

PROBLEMS IN MULTI-VENDOR GLOBALLY DISTRIBUTED AGILE SCRUM PROJECTS AND IMPACT ON PROJECT PERFORMANCE – EXPLORATIVE STUDY

MSc Engineering Project Management - Dissertation Report

FEBRUARY 9, 2021

Prabhakaran Chinnaya Karuppannan
Student ID: 2019153



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

Abstract

Recent trends towards globalisation and a need for flexible and cost-effective project management method has led organisations to adopt outsourcing and agile project management method. Although there are several researches performed on success and failure factors in agile projects, there are no significant research available on problems faced by agile scrum project that are outsourced to multiple vendors and the project team is globally distributed. This motivated us to perform research on this area to identify problems or failure modes in multi-vendor globally distributed projects that can impact project performance.

Acknowledgement

I wish to record my deep sense of gratitude and profound thanks to my research guide Mr. Andrew Thorn, Program Manager, MSc Engineering Project Management, University of Wales Trinity Saint David for their keen interest, inspiring guidance, and constant encouragement with my work during all stages, to bring this research into fruition.

I would also like to take this opportunity to thank all my friends and parents Dr.C. Karuppannan & K. Rajamani, siblings K.Saranya & K.Sukanya for their support with my research and studies towards achieving a Master's degree.

Declaration Statement

This dissertation is the result of my original research. It has been composed by me and has not been previously submitted for examination which has led to the award of a degree.

Signed: Prabhakaran Chinnaya Karuppanan

Date: 09/02/2022

A handwritten signature in black ink, appearing to read 'Prabhakaran Chinnaya Karuppanan', with a stylized, cursive script.

Contents

Abstract.....	2
Acknowledgement.....	3
Declaration Statement.....	4
Introduction	7
Research Aim:.....	9
Research Question.....	9
Research Objectives.....	9
Research Overview	10
Performance Objectives:	10
Critical Success Factors:.....	11
Agile Scrum Method:.....	13
RESEARCH METHODOLOGY AND RESEARCH METHODS	16
Research Methodology	16
Research Design/Activities involved in the Research.....	17
Literature Review	18
Unclear Business Goals.....	19
Lack of clear boundaries defining scope and tasks	19
Poor Project Integration.....	19
Lack of Realistic Schedules	21
Lack of Effective Communication	22
Interruptions in Agile Projects	24
Conflicts and Lack of Trust.....	25
Constantly Changing Resources.....	26
Survey of Agile Scrum Practitioners in Software Development	30
Testing Draft Version of Survey Questionnaire.....	31

Survey Analysis	32
Interview Agile Scrum Practitioners working in Multi-Vendor Distributed Projects...	34
Research Analysis and Findings.....	35
P20:	35
Lack of different skillsets, appropriate tools or restricted access to tools hampers the team to be fully cross-functional and makes the team heavily reliant on component teams in possession of necessary skills or tools.....	35
P10:	39
Frequent changes in requirements or user story especially towards the end of a sprint causes excessive delays, re-planning and rework	39
P16:	42
Constantly changing resources or adding new resources in the project affects team's chemistry and velocity	42
P23:	45
Lack of project management competence or appropriate leadership style for agile projects by project manager or scrum master, i.e., project manager adopts an autocratic leadership characterized by individual control over all decisions and little input from group members.....	45
P19:	49
Lack of continuous Client/Business User involvement due to distributed nature of the project, such as communicating to business and getting feedback on matters	49
P18:	52
Inability to breakdown backlog into smaller user stories and prioritise based on most business value that can be delivered within a sprint 'or' horizontal breakdown of stories based on tasks such as design, coding, testing, etc., instead of product features often leads to work spanning over multiple sprints and not delivering working software or business value at the end of each sprint.....	52
P2:	54

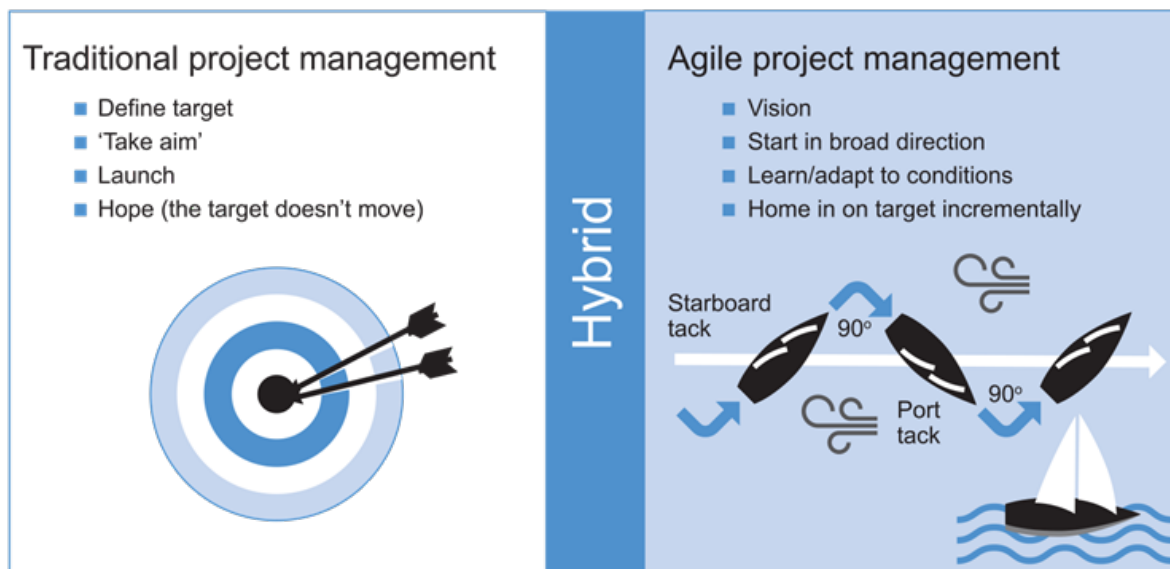
Problems such as changing priorities, changing scope and conflicts with resource availability by teams within the project and interfacing teams outside project prevents tasks from being completed within sprint timelines	54
P11:	57
People often allocated to multiple projects/teams on shared basis results in discontent and stress personally and interruption to team members seeking input on overlapping urgent issues	57
Conclusion	59
Limitations.....	60
Recommendations for Future Work.....	61
References.....	61
Appendix A	66
Appendix B	96

