale IT agile development.

H2.2: Team capability is a factor that positively affects inter-team coordination in large scale IT agile development.

Organizational culture

An organization's culture refers to how people must behave within the organization. It consists of a shared belief system that permeates all levels of an organization or subunit and by which people and workgroups actions are influenced [46]. Organizational culture is established by the top management and then communicated and reinforced through various methods [47]. Agile practitioners and researchers affirm that the organizational culture in which the agile method is embedded has an impact on its use [48], prompting this research work to propose the following hypotheses:

H3.1: Organizational culture directly determines the team capability in large scale IT agile development.

H3.2: Organizational culture positively influences a large scale agile IT development project's success.

H3.3: Organizational culture positively acts on inter-team coordination in large scale IT agile development.

H3.4: Organizational culture directly impacts the level of customer involvement in large scale IT agile development.

Management commitment

Top management needs to focus more on the actual project vision and why they chose to go on board with it [49]. Santiago Comella et al. stated that agile development must be a complementary effort from the technology organization as well as senior business executives, asking the latter to include it on their agendas, thereby signalling the importance of making the required technology and cultural changes. The top team's attention helps to point out that software development entails an almost constant coalition between business and IT groups, and it requires widespread acceptance of a test-and-learn approach [50]. Based on the previous statements the following hypotheses are presented:

H4.1: Management commitment promptly impacts the organizational culture in large scale IT agile development.

H4.2: Management commitment positively influences a large scale agile IT development project's success.

Customer involvement:

User involvement and lack thereof has substantially been regarded as one of the most determinative factors when evaluating project success. It is the appropriate meeting of the user needs that ultimately qualify the service offered [51]. In large scale agile development the customer support involves several stakeholders in the customer company, and both the attitude and focus of the whole customer organization towards the project [52] as well a co-location within the development team [51]. The importance of customer involvement results in the proposition of the following hypothesis:

H5.1: Customer involvement positively impacts a large scale agile IT development project's success.

Inter-team coordination

Coordination mechanisms require different approaches for different scales. Major challenges arise when trying to manage various teams that are working towards a common goal due to high uncertainty in tasks, a high degree of interdependence between tasks and the large number of people involved [53]. Direct mutual adjustment among every individual member is no longer an

effective tool for multi-team systems, which are too large and specialized, thus positioning inter-team coordination as one of the main topics in the agile development research agenda [54], and encouraging this present research work to propose the following hypothesis:

H6.1: Inter team coordination positively affects a large scale agile IT development project's success

Results

Data

The target demographic comprised any individual who was (or had been) involved in at least one *large scale* agile software development project, either as a team member or direct stakeholder (i.e. agile coach, client, etc.). Respondents were allowed more than one answer accounting for different agile methodologies in case of multiple personal experiences with different agile frameworks on various projects, as long as they were all considered *large scale* setups. English was selected as the survey language and no geographical limitations were set.

A 5 point Likert Scale survey, ranging from totally disagree (1) to totally agree (5) was used to measure the respondents' agreement level towards each presented item, except on the perceived level of project success, which ranged from (1) very unsuccessful to (5) very successful.

An initial survey draft was sent out to knowledgeable agile practitioners working in mature agile environments via *LinkedIn: Agile and Lean Software Development* as a way of helping shape out a final version in terms of objective and clear language as well as goal clarification. Feedback provided was incorporated and the ultimate edition was shared online using various networks: LinkedIn groups, Universidad de Antioquia's and Universidad Pontificia Bolivariana's graduate networks, and personal contacts. Answers were openly received from November second 2020 to December 11th 2020. A total of 86 respondents completed the request which corresponds to an 86 % reach of the original 100 mark intention. One answer was deemed invalid due to inconsistency, leaving 85 proper answers left for analysis and result interpretation.

Survey results show that more than half of the participants (58.8%) were young adults with ages ranging from 20 to 29 years, followed by 29.95 % of adults with ages ranging from 30 to 39 years, and a 7.1 % comprised of mature adults with ages ranging from 40 to 49 years; the remaining percentage represents people of 50 + years. Most of the survey respondents (82.4 %) identified as

male. Regarding location parameters a vast 58.8% of answers were collected from South America, followed by European answers with 28%; the rest of the respondents were scattered across different regions around the world. Referring to educational status most respondents (around 94.1%) attended higher education; with more than half being people who got a bachelor's degree (56.5%). The remaining 5.9% are individuals who completed 12th grade or equivalent. Agile usage experience showed an evenly distributed tendency; 44.7% of respondents reported having 1 to 2 years of agile experience, while a 44.7% reported more than 3 years, and 10.6% reported having less than one year of experience. An overwhelming 74.1% of individuals acknowledged using Scrum at Scale as the implemented agile framework, while SAFe remained second most used, with 12.9% of respondents identifying it as the selected agile framework. The most assumed role by individuals was that of the developer, comprising 56.5% of all respondents, product owner and tester directly follow it, with 16.5% and 11.8% respectively. Details and summarized information is displayed in tables 1 and 2.

Table 1. Demographic data.

Age	**N°**	**Education**	**N°**
21-29	50	High School	5
30-39	25	Bachelor's Degree	48
40-49	6	Master's Degree	30
50 +	4	PhD or higher	2
Location	**N°**	**Agile experience**	**N°**
South America	50	< 1 year	9
North America	6	1–2 years	38
Europe	24	3–5 years	21
Central America	1	5 + years	9
Asia	4		

Table 2. Demographic data: frameworks and roles.

Agile framework	**N°**	**Agile role**	**N°**
Scrum at Scale	63	Developer	48
LeSS	4	Tester	10

SAFe	11	Product owner	14
DAD	1	Scrum Master	6
Kanban	1	External stakeholder	7
Other	5		

Reliability analysis

A total of 17 questions were made to evaluate each proposed factor, while four variables (cost, time, stakeholder contentment and customer satisfaction) were used to quantify perceived project success. Results were tabulated together and are represented graphically using bar graphs for easier understanding as seen below in the appendix, section 2.

The resulting Likert scale data was analysed at the interval measurement scale given that the survey was composed of a series of multiple Likert-type items that represent similar questions combined into six single composite score/variable whose relationship to the final dependent variable is to be studied [55]. Collected data was summarized into simple numerical form according to descriptive statistics; scores from each participant were added up to get the total score for each of the six major survey factors as well as the perceived level of project success as can be seen in table 3. Following allegations made by Dr Geoff Norman, who provided evidence on the use of parametric tests with ordinal data, (such as data from Likert scales), stating that parametric tests tend to give "the right answer" even when statistical assumptions—such as a normal distribution of data—are violated [56], the mean was calculated as a form to evaluate central tendency and the standard deviation was selected as a measure for variability [55]. Given this dissertation objective on trying to prove people factors (which can be interpreted subjectively) and perceived project success, a non-concrete factor, with 6 variable capturing the concept being assessed, the Cronbach alpha calculation was used to provide evidence for the different related bundles of the scale being sufficiently intercorrelated and that the grouped items measure the accurate underlying level of project success [56]. Calculated values are represented in tables 4 and 5.

Table 3. Comprised summarized survey variables.

Case Processing Summary		
	N°	Marginal Percentage

PS	unsuccessful	2	2.4%
	neutral	16	18.8%
	successful	49	57.6%
	very successful	18	21.2%
PC	neutral	13	15.3%
	agree	50	58.8%
	totally agree	22	25.9%
TC	disagree	2	2.4%
	neutral	6	7.1%
	agree	50	58.8%
	totally agree	27	31.8%
MC	totally disagree	1	1.2%
	disagree	3	3.5%
	neutral	18	21.2%
	agree	48	56.5%
	totally agree	15	17.6%
OC	disagree	3	3.5%
	neutral	14	16.5%
	agree	50	58.8%
	totally agree	18	21.2%
IC	disagree	1	1.2%
	neutral	21	24.7%
	agree	36	42.4%
	totally agree	27	31.8%
CI	totally disagree	1	1.2%
	disagree	4	4.7%
	neutral	18	21.2%
	agree	46	54.1%

	totally agree	16	18.8%
Valid		85	100.0%
Missing		0	
Total		85	

Table 4. Latent variables: means and standard deviations (SD).

Survey item	**Mean**	**SD**
Personal characteristics	3.99	0.578
Team capability	4.18	0.629
Organizational culture	4.07	0.790
Management commitment	3.88	0.737
Customer involvement	3.84	0.803
Inter-team coordination	3.85	0.778
Large scale IT development project success	3.90	0.663

*Table 5. Latent variable: CA (*Cronbach alpha*).*

Survey bundle	***CA***
PC-TC-OC-MG-CI-IT	0.87
Large scale IT development project success	0.72

Cronbach´s alpha calculations are done before hypotheses testing and examination purposes to confirm model validity [57]. Alpha is based on the 'tau equivalent model' which assumes that each

test item estimates the same attribute on an identical scale [58]. Different reports about the acceptable values of alpha ranging from 0.70 to 0.95 exist in the literature [59]. Results show that the implemented model has internal consistency, having both survey bundles scoring values inside the proper reported range, meaning that there is a high degree to which the test measures what it claims to measure (people factors and perceived level of project success). A high correlation between test items ensure a high alpha value, however, test length must also be taken into account, as alpha is directly related to it. Low alpha values can be obtained due to a low number of questions even if the constructs are homogeneous [58], which is the reason why the whole bundle of people factors were taken into account in the calculations shown above.

Regarding individual factor reliability, the tau-equivalence assumption is violated due to the low number of questions asked for some of the constructs, rendering alpha reliability values underestimated [58]. Inter-item relations and Item-total correlation amongst items within a specific construct are then used to properly assess that each item has a good correlation with other items as well as an acceptable correlation within the total score for each factor evaluated [60]. Literature reports mention that the average inter-item correlation for a set of items should be between 0.15 and 0.6, suggesting that while there is a reasonable homogeneity, they are different enough to not commit redundancy [61], [62]. Results displayed in table 6 show inter-item statistics ranged from 0.043 to 0.571. The first construct: *Personal characteristics,* was the only one containing lower than acceptable inter-item correlation values, however, results were still all positive, an indication of the conceptual fit of the items [60]. This situation can be explained as a result of the subjective content domain that personal characteristics represent, making it difficult to find a low number of representative enough items. Corrected item-total correlation coefficients were found to be acceptable as well, revealing that the correlation between the scores of one omitted item and the total score of all other items was high enough to consider each item a valid one [63]. The item-total correlations were seen to be within 0.254 to 0.60, with a 0.42 average, and most being placed above the generally reported minimally required cut-off value of 0.3 [64]. 2 values (Personal characteristics: 0.254, Team capability: 0.287) are left outside of the reported acceptable range, however, according to Cristobal et al. [65], “for exploratory studies 0.20 is an acceptable value for item-total correlation”, which encourages this study to keep the totality of previously proposed items for further analysis as the overall results are considered to be valid and reliable.

Table 6. Summary of Item Reliability Analysis for each survey construct.

Survey construct	N° of items	Inter-item correlation range	Corrected Item-Total correlation range
Personal characteristics	4	0.043- 0.419	0.254- 0.478
Team capability	3	0.191-0.513	0.287-0.547
Organizational culture	2	0.353	0.353
Management commitment	3	0.395-0.571	0.451-0.585
Customer involvement	3	0.389-0.55	0.465-0.60
Inter-team coordination	2	0.315	0.315

Statistical analysis

Due to the characteristics of the data, non-parametric tests, which do not require normality and homogeneity of variance assumptions must be used to properly prove the previously proposed hypotheses. To examine if there are statistically significant differences between each answered level of agreeableness on a designated people factor and the different levels of perceived project success a Kruskal-Wallis test was conducted [66]. This test, however, delivers conservative results, as it assumes independent data as nominal and does not inform the order in which the people factors differ, hence not a suitable choice to attain the insight of orders of median levels of perceived project success in which the scaled data from each people factor is different [67]. To overcome this, the Jonckheere-Terpstra test was chosen to prove significant trends existed in the collected data, that is, whether an increase in one independent ordinal variable results in an increase or decrease in another ordinal variable [68]. The Jonckheere-Terpstra test will, essentially, help showcase whether and how large scale IT development project success, measured on an ordinal scale from 0 to 5, differs based on the people factors possessed by the project teams, which have five ordinal independent groups (0 to 5) as well.