Introduction

Agile project management has gained popularity in software development projects over the last few years due to its flexible nature and ability to solve problems with delivering high quality software under strict time constraints and constantly changing business requirements. Studies show that at least 70% of the organisations world-wide adopt agile methods in managing the projects (Wysocki, 2014). So, what is an agile project management and why is it preferred over the traditional project management methods? Projects using a traditional waterfall model requires project requirements to be defined upfront and directs change through a change management practice which is a detailed plan across a long horizon following a linear path from inception to implementation. Traditional waterfall projects follow a linear and structured approach that makes it difficult to change directions or make changes to the project plan later in the project lifecycle (Walsh, et al., 2021). This makes the traditional method less effective for projects with high levels of change, risk, complexity and those that are vulnerable to uncertainty as it is difficult to set out all the requirements upfront or the customer changes the requirements frequently and especially if the end product is not as per the customer's needs, it becomes difficult to move back to the previous phase to make required changes. This causes

the projects to again go through rework of designing, development and testing phases to meet the customer needs incurring further delays and additional cost to the organisation.

These problems led to the development of flexible project management methods and those that can respond to uncertainty and change. This led a team of software developers in search of an alternative method for the documentation driven traditional software development process to develop an “agile manifesto for software development” in 2001 which emphasised on areas of project management in software development that an individual team seeks over standard practices in traditional project management practices such as “individuals and interaction over processes and tools, working software over comprehensive documentation, customer collaboration over contract negotiation, and responding to changes over following a plan” (Beck, et al., 2001). Agile project management is a method focused on collaboration and frequent communication that promotes software development in iterations to identify the existence of problems in initial design (Walsh, et al., 2021) and an iterative approach focused on delivering a minimum viable product, i.e., something of value to the customer as soon as possible and further improvements in successive iterations (Association for Project Management, 2021). Projects using iterative and incremental approaches reduce rework and waste by frequently adapting to change through short feedback loops and constantly re-prioritizing and updating plans for frequent delivery (Project Management Institute and Agile Alliance, 2017). Agile projects employ practices that focus on flexibility, collaboration, efficiency, and simplicity by eliminating non-value-added activities and using a minimal set of rules in software development process in order to be more responsive to changes (Conforto & Amaral, 2016)



Despite various project management methods employed, software development has not been always successful, thus resulting in software projects frequently delayed, abandoned, or fail to achieve planned objectives and deliver anticipated benefits. Research on project success factors conducted by Association for Project Management with over 850 project professionals and thought leaders from across the private and public sector revealed that around 80% of the projects failed to wholly meet

their planned objectives (Association for Project Management, 2015). The main challenge involved is to identify the problems of inefficiency and waste that can be reduced to improve the performance of software development projects.

Research Aim:

Having chosen Agile Project Management as an area of interest for research, existing literature were studied to formulate the research problem. Formulation of general topic into a particular research problem in a meaningful context by examining literatures regarding concepts & theories or empirical literature about studies in the similar field of interest constitutes the first step in research (Kothari, 2004). While there has been numerous research work on agile project management and its benefits, there are a few exploring the problems in multi-vendor globally distributed agile scrum projects that may impact the success of the project.

Besides this concept of research methodology, my experience as an agile scrum practitioner for two years and being involved in organisational change from traditional to agile project team encourages to perform an explorative study of the problems faced in agile project management.

This led to the proposal of this research with an *'aim to investigate and study in more detail the problems that can impact the performance in software development projects following agile scrum project management method and outsourced to multiple vendors that are not co-located'*.

Research Question

RQ1: What are the day-to-day project management problems faced by globally distributed multi-vendor Software development teams following agile scrum project management practices?

RQ2: Of these common problems, are there any significant problems that can potentially impact the success of projects?

RQ3: How do these problems affect the project performance and are there any relativity between these problems that can have a compounded effect on project performance?

These three questions form the prime focus of this research.

Research Objectives

This research was performed to obtain answers to the research questions in previous section. This research is an explorative study to gain a better understanding of the problems faced in globally distributed software development projects that adopt Agile Scrum Project Management practices.

The main objective of the research is to:

-
- Identify the critical success factors that contribute to project performance through literature review.
 - Identify the common problems in agile projects adopting scrum method through literature review.
 - Identify the significant problems out of the problems identified by literature review by surveying agile software development practitioners.
 - Investigate how the main problems identified through survey impacts the project performance by interviewing selected agile scrum project teams.

Research Overview

This section provides an overview of the proposed research scope and the factors considered in the research and the rationale behind it.

Performance Objectives:

Success of a project in terms of the objective has been met can be judged by five performance objectives (Slack, et al., 2013) :

- Quality – *do things right* to avoid mistakes and satisfy customers with error-free products,
- Speed – *do things fast* to minimise the lead time between customer order and delivery,
- Dependability – *do things on time* to meet the promised delivery timelines,
- Flexibility – *be able to change what you do* to adapt to unexpected changes in customer requirements,
- Cost – *do things cost effectively* to give better value to the customer while still allowing a return to the organisation.

Slack further suggests combining speed and dependability into one composite objective – ‘time’ and relates agility to all these five performance objectives especially flexibility and time. Based on this, four performance measurement factors – Quality, Time, Cost and Flexibility will be considered for our research.

We will also consider a fifth performance factor ‘Morale’ in our research as Agile is a practice that values individuals and interaction over process and tools to promote self-discipline, self-management, ethics and team culture. (Rosenau & Githens, 2006) suggests Morale as an important performance measurement factor in their book as it is difficult to claim a project is successfully managed if there are certain project management factors that develops hatred in project participants towards their involvement in the project. Research by (Weakliem & Frenkel, 2006) implies that workers with ‘higher morale are determined and willing to put more effort into tasks even if its disagreeable, leading to higher productivity’, ‘higher morale will direct their effort in accomplishing organisational goals rather

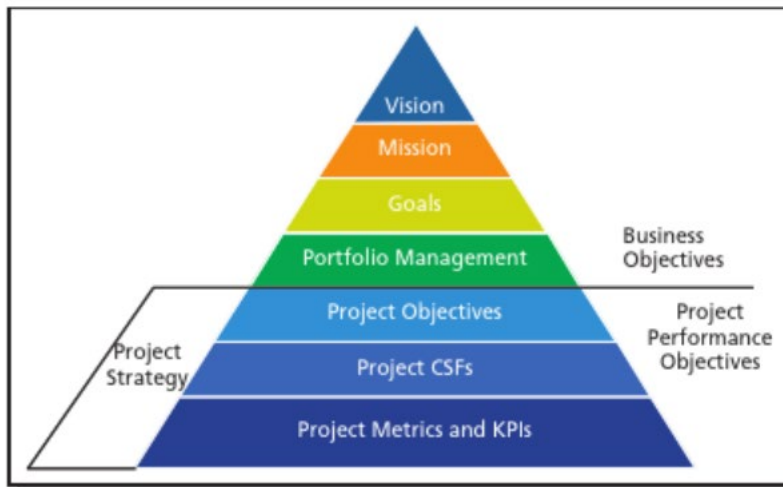
than being self-defensive against criticisms’ and ‘higher morale believes organisation proposed changes adds benefit thereby enabling organisations to adopt effective practices and productivity enhancing changes’.

Performance Objective	Agile Manifesto Principles
Quality	1. Working software is the primary measure of progress. 2. Continuous attention to technical excellence and good design enhances agility.
Time	3. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software. 4. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale. 5. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
Flexibility	6. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage. 7. Business people and developers must work together daily throughout the project. 8. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation. 9. The best architectures, requirements, and designs emerge from self-organizing teams.
Cost	10. Simplicity--the art of maximizing the amount of work not done--is essential.
Morale	11. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done. 12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly.

Critical Success Factors:

High performance results when individuals are aligned to organisational goals through appropriate behaviour, effective use of knowledge, skills, and competencies which means every activity a worker performs at work results in outcomes that adds to achievement of organisational objective. Critical success factors indicate such activities vital to accomplishing organisational goals and are considered as drivers of organisational performance (Armstrong, 2006). Critical success factor method was introduced by (Rockart, 1979) as a means to identify a limited number of business areas that should receive continual and careful management attention to produce satisfactory results in order to ensure organisation goals are attained and successful competitive performance for the organisation. Critical

Success Factors (CSF's) are indicators that represent the critical activities or conditions in a project required to achieve the project objective or meet the desired deliverables of the customer and maintain an effective stakeholder relationship (Kerzner, 2017).



Relationship between Project Performance Objectives and Project Critical Success Factors - adopted from (Kerzner, 2017)

We consider five areas of critical success factors for this research based on the literatures by (Nerur, et al., 2005) on issues faced in migrating to agile methodologies, (Chow & Cao, 2008) critical success factors contributing to the success of agile projects..