Kruskal-Wallis test

Six Kruskal-Wallis test calculations were carried out via SPSS v. 21.0. Mean ranks, *p*-values and overall test statistics are reported on tables 7 and 8. To determine whether any of the differences between the medians are statistically significant, each resulting *p*-value was compared to a significance level of 0.05 to assess the null hypothesis [69]. Results prove that there is a significant difference between factor agreeableness level groups, with the highest *p*-value being 0.005. Mean

rank values provide qualitative information on the effects of the level of factor agreeableness towards perceived project success [70], as high mean ranks translate into high values of perceived project success, and, in all reported cases, groups who selected totally agree also resulted with the highest mean ranks of perceived project success. Kruskal-Wallis H. values could be interpreted as unusually large, however, this can be explained as the populations studied are strongly skewed (asymmetrical) [71], which helps make sense of the larger differences between the groups that are being compared.

Table 7. Kruskal-Wallis test ranks.

Independent variable	**Group**	**N°**	**PS Mean Rank**
PC	neutral	13	16.62
	agree	50	43.16
	Totally agree	22	58.23
TC	disagree	2	10.50
	neutral	6	25.25
	agree	50	37.72
	Totally agree	27	59.13
OC	disagree	2	1.50
	neutral	11	52.23
	agree	32	32.92
	totally agree	40	49.80
MC	Totally disagree	1	1.50
	disagree	3	32.17
	neutral	18	35.94
	agree	48	41.60
	Totally agree	15	60.87
CI	Totally disagree	1	1.50
	disagree	4	41.00
	neutral	18	28.56
	agree	46	43.17
	Totally agree	16	61.84

IC	disagree	1	43.00
	neutral	21	33.86
	agree	36	39.25
	Totally agree	27	55.11

Table 8. Kruskal-Wallis test statistics.

	PS		
Independent variable	***Kruskal-Wallis H***	***df***	***Asymp. Sig.***
PC	29.320	2	<0.001
TC	25.734	3	<0.001
OC	18.372	3	<0.001
MC	16.268	4	0.003
CI	23.155	4	<0.001
IC	12.891	3	0.005

Jonckheere-Terpstra test

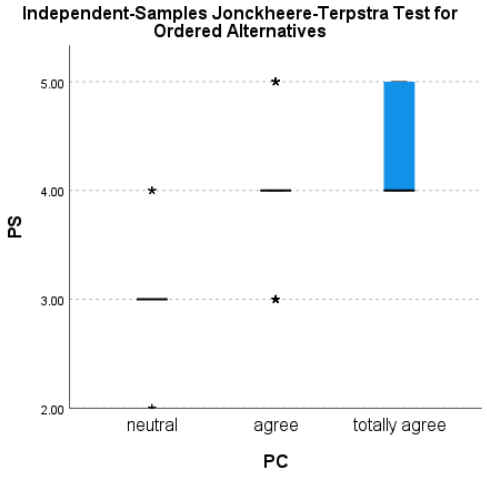
Individual tests via SPSS v. 21.0. Were carried out for each factor reaching a total of six processes as the Jonckheere-Terpstra test only allows for the analysis of pairs of variables between one independent variable and one dependent ordinal variable [72]. A null hypothesis indicating that the distribution of perceived project success is the same across the different agreement categories of people factors was set out to be tested. The significance of a potential trend was then determined by calculating a Jonckheere-Terpstra test statistic, followed by calculating an appropriate p-value. Results demonstrated the rejection of the null-hypothesis with the lowest *p*-value being 0.026, demonstrating higher median scores of perceived project success with higher levels for most of the people factors [73]. Table 9 summarizes test results.

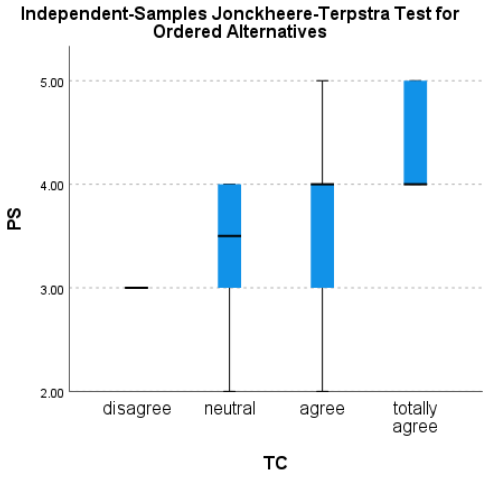
Table 9. Independent-Samples Jonckheere-Terpstra Test for Ordered Alternatives summary.

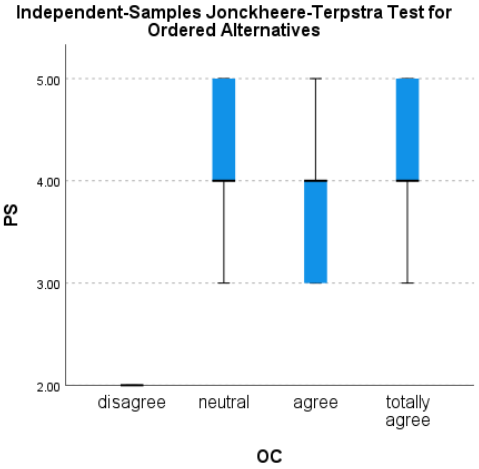
	Test Statistic	**Standard Error**	**Standardized Test Statistic**	**Asymptotic Sig. (2-sided test)**
PS across PC	1571.000	102.999	5.368984	<0.001

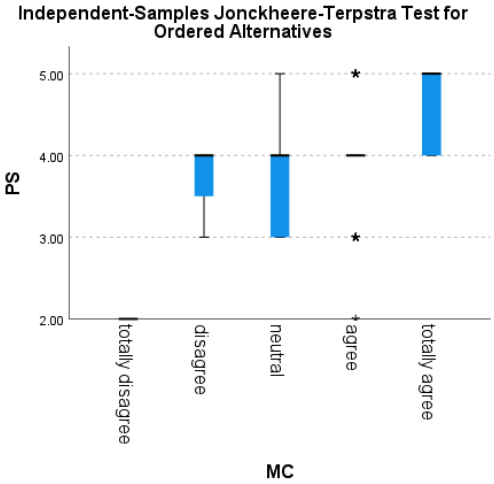
PS across TC	1511.000	102.230	5.106110	<0.001
PS across OC	1357.500	107.212	2.225	0.026
PS across MC	1474.500	104.934	3.659440	<0.001
PS across CI	1611.000	106.254	4.545718	<0.001
PS across IC	1567.000000	109.528	3.446604	<0.001

Figure 3 illustrates the results of the six Jonckheere-Terpstra tests. The graphs depict the trend-shaped progression and allow for a better understanding of how perceived project success increases as the personal characteristics, team capability, management commitment, customer involvement and inter-team coordination increase. Organizational culture, however, does not display a growth pattern, as high project success median scores can be seen associated with both “neutral” and “totally agree” grouping levels from organizational culture, while lower values can be seen associated with the “agree” grouping levels, indicating that perceived project success does not necessarily depend on the focus of a company’s organizational culture towards agile practices. Graphical analysis of the medians from all the other five people factors showcase a trend towards growth going through the people factors scaled levels of agreement (i.e. from 1 to 5) especially at the highest levels of agreement, however, if you take into account whole sets of data and its variations the impression becomes less clear, meaning that the Jonckheere-Terpstra test only allows for a general behavioural conclusion. The population can also be seen as highly asymmetrical, with a strong variability between groups, being customer involvement and management commitment the only factors with a population distribution across all 5 agreement groups.


Independent-Samples Jonckheere-Terpstra Test for Ordered Alternatives
PS
5.00
4.00
3.00
2.00
neutral
agree
totally agree
PC


Independent-Samples Jonckheere-Terpstra Test for Ordered Alternatives
PS
5.00
4.00
3.00
2.00
disagree
neutral
agree
totally agree
TC


Independent-Samples Jonckheere-Terpstra Test for Ordered Alternatives
PS
5.00
4.00
3.00
2.00
disagree
neutral
agree
totally agree
OC


Independent-Samples Jonckheere-Terpstra Test for Ordered Alternatives
PS
5.00
4.00
3.00
2.00
totally disagree
disagree
neutral
agree
totally agree
MC

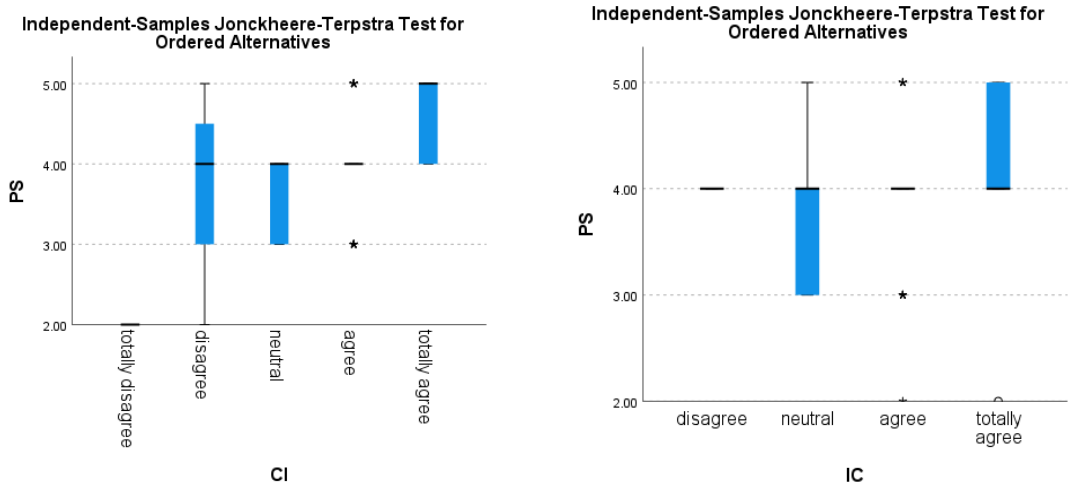


Figure 4: Illustration of Jonckheere-Terpstra tests on perceived project success.

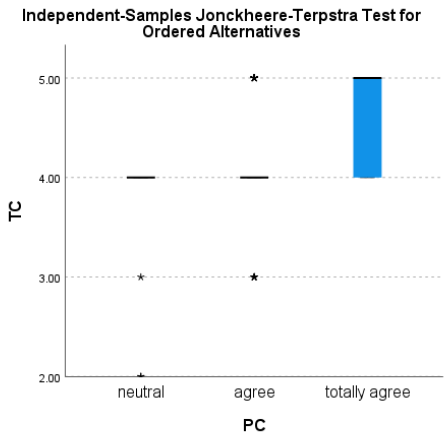
Inter factor dependency

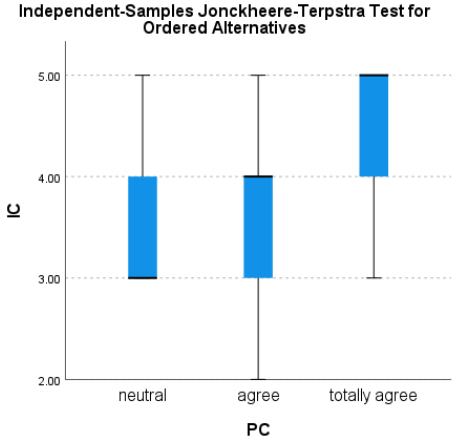
Seven additional Jonckheere-Terpstra tests were carried out to analyse inter factor dependency as stated on hypotheses 1.1, 1.3, 2.2, 3.1, 3.3, 3.4 and 4.1. A null hypothesis indicating that the distribution of the now considered dependent factor variables is the same across the different agreement categories of people factors was set out to be tested. All null hypotheses turned out to be rejected, with the highest p-value being 0.004, meaning there is a significant level of different distributions across the different categories analysed as can be seen in table 9.