Dimensions of Critical Success Factor	Critical Success Factors	Problems
Management and Organisational	Organisation structure Organisation culture Management Commitment Leadership style and collaboration Managing software development knowledge Performance measurement systems	Lack of management commitment Autocratic leadership styles – command and control oriented Organisation culture too political
People	Communication and Collaboration Technical competence and skill set Customer involvement and relationship – collaborative decision making Trust, Respect and Morale amongst employees	Lack of core skill sets to perform development tasks Lack of team work or collaboration by group or individuals Lack of co-operation or involvement by customers
Process	Agile project management competence Team's knowledge of agile methodology adopted Communication channels	Project Managers lacking agile project management knowledge Poor project planning Poor estimation and feature prioritisation techniques
Technical	Well defined design, coding and test standards Documentation methodology Appropriate selection of Technology and Tools for development and tracking agile progress Continuous knowledge upskilling and technical training to the team	Lack of proper agile practices in the team Appropriate tools not available for tracking agile project progress Lack of appropriate tools and technology for development
Project	Project risk analysis Project schedule and cost Project dependencies	Project dependencies and risks not identified or identified later in the project Co-ordination problems with other project teams especially non agile teams

Agile Scrum Method:

Many agile methodologies have been developed to manage software development projects, but the literatures on agile methodologies refer to methods such as Scrum, Extreme Programming, Feature Driven Development, Adaptive Software Development, Crystal, Lean Software Development (Tarwani & Chug, 2016). In this research, we will focus on the Scrum method as it is the most widely used agile approach in software development projects. A survey conducted by digital.ai with more than 4000 participants from software development communities world-wide indicates Scrum as the most popular agile method since 66% of the respondents mentioned agile as the method adopted by their organisation (Digital.ai, 2021).

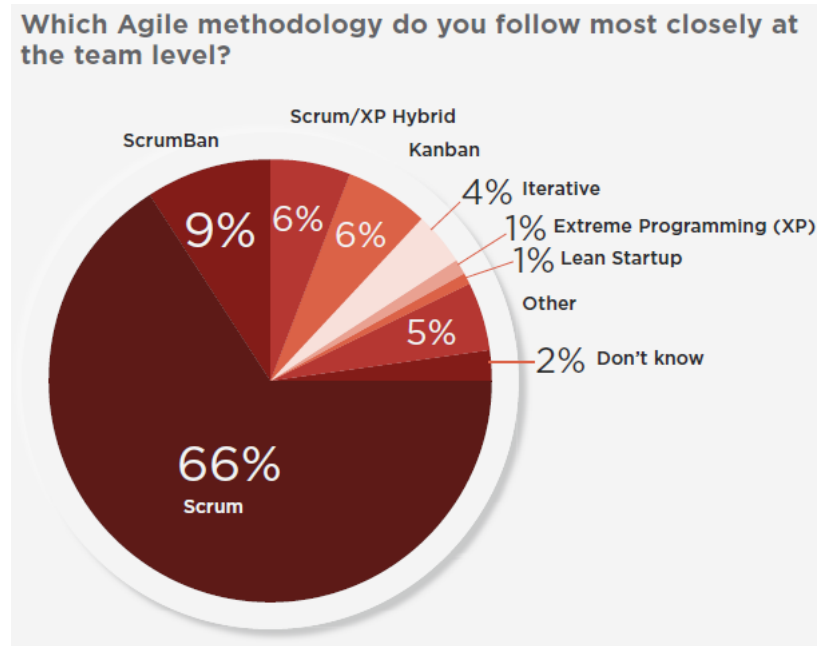
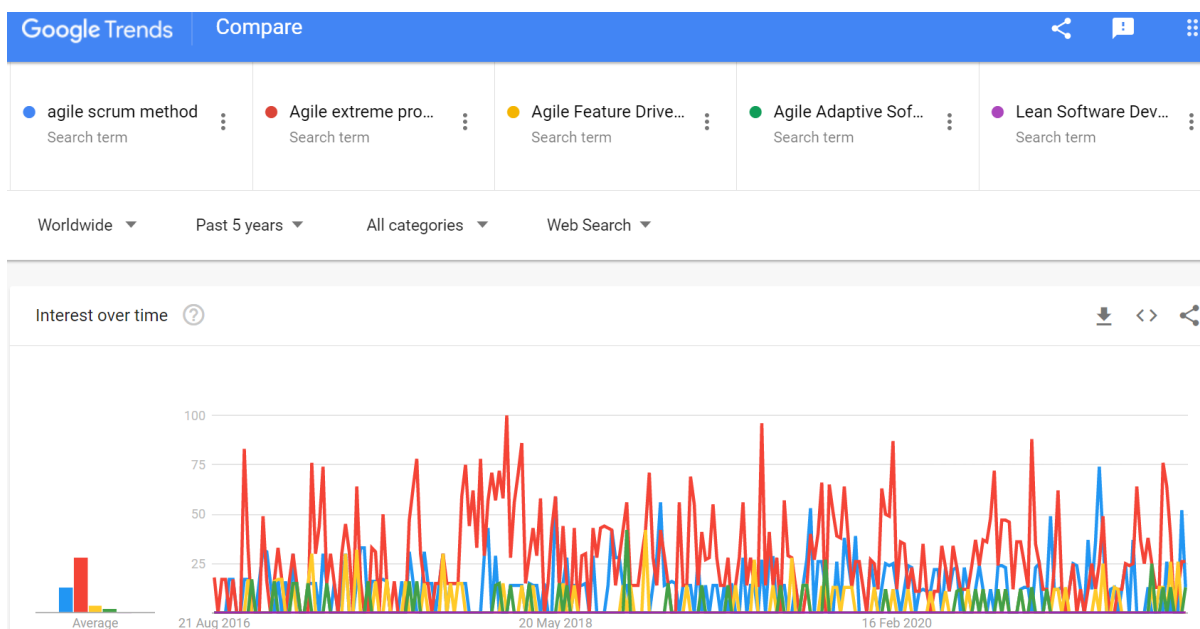


Figure adopted from (Digital.ai, 2021)

It is also evident from the google trends that scrum and extreme programming are the most frequently searched agile methodologies compared to others. We consider Scrum over Extreme Programming for this research as Extreme Programming follows a pre-defined project plan accepting only few agile methodologies with its primary focus on technical aspects in software development phase such as pair programming etc., rather than the managerial aspects whereas the Scrum method develops the highest values requirement for the customer first in an iterative and incremental manner with its focus not only on development but also on managerial aspects (Tarwani & Chug, 2016).

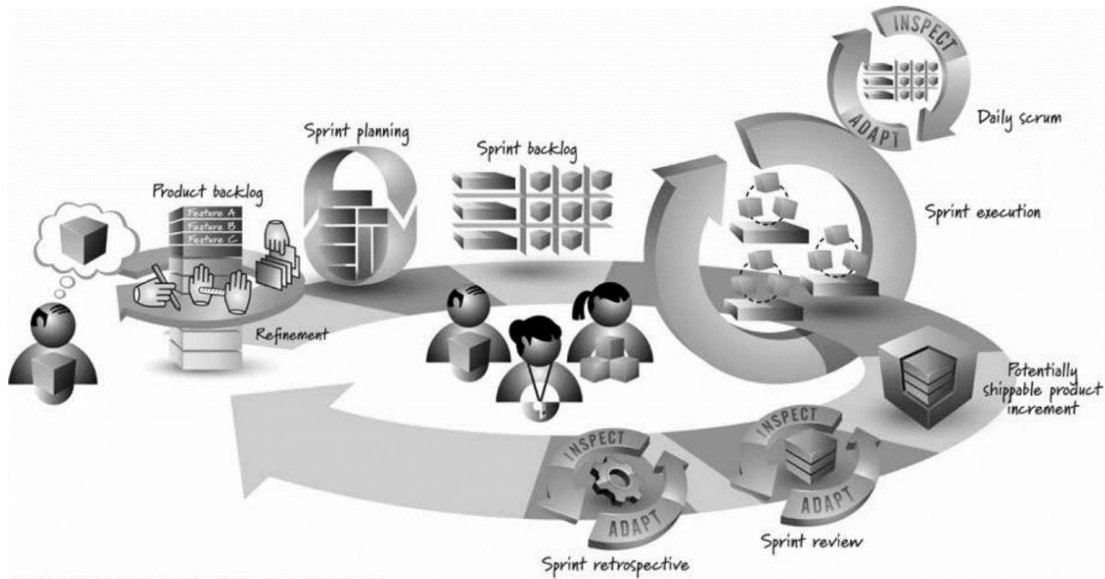


Having discussed the rationale for choosing Scrum as an agile method for this research, we will examine in a nutshell, what a scrum method means and its features in the below paragraph.

Scrum is a project management framework that integrates agile principles into organising and executing the work and helps organisation deliver highest business value in shortest time by empowering self-organised teams to determine the right course of action and appropriate time to deliver the priorities set by the business (Babaian, 2019). In general, a scrum team is composed of Product Owner focused on delivery aspects, Scrum Master leading the scrum team, development team and the process involves the following (Hayat, et al., 2019),

- Formation of small self-organising teams that is cross-functional.
- Formation of iterations or sprints that are not longer than one month in duration, usually most organisations follow a 2-week sprint.
- Breakdown of project scope into small, concrete deliverables such as features, defect fixes resulting in a list of iterative deliverables called product backlog.
- Planning the tasks for the sprint based on the effort estimation from sprint team and priorities obtained through collaboration with customer and insights from previous iteration.
- Sprint execution is a time-boxed period during which the entire development team collaboratively work on tasks from planning to development to retrospective resulting in a product that meets the sprint objectives and delivered to the customer.
- A time-boxed event called daily scrum chaired by Scrum Master is scheduled on a daily basis where the team members provide updates on tasks and discuss impediments blocking the project progress.
- Sprint review commemorates the end of the sprint when the sprint team and project stakeholders assemble to review the tangible results of the sprint, obtain customer feedback and discuss readiness for the release.
- Sprint retrospective is a time-boxed event at the end of each sprint chaired by Scrum Master where the sprint team share their lessons learnt and recommendations based on current sprint to help the team to continuously improve their ways of working.

In general, a scrum team works based on the agile manifesto principles discussed earlier: Self-organisation, transparency, collaboration, adaptation, prioritization, focus, accountability, commitment, rhythm, inspection and feedback (Babaian, 2019).



Agile Scrum Framework - figure adopted from (Babaian, 2019)