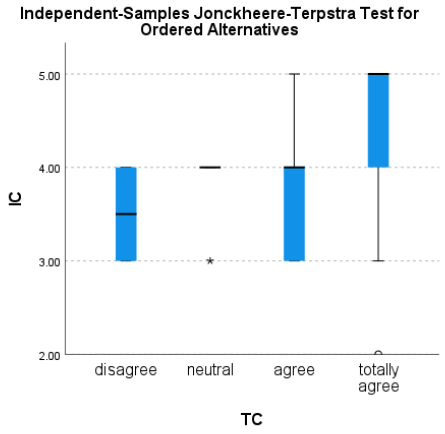
Table 9. Independent-Samples Jonckheere-Terpstra Test for Ordered Alternatives summary (Inter factor dependency).

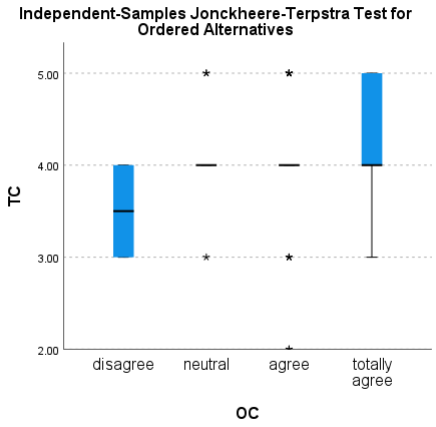
	Test Statistic	**Standard Error**	**Standardized Test Statistic**	**Asymptotic Sig. (2-sided test)**
TC across PC	1494.000	101.145	4.706	<0.001
IC across PC	1453.000	108.364	4.014	<0.001
IC across TC	1321.500	107.555	3.091	0.002
TC across OC	1418.500	105.281	2.845	0.004
IC across OC	1596.000	112.796	4.229	<0.001
CI across OC	1440.000	109.424	2.934	0.003
OC across MC	1502.500	108.065	3.813	<0.001

Trend analysis is studied with the graphical aid presented in figure 4, which illustrates the results of the seven Jonckheere-Terpstra tests conducted. No clear growth trends can be appreciated alongside the distribution of all levels of factor agreement. Growth trends in the last stage of factor agreement regarding medians can be seen in the TC-PC, IC-PC, IC-TC, TC-OC and OC-MC relationships, indicating that, even though a general positive behavioural conclusion can be proposed, there is a lesser level of influence between the variables. The IC-OC and CI-OC relationships show changes in the overall trend due to variances in the population distribution, meaning there is no clear proof that a company´s organizational culture positively influences its inter-team coordination mechanisms nor how consumer centred the project can be.









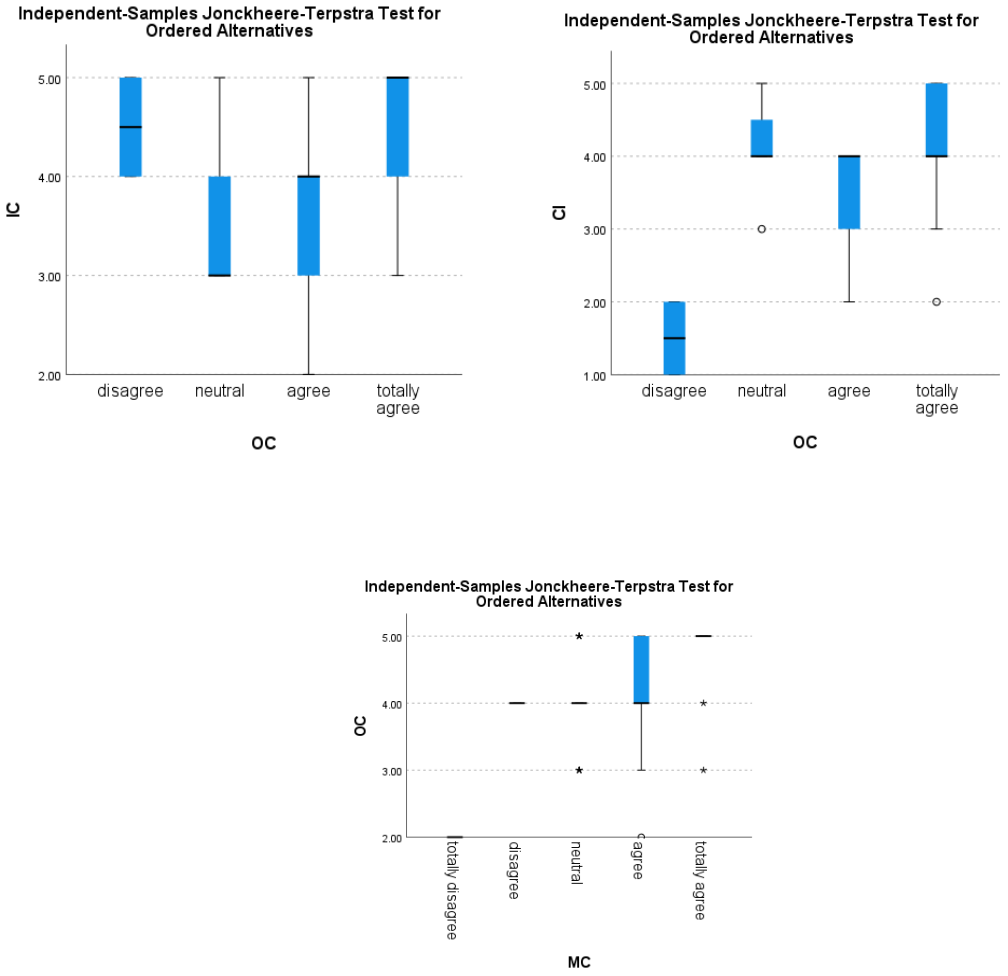


Figure 5: Illustration of Jonckheere-Terpstra tests on people factors.

Discussion

After properly identifying a series of already reported success factors of an agile implementation in regular scale setups, this study aimed to translate the level in which the factors proved to be key in the success of the large scale counterpart, with new insights on inter-team coordination and its relationship towards the previously established variables. A combination of six factors was used to explain large scale agile IT development success. After performing non-parametric tests to prove group population differences and underlying trends it is now possible to provide a concise answer to the research question. Results show that at a behavioural level almost all hypotheses, except for H3.2, H3.3 and H3.4 are supported.

The proposed research model validates the proportional relationship between personal characteristics, technical capabilities, management commitment, customer involvement, inter-team coordination and successful implementation of agile development. Model results, however, do not display a growth trend between organizational culture and project success. The model adds inter-team coordination as a key large-scale factor, establishing it as dependant on personal characteristics, team capability and organizational culture. Results display organizational culture does not directly affect inter-team coordination, meaning that low values of a company´s organizational culture focus on agile do not necessarily translate into low values of inter-team coordination. Personal characteristics and team capability are indeed corroborated as variables that affect a large scale projects' inter-team coordination ability on a behavioural level, as a general growth trend was identified for both combinations of variables.

Based on the results, it can be argued that even though the factor inter-team coordination leads to large scale agile IT development project success further research should be carried out to prove the exact level of criticality, as literature reports state that scaled agile should take a particular focus on improved interaction between teams which in turn speeds up the delivery process, and timely completion of projects [74]. Results point towards team capability and personal characteristics as having a greater impact on agile IT development project success. In essence, to amplify the chances of project success, team capability should be the priority, followed closely by personal characteristics; a conclusion that goes in line with previous success stories of scaled agile, which report that for an effective agile transformation efforts must centre on choosing "the right people" [75], and that training all employees and team members in popular agile certification courses that focus on scaled agile methodology will increase work efficiency and interaction as a whole unit, making all processes and projects faster and more accurate [74]. Results also follow newly released insights on scaled agile which focus on customer satisfaction, delivering the right products and, most importantly, making sure the agile teams are engaged and understand the business they are a part of [76], meaning that taking special considerations on making agile teams the centre of attention will ultimately not just produce better products and services, but also produce more engaged employees.

The lack of direct effect from organizational culture towards project success may be justified by the measured items in the survey: (1) the company's organizational culture encouraged communication and assertiveness between teams and (2), the company's organizational culture supported an agile

approach during project development, whose immediate repercussions in regards of people behaviour can be overshadowed by more easily identifiable factors, given that a company´s organizational culture does not dictate but rather influences the actions of people and work groups that were permeated by it. Ultimately, results lead to the suggestion that even though certain types of company values would negatively contribute and make it harder to successfully use an agile method, the degree to which the organization's actual values affect the process of system development remains secondary when compared to the remaining factors evaluated.

To make informed decisions previous results should be taken into account as they carry particular managerial implications. Agile teams should be composed of aggregable and honest people, focussing on the individual's communication strengths first and foremost, making sure that they are in line with the requirements of the position, as a specific fit between a type of project and the team's personal characteristics ensures positive repercussions on the team's capabilities and improves inter-team coordination, ultimately optimizing project outcome.

Proper technical training should be a company priority. Continuous learning via specialized programs and stimulating informal conversations should be encouraged as a way to ensure not only skilful thinking but also to increase team synchronization, along with tools that help to keep high levels of motivation and commitment towards project success.

Top management should be committed to project development, assuring that team facilitators are knowledgeable in agile processes and principles, emphasising implementing an adaptive management style that centres on providing the appropriate resources for agile implementation. Agile teams should not be kept apart from the business-oriented units as more often than not agile teams lack initiative and require capabilities possessed from other company segments to develop and execute a new initiative [75].

Project development should be built around the customer at all times, making sure of their direct involvement in the project while verifying their expectations are met. Having multiple representatives on-site working hard and full-time as a member of the project team is highly encouraged, and the presence of at least one representative throughout the whole development process must be assured. Agile bases itself on frequent software delivery making the customer role a crucial one. The more involved a customer is, the more satisfied he may be with the project.

Common agile characteristics such as frequent feedback, constant change and self-organization will require efficient communication between the project's members and with the customers. Communication mechanisms should be geared towards the interconnection of multiple teams, making sure that all participants are aware of the overall project status at all times so that shifts in scope requirements become less complex and easier to tackle. The communication must be direct and assertive since direct communication will eventually lead to the reduction of ambiguity and the clarification of goals.

In brief, agile IT development should be built around committed, talented, motivated and professional people. Top management and team facilitators should provide resources that help to get a big-picture view of problems, understanding distinctions in different locations or clients and anticipating rapidly shifting competitive landscapes. Theoretical information provided should be a starting point so that each scaled agile endeavour identifies the importance of which factors bring the highest success and helps setting out scalability mechanisms that correctly prioritize them, thus improving project outcomes.

Limitations and future work

The inherent complexity of the measured variables represents a challenge, as there are certainly more items that could have been used to characterise the factors better, even at a quantitative level. Future research should focus on validating the model and making sure that items correspond to the variables that want to be measured. Population distribution and size also play a huge part in the research limitations, as exogenous variables may affect survey respondents and prompt them to answer in a specific manner. Highly experienced and matured agile practitioners gave higher success qualifications to the project they were involved in, having that 88.23 % of the total referred to the project as either successful or very successful, whereas 81.71 % and 72 % qualified the project on the same groups for medium and low experience levels respectively, meaning Lower years of experience in agile usage translated into lower perceived project success qualifications. Geographical location may also play a role in the level of perceived project success, South America containing the highest perceived project success qualification, with 26 % of its respondents agreeing that the projects they were a part of were very successful, while only 12.5% of European respondents agreed on the same level; none of the remaining regions provided the same level of project success. In regards to the evaluated factors, most people were distributed on neutral and

positive levels of agreement with no clear pattern regarding location, agile experience or age, as there was no sufficient data to prove correlation, so further research on bigger populations with relation to exogenous variables should be carried out.

Conclusion

Agile IT development has gained enormous popularity since its official introduction in 2000 thanks to its ability to move easily and quickly respond to customer needs. Research has lately been focused on identifying the factors that influence the success of agile software development and available bibliography reports human capital as a key success factor for agile projects, consequently making it the main focus of this research proposal.

A model-based in the compilation of 6 success factors proven to be influential for agile software development project success and their validity was tested via SPSS v. 21.0. Data from 85 agile professionals from a variety of business areas was collected using a Likert Scale survey that measured the respondents' agreement level towards 21 survey items that were then comprised into the 6 proposed success factors. Results emphasize that personal characteristics and technical capabilities exert great influence in the variance of agile IT development success. Management commitment, customer involvement and inter-team coordination were also proven as factors that should not be overlooked, as they exhibit some level of impact on the success of the project. There was no indication that organizational culture is an important factor in the large scale context, as it is ultimately eclipsed by factors with a higher level of a direct effect on the team.

Fundamentally, findings suggest that an agile organization can increase its value proposition when selecting teams of talented and highly capable people while promoting customer involvement and collaboration since these factors are more likely to lead an agile software development project to success. Theoretical implications found in this research work as a guide for companies that are looking into scaling their agile IT development practises, providing insight on how agile teams are embedded in broader collaborative networks that can be constituted so that both internal and external stakeholder satisfaction is maximized, averting potential disruptions, and achieving objectives better and faster.

Appendix

Section 1: Survey items

Table 1: Tested success factors in large scale agile development.

Item	Construct	Adapted from
Personal characteristics	The project teams consisted of people with strong interpersonal and communication skills.	[43]
	The project teams consisted of people with the ability to handle ambiguity.	[42]
	The project teams consisted of people with a high perceived sense of emotional stability.	[42]
	The project teams consisted of people who were agreeable and honest.	[43]
Team capability	The project teams were comprised of technically skilled and competent people (i.e. programming, design, subject matter,).	[45]
	The project teams were comprised of motivated people with a commitment to project success.	[45]
	The project team members were, in general, always willing to learn and train each other through	[45]

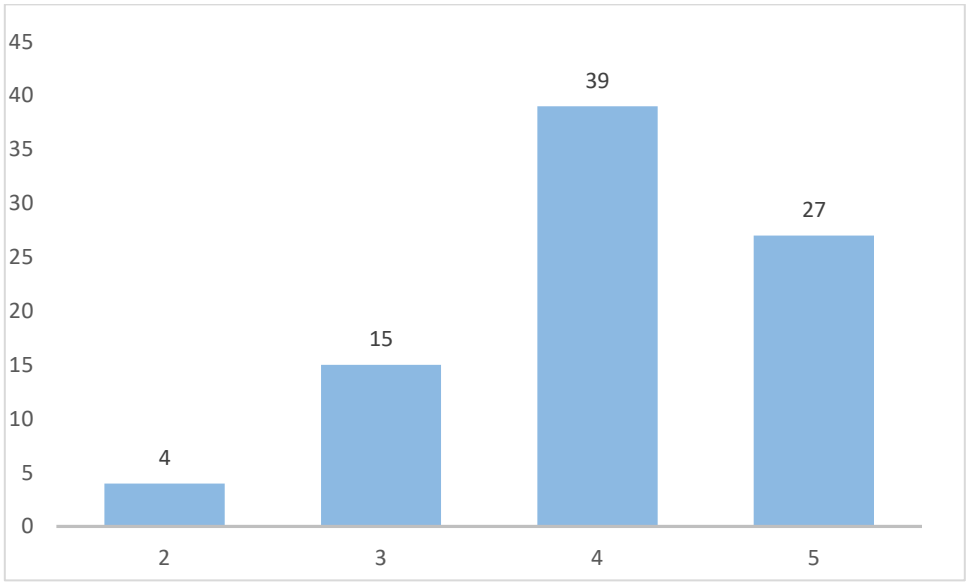
	mentoring and professionally guided discussions as well as formal training.	
Organizational culture	The company's organizational culture encouraged communication and assertiveness between teams.	[48]
	Count of The company's organizational culture supported an agile approach during project development.	[48]
Management commitment	The project team facilitators/coordinators had a clear vision of the scope and reach of the project.	[49]
	The project team facilitators/coordinators were agile knowledgeable managers with an adaptive management style.	[50]
	Top management dedicated time towards making the required technology and cultural changes for agile implementation.	[50]
Customer involvement	The project had strong customer commitment and presence (i.e. having multiple representatives on-site working hard and full-time as a member of the project team).	[51]

	There was a good customer relationship within the project.	[51]
	The project was customer centred (i.e. an exceptional amount of time and effort was put towards the understanding of the customer needs).	[52]
Inter-team coordination	The project implemented appropriate coordination mechanisms that catered towards inter-team dynamics.	[53]
	All teams participating in the project were aware of the overall project status and deadlines.	[54]
IT agile development project success	Costs: The project was delivered under or within budget.	[22]
	Time: the project respected initial deadlines and was delivered on time.	[22]
	Customer satisfaction: The product's performance managed to overcome the end-users' expectations.	[22]
	Internal Stakeholder contentment: Your participation on the project was overall a positive experience.	[22]

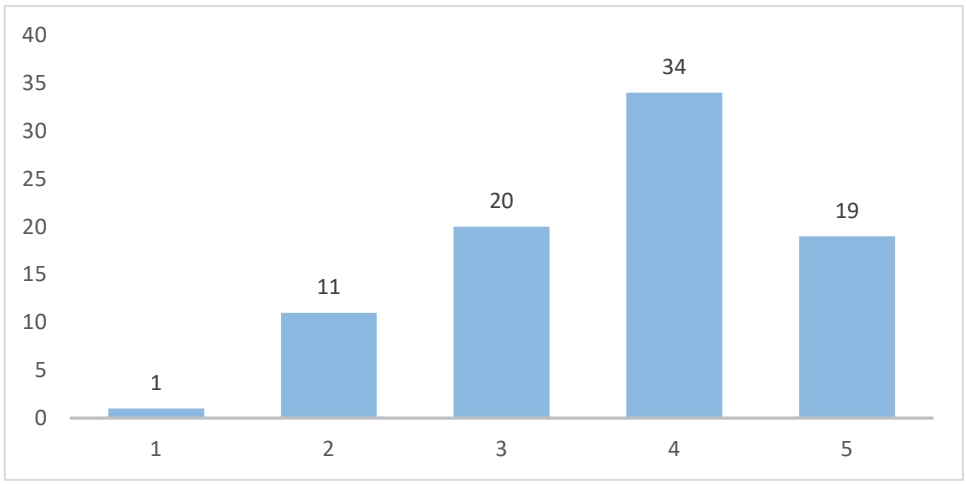
Section 2: survey results

Personal characteristics

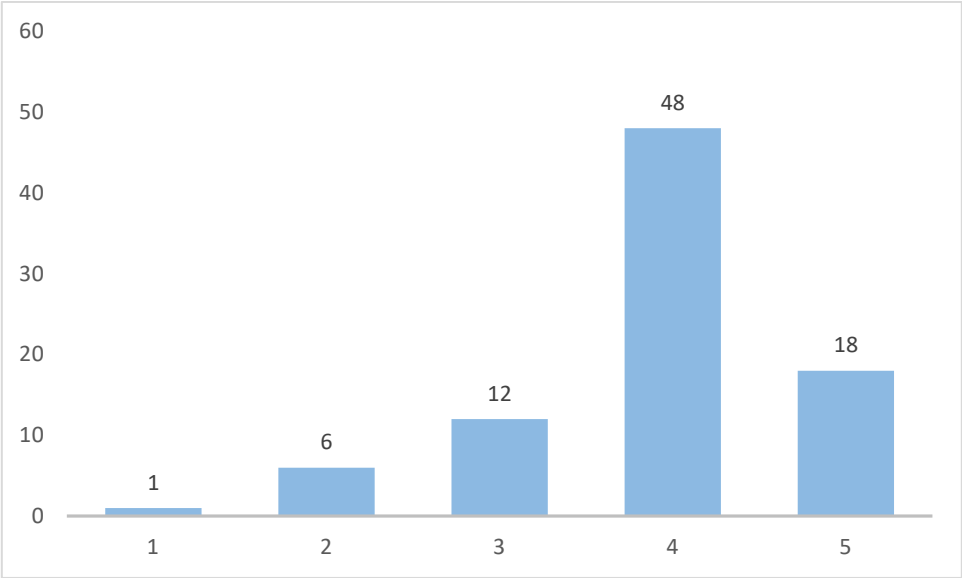
a) Count of “the project teams consisted of people with strong interpersonal and communication skills”.



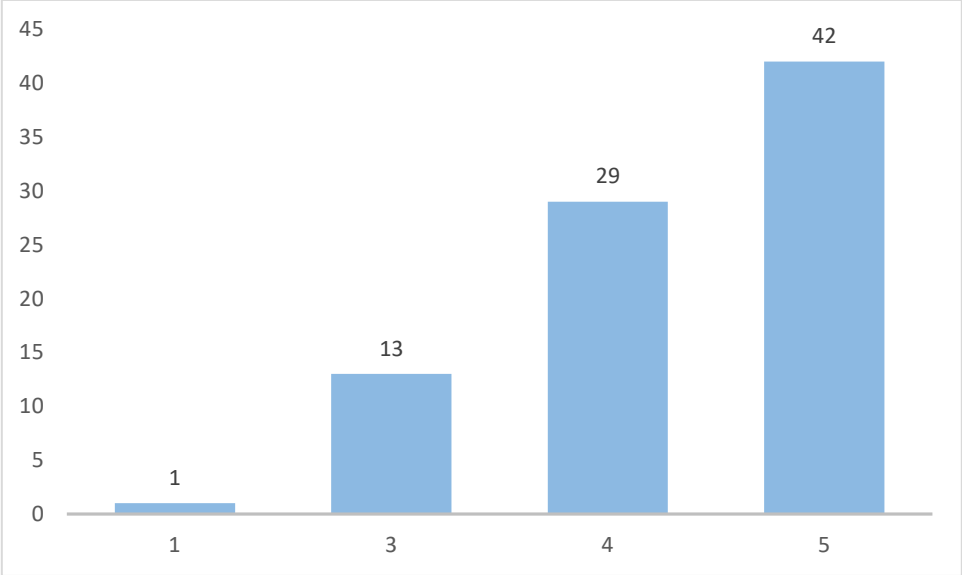
b) The project teams consisted of people with the ability to handle ambiguity (i.e. can effectively cope with change, shift gears comfortably, decide and act without having the total picture, and handle risk and uncertainty.)



c) The project teams consisted of people with strong interpersonal and communication skills.