RESEARCH METHODOLOGY AND RESEARCH METHODS

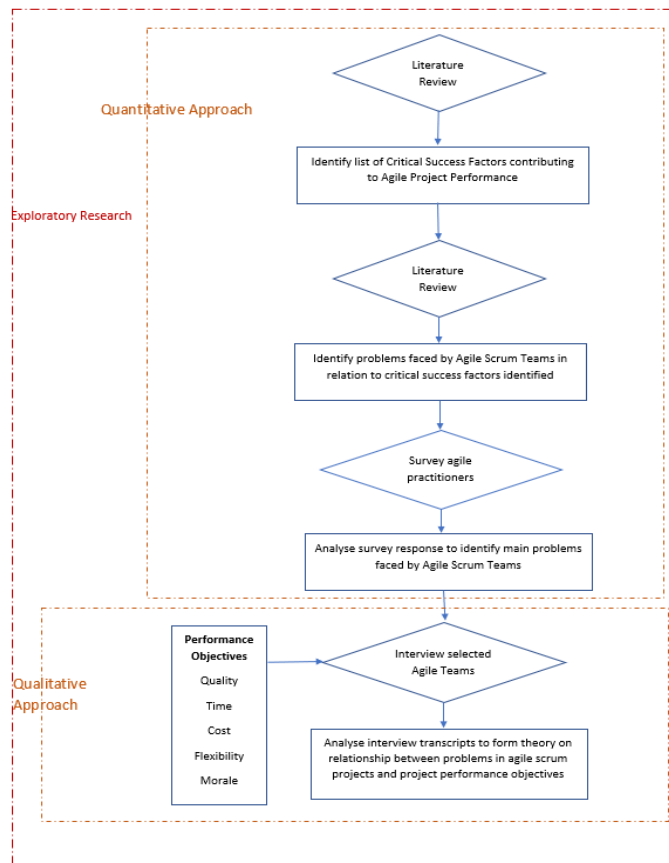
Research Methodology

The research method adopted is exploratory research aimed at identifying the main problems that impact the project performance in globally distributed multi-vendor agile scrum projects. The research follows a mixed approach of quantitative and qualitative approach, i.e., quantitative approach to identify the main problems in agile scrum projects and qualitative to study the inter-relationship between these problems and their impact on project performance. The intention of following a mixed approach is that a survey of agile practitioners in Software Development industry can be performed and the resulting variety of data can be quantitatively assessed to arrive at a solution for RQ2, i.e., identify main problems in multi-vendor globally distributed agile scrum projects and the opinions of agile scrum practitioners obtained by conducting interviews can be qualitatively assessed to arrive at a solution for RQ3, i.e., inter-relationship between the problems and their impact on project performance. Quantitative approach involves generation of data that can be quantitatively assessed to interpret solutions to a problem and qualitative approach involves obtaining subjective assessment of attitudes and opinion of focus groups through process such as interview (Kothari, 2004).

Research Design/Activities involved in the Research

Following are the activities that was performed as part of this research:

1. Identification of CSF areas and common problems leading to project failures in agile scrum teams with respect to those critical success factors by performing literature review.
2. Data Collection involving survey of agile scrum practitioners in software development to identify the current industry trends on the problems identified through literature review.
3. Analysing the survey data using statistical analysis method such as mean or mode calculation of the survey responses to reduce the whole list of problems to a reduced list of critical problems i.e., those that occur frequently and have more impact on project performance.
4. Identifying Agile Scrum teams and interviewing to study in detail the causes, impact and any measures they adopt for prevention of the main problems identified.
5. Analysing the results of the interview and formulise a theory on the inter-relationship between the problems and their impact on the project performance.



Literature Review

A wide range of literatures such as Journal Articles, Conference Proceedings, documents from professional bodies, IT organisations, and miscellaneous information such as blogs, related to agile project management, agile scrum, project management, and outsourcing that were reviewed to identify the critical success factors of projects, associated failure modes or problems and how it relates to the research scope – multi-vendor globally distributed agile scrum projects.

Knowledge Sharing

(Desouza & Evaristo, 2004) classifies knowledge generated by projects into three categories – *Knowledge in Projects* where each team member knows what, how, why, when, where and by whom an activity is being performed such as project schedules, minutes of meeting, training materials etc., *Knowledge About Projects* where organisation knows details of all projects in pipeline for effective planning and control of resources such as project objectives, customer expectations, resource allocation, timelines, cost and benefit analysis etc., and *Knowledge From Projects* is the knowledge obtained by carrying out projects which serves as a key enabler of success in future projects through organisational learning.. (Al-Salti, et al., 2011) classifies knowledge in IT projects into two types: *Business Knowledge* that enables IT professionals to understand business terminologies and interact with business partners, such as knowledge of business process, risk assessment, project management etc., and *Technical Knowledge* to convert business requirements into a working software, such as knowledge of programming, design and testing etc. Software Engineering is a knowledge profound domain, in which the success factor relies heavily on one's knowledge in the following areas: analysis, design, build, testing, IT infrastructure and implementation. Most IT organisations seek to foster knowledge sharing between team members and develop a centralised knowledge repository containing application information documents, technical design, defect reports, etc., to facilitate mutual learning and learn from the peer's past experience, i.e., if a tester finds a defect, the team can access the knowledge repository to know if similar defects were encountered in the past and their root cause to quickly resolve the defects, thereby reducing delays in delivering the project. Organisations aim to develop knowledge sharing between individuals and teams as efficient re-use of knowledge is seen as an effective way to deal with repetitive problems (Hansen, et al., 1999). However, this can become difficult with project team distributed globally and even more challenging with the team comprised of members from multiple vendors. Globally distributed project teams, comprising of two or more resources working together to accomplish project objectives from various geographical locations often face difficulties in transferring knowledge across different sites (Oshri, et al., 2008) as remote teams follow local policies for working, training and learning that may create a hindrance to shared practices for knowledge sharing; differences in technical skillsets, infrastructure, engineering tools and methodologies can obstruct knowledge transfer; differing time zones can limit the window for real

time interactions to discuss opinions and achieve consensus amongst all team members. Nature of relationship and organisational differences either strategically or geographically are notable factors for determining successful knowledge sharing between organisations (Al-Salti, et al., 2011).