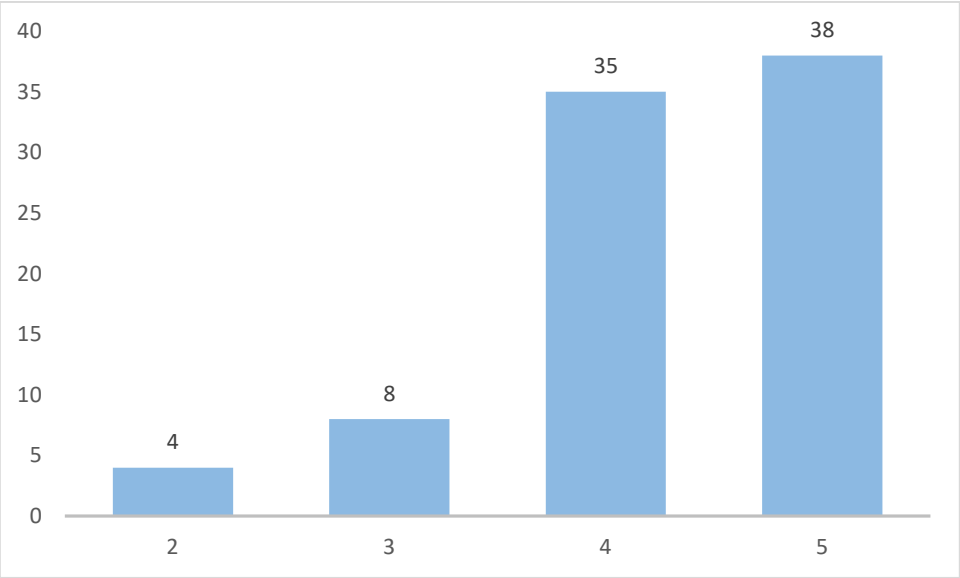


d) Count of The project teams consisted of people who were agreeable and honest.

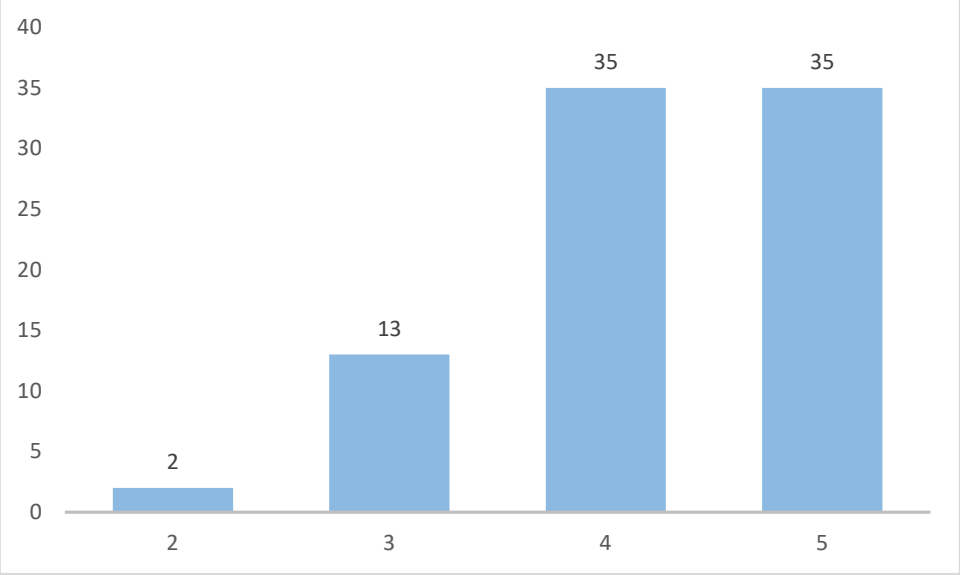


Team capability

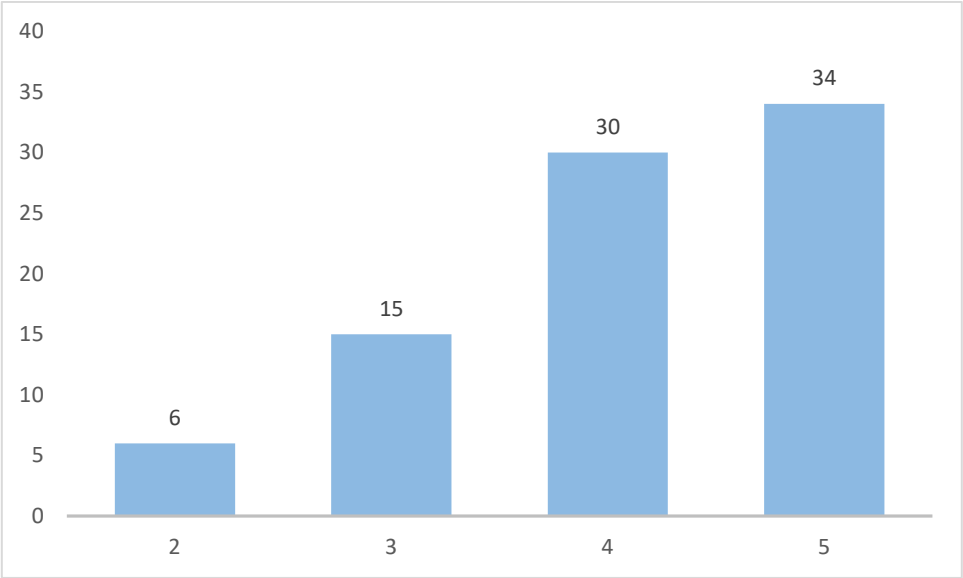
a) Count of The project teams were comprised of technically skilled and competent people (i.e. programming, design, subject matter,).



b) Count of The project teams were comprised of motivated people with a commitment towards project success.

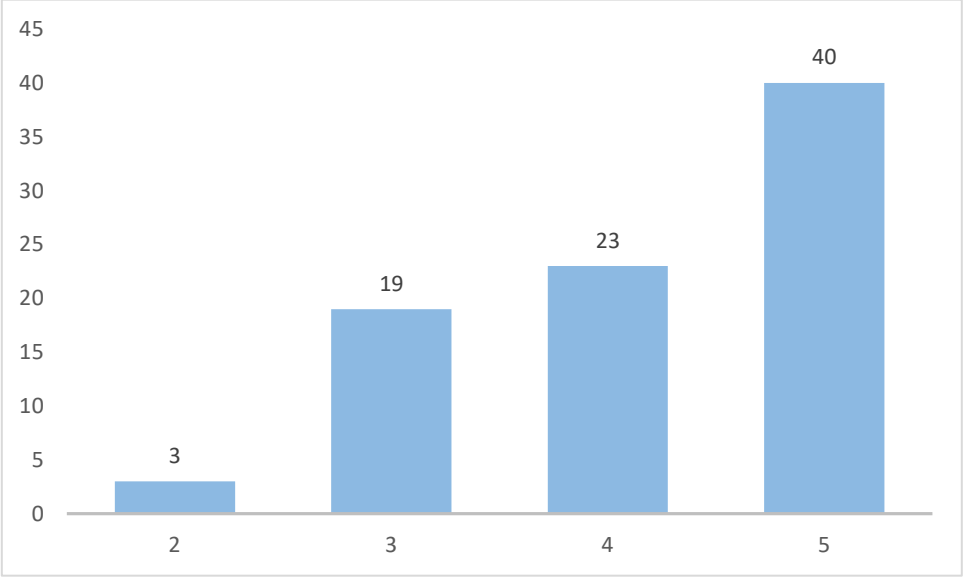


c) Count of The project team members were, in general, always willing to learn and train each other through mentoring and professionally guided discussions.

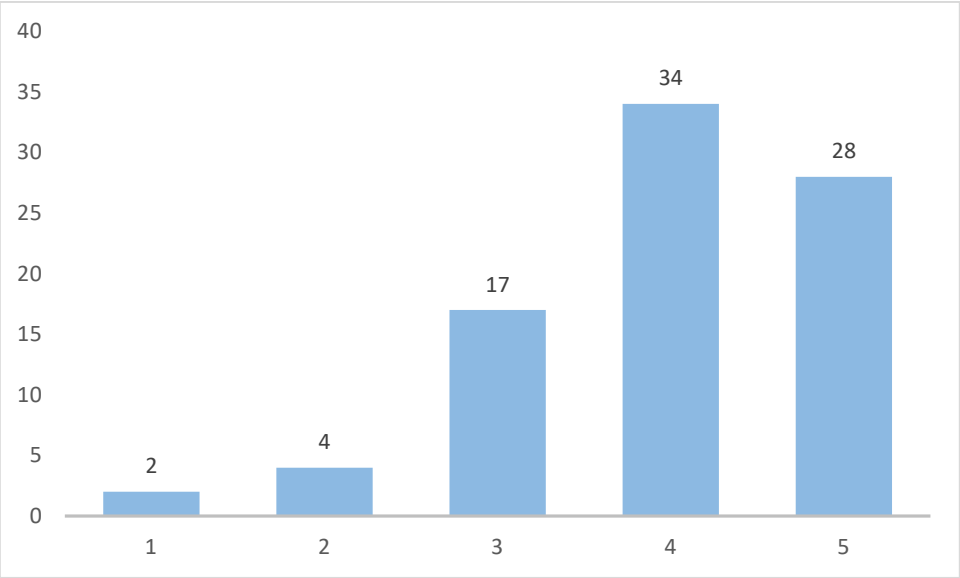


Organizational culture

a) Count of The company's organizational culture encouraged communication and assertiveness between teams.

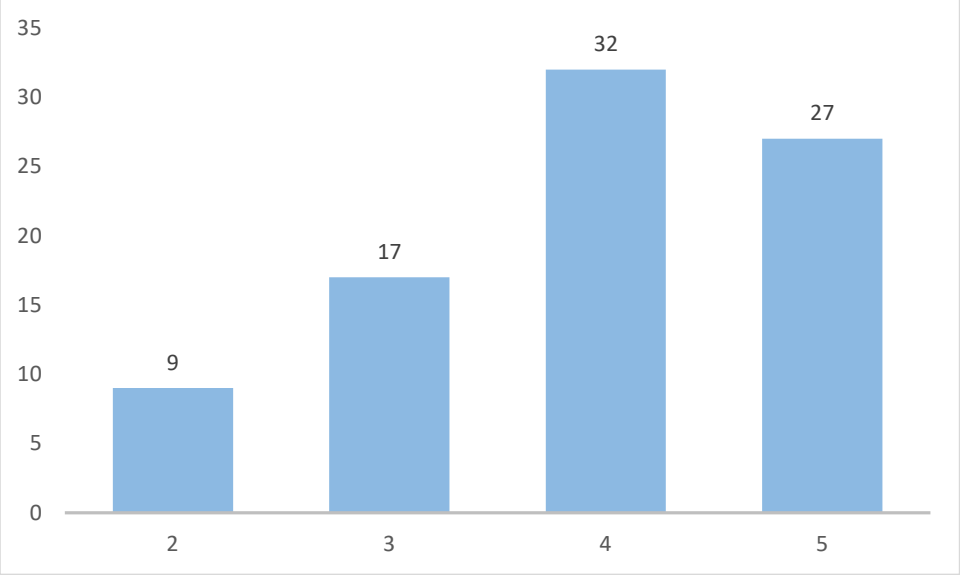


b) Count of The company's organizational culture supported an agile approach during project development.