Unclear Business Goals

Team members of multi-vendor projects often work on concurrent tasks focusing only on their own deliverables without due consideration to other vendors deliverables, and this self-centric approach often leads to misalignment and limited understanding of overall business goals (Akshanthula, et al., 2019). In order to achieve the overall success of the project, the team needs to work in cohesion towards a common goal, i.e., the team needs to have a wider picture of WHY they're doing it, regardless of HOW and WHICH component they're delivering (Senapathy, 2021).

Lack of clear boundaries defining scope and tasks

Agile projects tend to adopt cross functional teams over the siloed teams commonly found in traditional waterfall projects, to promote self-organising and managing, flexibility, accountability, productivity, and eliminate bottlenecks, to provide on-time delivery of better-quality products (SAFe Scaled Agile Framework, 2021). According to (Schwaber & Sutherland, 2020), cross-functional team is that possess all skills required to create value and responsible for activities from research analysis, development, testing, implementation, maintenance, and stakeholder collaboration without dependence on others not part of the team. A cross-functional team is a cohesive set of individuals possessing different skillsets and having high degree of interdependence, focused on effectively delivering a common organisational goal (Holland, et al., 2000). However, there could be challenges with the effective functioning of cross functional teams, when the project team is composed of multiple vendors and members distributed globally. Often the lack of demarcating the scope and activities to be performed by each team member can result in one vendor team stepping into another vendor team's domain (Akshanthula, et al., 2019). Fluidity in work is productive, but the same in scope boundaries is counter-productive, that can result in stepping into other team members domain and duplication of work (Senapathy, 2021).

Poor Project Integration

Software development usually involves multiple interfacing teams working on different components of a project such as design, development and testing, which then grouped together constitutes a fully functional integrated project. Most project failures occur at the interface of sub-systems, as individual team members focus mainly on their allocated tasks with less or no attention towards their interfaces until or after they have completed their tasks (Rosenau & Githens, 2005). With multiple teams working together, the project can often face the risk of unresolved problems at the integration touch points, especially in projects involving multiple vendors (Akshanthula, et al., 2019). This risk can increase exponentially in multi-vendor globally distributed teams with increasing number of interfaces between the client, different vendors, and their own distributed teams. So, an effective project management that drives the project towards success, calls for a tighter integration at these interfacing points to manage and keep the system together (Senapathy, 2021).

(Struckenbruck, 1998) defines Project Integration as the process of bringing together all the interrelated and interconnected elements of a project, its tasks, subsystems, organisational units, and people at the right time to enable the project to function as a whole integrated unit as per the plan. Interfaces are often a problem for project manager as most of the problems encountered are usually due to two organisational entities going in different directions and can become even more complicated when the interfacing units are not under the direct management of the project manager or completely outside of the client organisation (Struckenbruck, 1998). He further goes on to classify the interfaces into three categories: *People interfaces* are those which represent the interface between people within or outside the project team, *Organisational interfaces* are those which represent the interface between different organisational units such as customer, contractors, sub-contractors, etc., that are usually outside the project organisation working on the same project; and *System Interfaces* are the product, hardware, or other types of non-people interfaces present in the system being developed by a project such as a product developed by another interfacing project, transmission of information from one task to another task etc. (Struckenbruck, 1998) further attributes three types of problems to these interface levels: *Administrative problems* such as roadblocks, changing priorities, conflicts involving different organisational units, people, or resources; *Technical problems* such as changing scope, non-consensus on decisions, alternative opinions and trade-offs on schedule, cost or quality by interfacing teams; *Customer or client problems* such as the interfacing teams not developing a common understanding or conforming to customer specifications.

These problems can be hard to deal in an agile scrum environment where the prime focus is to satisfy customers through early and frequent delivery of working software, with a preference to shorter timescales, usually in sprints lasting couple of weeks. In software development, there is often dependencies on people, resources, infrastructure, systems, and tools outside the project team to complete an assigned task, i.e., project might need infrastructure personnel from another team for implementation of their software being developed, or a dependency on another system or task that is being developed or performed concurrently by an interfacing team or project to be completed, in order to accomplish their own tasks or the system being developed. So, problems such as changing scope, changing priorities, conflicts with resource availability etc., from interfacing teams can hinder the agile scrum team from fulfilling their tasks within the sprint timelines. Agile scrum approaches work better for smaller independent teams with well-defined interfaces but the approach does not seem to be beneficial in large and complex team structures with considerable overlaps between interfaces (Rising & Janoff, 2000).

(Richter, et al., 2016) in their research article on problems in agile software projects within traditional organised corporations, emphasises almost all the agile projects are run within a traditionally organised corporations and face difficulties in translating agile process, methods, values and metrics to their non-agile counterparts, especially at interfaces to rest of the organisation. It is not uncommon for an agile project to have members from vendor organisations that are traditionally managed or much of the client organisation that owns the agile project or atleast the stakeholders dealt with are from a non-agile traditional team. This often results in the agile project to bend some of the agile processes to integrate with the non-agile interfaces. For example, an agile project team reporting to a project sponsor or manager from a non-agile team would expect the project team to issue a traditional project status report on a day-to-day basis that includes burndown chart, key issues or risks etc., but this leads to a conflicting situation between stakeholder and agile project team as it contradicts the agile status reporting of daily-stand-up meeting where the team gets together to discuss the project progress.