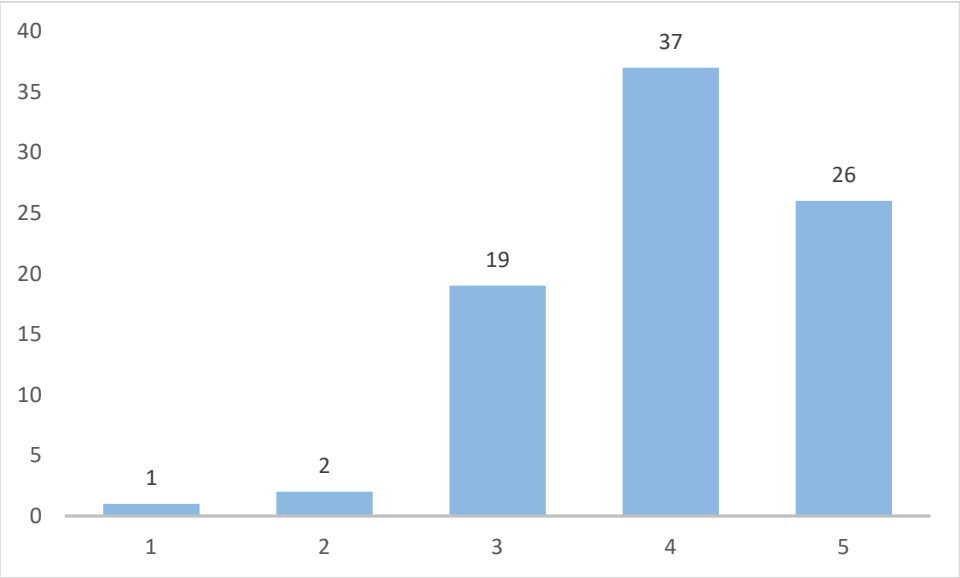


Management commitment

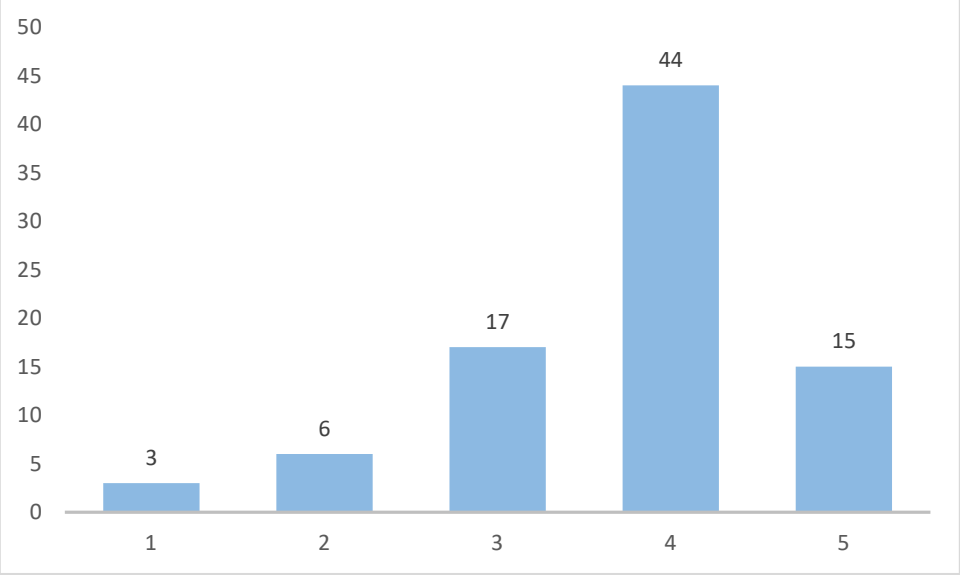
a) Count of The project team facilitators/coordinators had a clear vision of the scope and goal of the project.



b) Count of The project team facilitators/coordinators were agile knowledgeable managers with an adaptive management style.

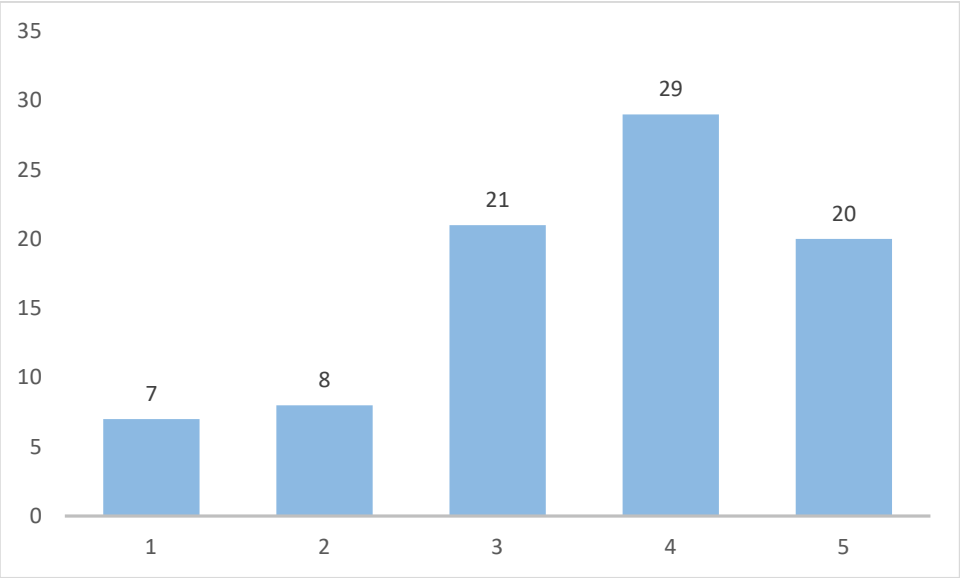


c) Count of Top management dedicated time towards making the required technology and cultural changes for agile implementation.

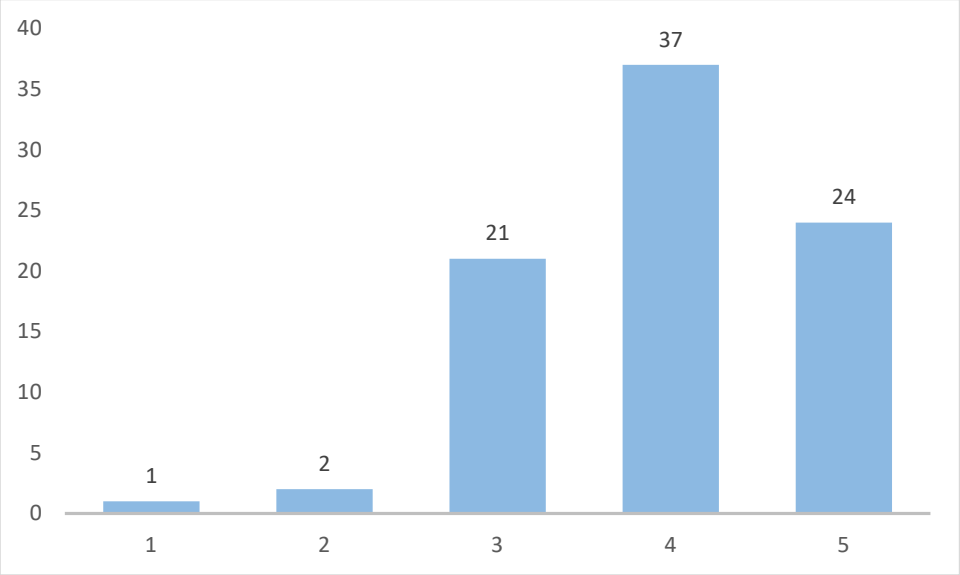


Customer involvement

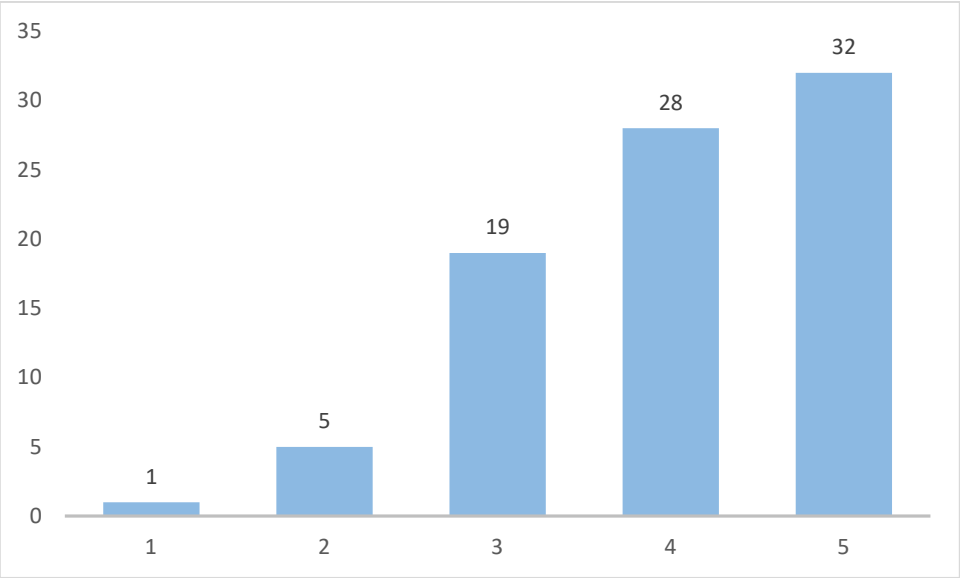
a) Count of The project had strong customer commitment and presence (i.e. having representatives on site working hard and full-time as a member of the project team).



b) Count of There was a good customer relationship within the project.

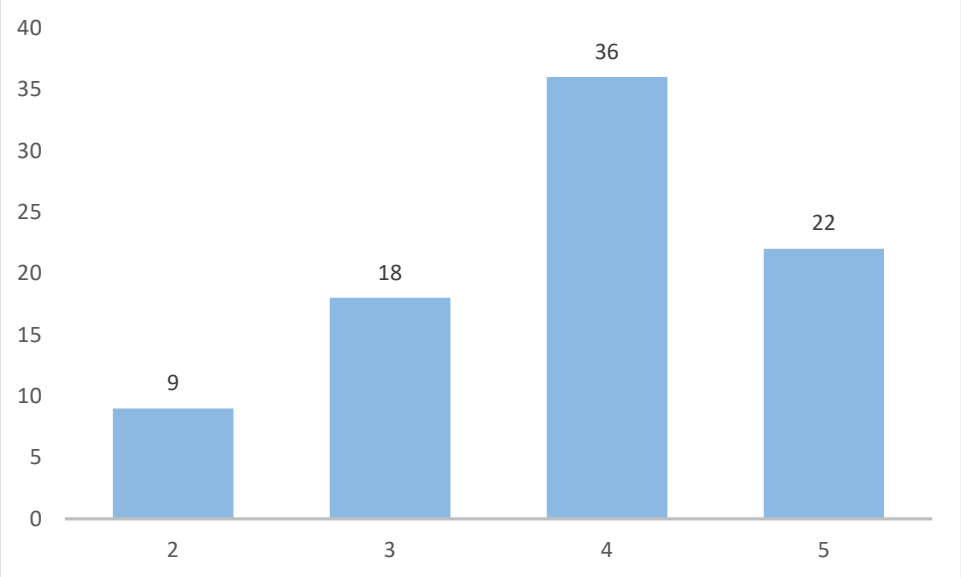


c) Count of The project was customer centred (i.e. an exceptional amount of time and effort was put towards the understanding of the customer needs)

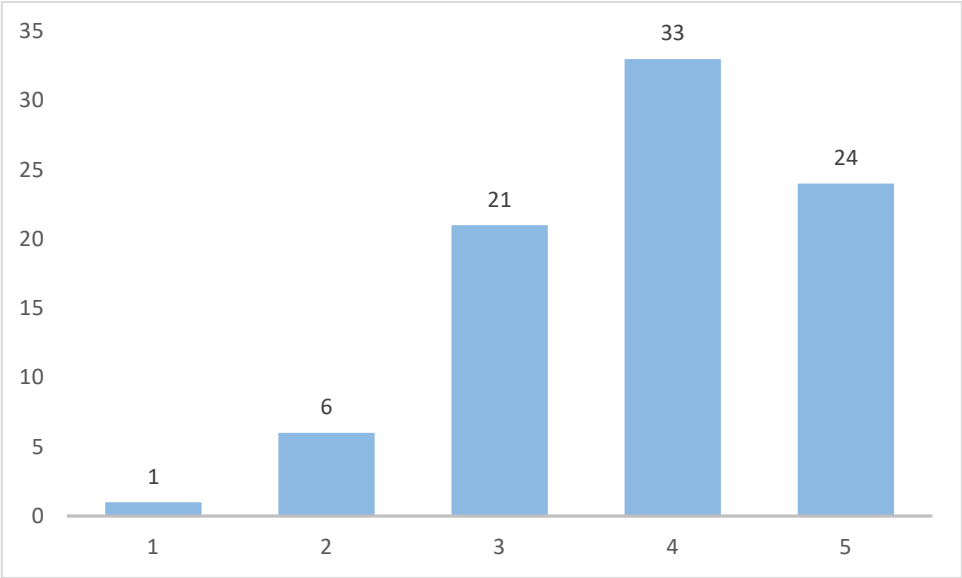


Inter-team coordination

a) Count of The project implemented appropriate coordination mechanisms that catered towards multi-team systems.

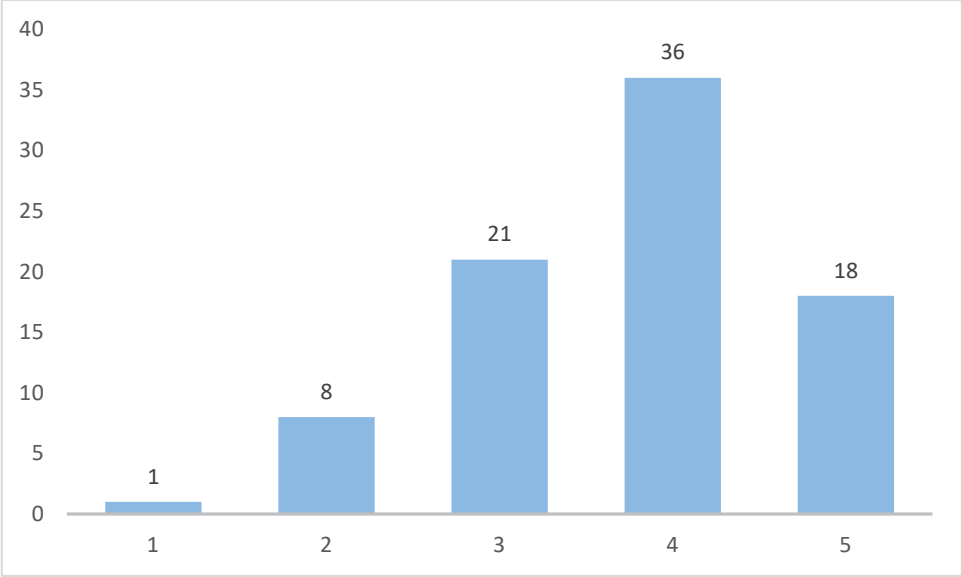


b) Count of All teams participating in the project were aware of the overall project status at all times.

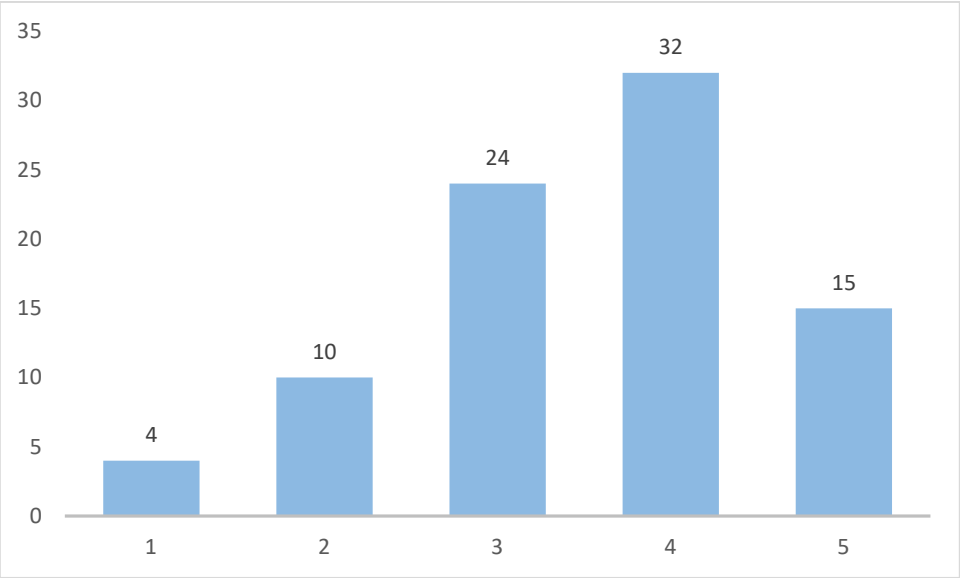


Project Success

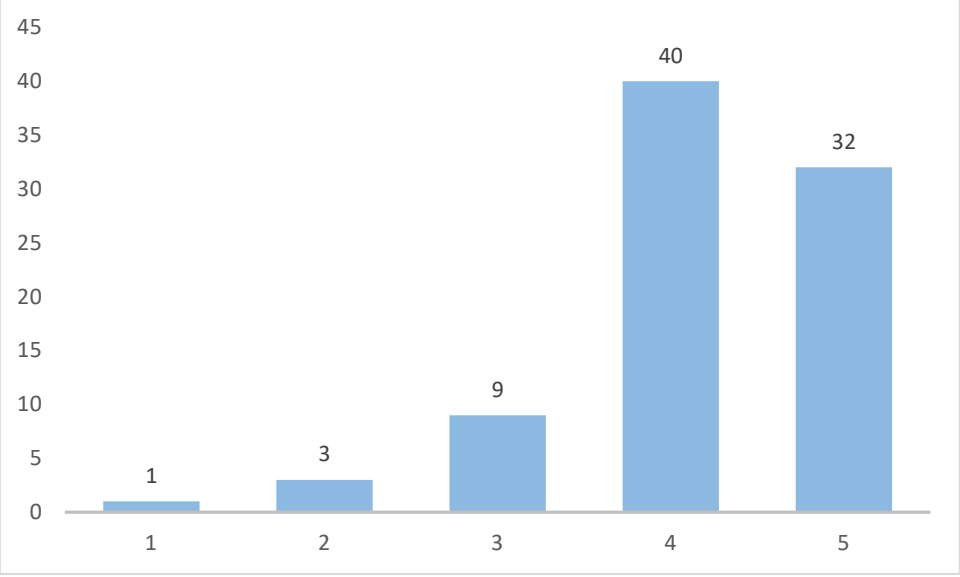
a) Count of Costs: The project was delivered under or within budget.



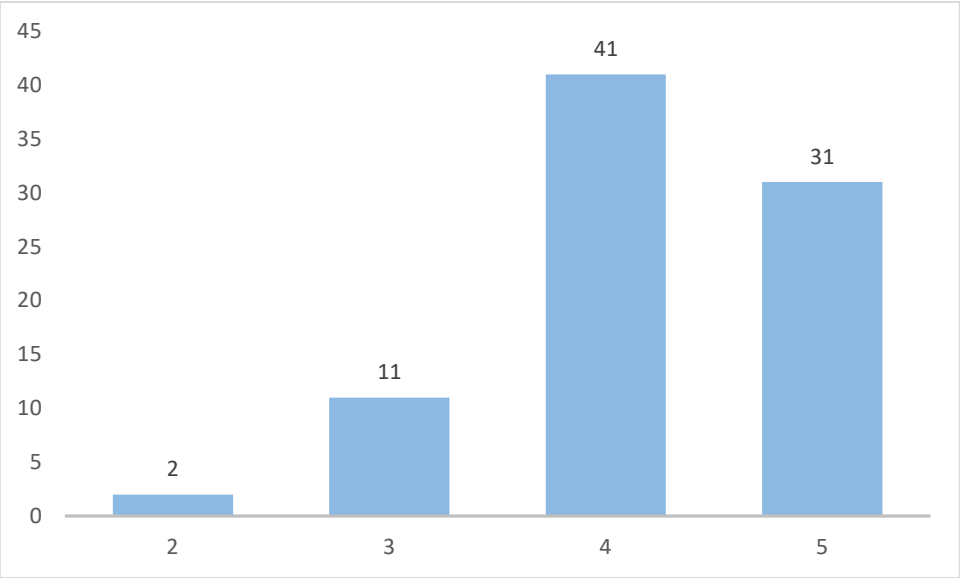
b) Count of Time: the project respected initial deadlines and was delivered on time.



c) Count of Customer satisfaction: The product's performance managed to overcome the end-users' expectations.



d) Count of Internal Stakeholder contentment: Your participation on the project was overall a positive experience.



References

[1] N. Ozkan and A. K. Tarhan, “Investigating Causes of Scalability Challenges in Agile Software Development from a Design Perspective,” in *2019 1st International Informatics and Software Engineering Conference (UBMYK)*, Ankara, Turkey, 2019.

[2] N. Ozkan and A. K. Tarhan, “Evaluation of Scrum-based Agile Scaling Models for Causes of,” in *ENASE 2020 - 15th International Conference on Evaluation of Novel Approaches to Software Engineering*, Ankara, Turkey, 2020.

[3] A. Putta, “Scaling Agile Software Development to Large and Globally,” in *2018 ACM/IEEE 13th International Conference on Global Software Engineering*, Gothenburg, Sweden, 2018.

[4] T. Gustavsson, “Dynamics of inter-team coordination routines in large-scale agile software development,” in *27th European Conference on Information Systems - Information Systems for a Sharing Society, ECIS 2019*, Karlstad, Sweden, 2019.

[5] M. F. Abrar, M. Sohail, S. Ali, M. F. Majeed, I. A. Shah, N. Rashid and N. Ullah, “De-motivators for the adoption of agile methodologies for large-scale software development teams: An SLR from management perspective,” *Journal of Software: Evolution and Process,* 24 June 2020.

[6] N. B. Moe and T. Dingsøyr, “Emerging Research Themes and updated Research Agenda for Large-Scale Agile Development:,” in *the XP2017 Scientific Workshops*, Cologne, Germany, 2017.

[7] T. Kovaleva, “Assessment of Success Measures and Challenges of the Agile Project Management,” in *International Conference Cyber-Physical Systems and Control*, St. Petersburg, Russia, 2019.

[8] “Association for Project Management,” 2020. [Online]. Available: https://www.apm.org.uk/resources/find-a-resource/agile-project-management/. [Accessed 1 October 2020].

[9] “Wrike,” 2020. [Online]. Available: https://www.wrike.com/project-management-guide/faq/what-is-agile-methodology-in-project-management/. [Accessed 1 October 2020].

[10] M. Alexander, “Agile project management: 12 key principles, 4 big hurdles,” *CIO,* 2018.

[11] “Productplan.com,” 2020. [Online]. Available: https://www.productplan.com/glossary/agile-framework/. [Accessed 1 October 2020].

[12] “Scrum.org,” 2017. [Online]. Available: https://www.scrum.org/resources/what-is-scrum. [Accessed 2 October 2020].

[13] “Mendix Tech BV 2020,” 2020. [Online]. Available: https://www.mendix.com/agile-framework/. [Accessed 2 October 2020].

[14] “AgileAlliance.org,” 2020. [Online]. Available: https://www.agilealliance.org/glossary/kanban/#q=~(infinite~false~filters~(postType~(~'page~'post~'aa_book~'aa_event_session~'aa_experience_report~'aa_glossary~'aa_research_paper~'aa_video)~tags~(~'kanban))~searchTerm~'~sort~false~sortDirection~'asc~page~. [Accessed 3 October 2020].

[15] “Blueprint.com,” 2020. [Online]. Available: https://www.blueprintsys.com/agile-development-101/agile-methodologies. [Accessed 3 October 2020].

[16] “Airfocus.com,” 2020. [Online]. Available: https://airfocus.com/glossary/what-is-the-crystal-agile-framework/. [Accessed 3 October 2020].

[17] J. Stapleton, “DSDM: Dynamic Systems Development Method,” in *Proceedings Technology of Object-Oriented Languages and Systems. TOOLS 29 (Cat. No.PR00275)*, Nancy, France, 1999.

[18] S. W. Ambler, “AgileModeling.com,” 2005. [Online]. Available: http://agilemodeling.com/essays/fdd.htm. [Accessed 3 October 2020].

[19] A. Aldahmash, A. M. Gravell and Y. Howard, “A Review on the Critical Success Factorsof Agile Software Development,” in *European Conference on Software Process Improvement*, Southampton, UK, 2017.

[20] A. Srivastava, D. Mehrotra, P. K. Kapur and A. G. Aggarwal, “Analytical evaluation of agile success factors influencing quality in software industry,” *International Journal of System Assurance Engineering and Management,* pp. 247-257, 07 March 2020.

[21] J. Totten, *Critical Success Factors for Agile Project Management in NonSoftwar Software Related Product Development Teams,* Michigan, USA: Western Michigan University, Dessertations, 2017.

[22] C. Tam, E. J. d. C. Moura, T. Oliveira and J. Varajão, “The factors influencing the success of on-going agile software development projects,” *International Journal of Project Management,* pp. 165-176, April 2020.

[23] C. Perera and I. Perera, “The Impact of Client Involvement towards Agile Project Success in Sri Lankan Software Industry,” in *2019 Moratuwa Engineering Research Conference (MERCon)*, Moratuwa, Sri Lanka, 2019.

[24] “Atlassian.com,” 2020. [Online]. Available: https://www.atlassian.com/agile/agile-at-scale. [Accessed 5 October 2020].

[25] K. Dikert, M. Paasivaara and C. Lassenius, “Challenges and success factors for large-scale agile transformations: A systematic literature review,” *Journal of Systems and Software,* pp. 87-108, September 2016.

[26] T. Dingsøyr and N. B. Moe, “Towards Principles of Large-Scale Agile Development: A Summary of the workshop at XP2014 and a revised research agenda,” in *XP2014 Workshop on Principles of Large-Scale Agile Development*, Rome, Italy, 2014.

[27] H. Berger and P. Beynon-Davies, “The utility of rapid application development in large-scale, complex projects,” *Information Systems Journal,* pp. 549-570, November 2009.

[28] A. Henriksen and S. Pedersen, “A qualitative case study on agile practices and project success in agile software projects,” *Journal of Modern Project Management,* pp. 63-73, 1 January 2017.

[29] E. Bjarnason, K. Wnuk and B. Regnell, “A case study on benefits and side-effects of agile practices in large-scale requirements engineering,” in *ECOOP '11: European Conference on Object-Oriented Programming*, Lancaster United Kingdom, 2011.

[30] T. Dingsøyr, T. Fægri and J. Itkonen, “What is large in large-scale? a taxonomy of scale for agile software development,” in *International Conference on Product-Focused Software Process Improvement*, Helsinki, Finland, 2014.

[31] S. V. Zykov y A. Singh, Agile Enterprise Engineering: Smart Application of Human Factors: Models, Methods, Practices, Case Studies, Moscow, Russia: Springer, 2020.

[32] M. Fowler, *Put your process on a diet,* Amsterdam, Holland: Elsevier Science B.V., 2000.

[33] J. Karlsson, “Thenewstack.com,” 9 January 2019. [Online]. Available: https://thenewstack.io/principles-of-good-large-scale-agile/#:~:text=The%20usual%20definition%20of%20large,enterprise%2Dwide%20across%20different%20departments.. [Accessed 5 October 2020].

[34] K. Conboy and N. Carroll, *Implementing Large-Scale Agile Frameworks:,* Galway, Ireland: National University of Ireland Galway, 2019.

[35] D. K. Rigby, J. Sutherland and A. Noble, “Agile at Scale,” *Harvard Business Review,* vol. May–June 2018 Issue, 2018.

[36] “LeSS.works,” The LeSS Company B.V., 2020. [Online]. Available: https://less.works/less/framework/index. [Accessed 5 October 2020].

[37] “Atlassian.com,” 2020. [Online]. Available: https://www.atlassian.com/agile/agile-at-scale. [Accessed 5 October 2020].

[38] “Scaledagileframework.com,” Scaled Agile, Inc., 2020. [Online]. Available: https://www.scaledagileframework.com/about/. [Accessed 5 October 2020].

[39] H. Portman, “Thedigitalprojectmanager.com,” 10 December 2019. [Online]. Available: https://thedigitalprojectmanager.com/agile-methodologies/. [Accessed 5 October 2020].

[40] K. Schwaber, “Scrum.org,” January 2018. [Online]. Available: https://www.scrum.org/resources/online-nexus-guide. [Accessed 5 October 2020].

[41] E. Shipper, L. Mazer, S. Merrell, D. Lin, J. Lau and M. Melcher, “Pilot evaluation of the Computer-Based Assessment for Sampling Personal Characteristics test,” *Journal of Surgical Research, 215,* pp. 211-218, 2017.

[42] W. Aghina, C. Handscomb, J. Ludolph, D. West and A. Yip, “mckinsey.com,” January 2019. [Online]. Available: https://www.mckinsey.com/~/media/McKinsey/Business%20Functions/Organization/Our%20Insights/How%20to%20select%20and%20develop%20individuals%20for%20successful%20agile%20teams%20A%20practical%20guide/How-to-select-and-develop-individuals-for-successful-agil. [Accessed 20 October 2020].

[43] International Project Management Association, *Individual competence baseline for project, programme & portfolio management,* Zurich, 2015.

[44] M. Haas, “Knowledge Gathering, Team Capabilities, and Project Performance in Challenging Work Environments,” *Management Science,* vol. 52, no. 8, pp. 1170-1184, 2006.

[45] T. Chow and D. Cao, “A survey study of critical success factors in agile software projects,” *Journal of Systems and Software,* pp. 961-971, 2008.

[46] P. Wendorff, “Organisational Culture in Agile Software Development,” in *Product Focused Software Process Improvement, 4th International Conference, PROFES 2002*, Rovaniemi, Finland, 2002.

[47] “SHRM.org,” Society for Human Resource Management (SHRM), 2017. [Online]. Available: https://www.shrm.org/resourcesandtools/tools-and-samples/toolkits/pages/understandinganddevelopingorganizationalculture.aspx. [Accessed 20 October 2020].

[48] D. E. Strode, S. L. Huff and A. Tretiakov, “The Impact of Organizational Culture on Agile Method Use,” in *International Conference on System Sciences*, Hawaii, 2009.

[49] P. ZHU, *The Top Management Role in Agile Journey,* 2015.

[50] S. Comella-Dorda, K. Krishnakanthan, J. Maurone and a. G. Shenai, “A business leader’s guide to agile,” *McKinsey Digital,* 2017.

[51] D. Rooney, *Practical Agility,* 2006.

[52] BjørnarTessem, “The customer effect in agile system development projects. A process tracing case study,” *Procedia Computer Science,* pp. 244-251, 14 December 2017.

[53] T. Dingsøyr, N. B. Moe and E. A. Seim, “Coordinating Knowledge Work in Multi-Team Programs: Findings from a Large-Scale Agile Development Program,” *Project Management Journal,* pp. 64-77, 2018.

[54] F. O. Bjørnson, J. Wijnmaalen, C. J. Stettina and D. Torgeir, Inter-team Coordination in Large-Scale Agile Development: A Case Study of Three Enabling Mechanisms, Trondheim, Norway, 2018.

[55] J. Harry N. Boone and D. A. Boone, “Analyzing Likert Data,” *The Journal of Extension (JOE),* vol. 50, no. 2, 2012.

[56] G. M. Sullivan y J. Anthony R. Artino, «Analyzing and Interpreting Data From Likert-Type Scales,» *Journal of Graduate Medical Education,* vol. 5, nº 4, p. 541–542, 2013.

[57] “UCLA: Statistical Consulting Group,” UCLA, [Online]. Available: https://stats.idre.ucla.edu/other/mult-pkg/faq/general/faq-how-do-i-cite-web-pages-and-programs-from-the-ucla-statistical-consulting-group/. [Accessed 18 01 2021].

[58] M. Tavakol and R. Dennick, “Making sense of Cronbach's alpha,” *International Journal of Medical Education,* vol. 2, p. 53–55, 2011.

[59] K. S. Taber, “The Use of Cronbach’s Alpha When Developing,” University of Cambridge, Cambridge, 2017.

[60] R. Tapsir, N. A. N. Pa and S. N. A. B. S. Zamri, “Reliability and Validity of the Instrument Measuring Values in Mathematics Classrooms,” *Malaysian Online Journal of Educational Sciences,* vol. 6, no. 2, pp. 37-47, 2018.

[61] S. Glen, “Average Inter-Item Correlation: Definition, Example,” StatisticsHowTo.com: Elementary Statistics for the rest of us!, 16 June 2018. [Online]. Available: https://www.statisticshowto.com/average-inter-item-correlation/#:~:text=The%20ideal%20range%20of%20average,well%20(if%20at%20all).. [Accessed 18 06 2021].

[62] R. L. Piedmont, “Inter-item Correlations,” in *Michalos A.C. (eds) Encyclopedia of Quality of Life and Well-Being Research*, Springer, Dordrecht, 2014, pp. 142-143.

[63] “SPC for Excel,” BPI Consulting, LLC., [Online]. Available: https://www.spcforexcel.com/help/statistical-tools/miscellaneous/item-analysis. [Accessed 18 01 2021].

[64] E. A. O. Zijlmans, J. Tijmstra, L. A. v. d. Ark and a. K. Sijtsma, “Item-Score Reliability in Empirical-Data Sets and Its Relationship With Other Item Indices,” *Educational and Psychological Measurement (EPM) ,* vol. 78, no. 6, p. 998–1020, 2018.

[65] E. Cristobal, C. Flavian and M. Guinalíu, “Perceived e-service quality (PeSQ): measurement validation and effects on consumer satisfaction and web site loyalty,” *Journal of Service Theory and Practice,* vol. 17, no. 3, 2007.

[66] “Laerd Statistics,” 2018 Lund Research Ltd, [Online]. Available: https://statistics.laerd.com/spss-tutorials/kruskal-wallis-h-test-using-spss-statistics.php#:~:text=The%20Kruskal%2DWallis%20H%20test,continuous%20or%20ordinal%20dependent%20variable.. [Accessed 20 January 2021].

[67] A. Ali, A. Rasheed, A. A. Siddiqui, M. Naseer, S. Wasim and W. Akhtar, “Non-Parametric Test for Ordered Medians: The Jonckheere Terpstra Test,” *International Journal of Statistics in Medical Research,* vol. 4, no. 2, pp. 203-207, 2015.

[68] S. TOUPIN, “The Jonckheere-Terpstra test for ordered differences,” Edinburgh, 2017.

[69] “Interpret the key results for Kruskal-Wallis Test,” Minitab, LLC, 2019.

[70] A. C.Leon, “3.12 - Descriptive and Inferential Statistics,” in *Reference Module in Neuroscience and Biobehavioral Psychology*, Elsevier Science Ltd., 1998, pp. 243-285.

[71] R. G. v. d. Berg, “SPSS Ttutorials,” [Online]. Available: https://www.spss-tutorials.com/spss-kruskal-wallis-test-simple-tutorial-with-example/. [Accessed 20 January 2021].

[72] “Laerd Statistics,” Lund Research Ltd, 2018. [Online]. Available: https://statistics.laerd.com/spss-tutorials/jonckheere-terpstra-test-using-spss-statistics.php. [Accessed 20 January 2021].

[73] M. Haida, S. Graschitz and P. Heimerl, “A matter of Motivation - The Effects of Risk Preference and Task Complexity on the Auditor´s Motivation,” *WSB Journal of Business and Finance,* vol. 53, no. 2, 2019.

[74] B. Keita, “Success Stories That Will Make You Believe in Scaled Agile,” Invensis Inc., 2020.

[75] R. Cross, H. K. Gardner and A. Crocker, “For an Agile Transformation, Choose the Right People,” Harvard Business Review, 2021.

[76] S. Panditi, “The Key to Agile Success? Focus on Outcomes, not Metrics,” Harvard Business Publishing, 2018.