Lack of Realistic Schedules

As stated in the agile manifesto, agile processes maintain work at a constant pace to promote sustainable development by the team. So, is the necessity for working in sprints on realistically achievable targets as it enables the team to work quickly and effectively to deliver working software often and maintain a sustainable pace, without leaving the team knackered and demoralised due to long working hours and overtimes (Apptio, 2018). (Popli & Chauhan, 2013) calls the estimation of project scope in terms of time and cost and plans based upon those estimates as a significant factor that determines the success or failure of agile projects and also an unbiased estimation process combined with a biased software development planning process can enable projects to develop realistic and credible plans to fulfil project objectives and commitments. (Popli & Chauhan, 2013) further mentions *velocity measurement through disaggregation and expert opinion* as the most widely followed estimation approach by Agile scrum teams and the process involved is as follows: a) Requirements, features, or tasks to be performed are called user stories and a story point representing the relative size and time is assigned for each user stories through expert comment or human intuition methods, commonly such as planning poker by developers, testers and designers, etc., b) A joint consensus is made by the project team on the story point assigned and is in turn used to estimate the velocity of the project, i.e., the number of story points that can be developed per iteration or sprint. For example, the project team with a baselined team velocity of 20 story points based on previous iterations, may chose user stories with story points (7, 7, 3,3), or (7,5,3,3,2) or (7,7,5), etc., from a backlog of user stories with story points (7, 5, 3, 3, 5, 2, 7) such that the total story points of user stories selected does not exceed the team's maximum velocity of 20 story points per iteration. However, the estimation and planning of agile projects can become challenging when combined with the factors concerning multiple vendors and globally distributed teams.

(Whittaker, 1999) based on his empirical research emphasises that most of the project failures are due to scheduling overruns caused by unrealistic project schedules based on poor estimates, under-estimated scope, or over-estimated skillset of resources either by the project or its vendors causing an inability to meet commitments as planned. (Ramasubbu & Balan, 2012) states project estimation as one of the significant challenges in globally distributed software projects as in most cases the inherent properties of a distributed project such as work dispersion, work time overlap between distributed centers, experience of team members and other project performance factors like productivity, defect ratio, reuse and rework are not considered in the estimation process. Another major problem in sprint estimation and planning is that the product owner or the domain experts are those estimating story points and planning the tasks for a sprint and the team only accepts the tasks (Apptio, 2018). This practice can prove to be a problem especially with distributed teams. For example, in an offshore/onshore scenario, estimation and planning are performed by co-located members at the client site, usually the most experienced tester or developer, who based on their intuition may estimate the story points lower than it would be for the comparatively less experienced offshore counterparts and ignoring other factors characterising remote distributed teams. Team members may overcommit or overestimate their abilities early on at the project leading to a lack of pacing and work overtime to keep the sprint on track, often causing rift between team members, aggravating an already tenuous situation of defects and delays in the project (Overby, 2019).

Another common problem faced in agile project estimation is the price-to-win situation. In a joint estimation meeting where each member of the team estimates the story point, such as by planning poker, regardless of the team the task is to be actually performed, it creates a situation for the team owning the task to lower the estimation than it would actually require to perform the task. Price to win

situation arises when the best price to win the project contract is estimated rather than the cost that would be ideally required to develop the software functionality, which can eventually lead to delayed deliveries, force the project into overtime to avoid schedule over-run (Popli & Chauhan, 2013). (Dingsoyr, et al., 2006) in their article mentions the vendor or team who generally owns the task could unconsciously lower the estimation than they would during cross organisational joint estimation activities, when the customer or other vendors who usually don't perform the task or inexperienced with the task that is estimated, lowers the estimation implicitly. Political forces within a project or organisation such as managerial influence on estimates and pressures to stay within or meet the estimate, time or cost constraints by customers can often lead to biased estimates simply to satisfy customers or managers, thereby leading to inaccuracy (Popli & Chauhan, 2013).

Lack of Effective Communication

Effective communication is a critical factor for project success regardless of the development approach and is even more crucial in agile software development projects as informal and spontaneous communications plays a significant role. Agile manifesto insists the prominence of effective communication by emphasising individuals and interactions over process and tools in 2 of its 12 principles, "The most efficient and effective method of conveying information to and within development team is through face-face conversation" and "Business people and developers must work together daily throughout the project" (Beck, et al., 2001). Success of a project is dependent upon communicating key project information such as scope, cost, schedule, project outcomes and benefits on a timely manner through appropriate channel using non-technical language that provides sufficient clarity and detail to appropriate audience (Project Management Institute, 2013). Communication is a mediating element to achieve coordination and control through continuous and unambiguous exchange of information (Carmel & Agarwal, 2001). Communication is an activity that is essential to manage dependencies between different actors in a process, such as *task-resource dependency* that defines the resources for each task with key actors as software development team and management; *producer-consumer dependency* to ensure consistency between requirements and software development output with key actors as customer and development team; task-subtask dependency to break tasks into units of work that can be performed by development team with key actors as customers, development team and management (Pikkarainen, et al., 2008).

However, it can be challenging to establish spontaneous and regular communication in agile projects with distributed teams spread over different continents and time zones with less overlapping time and documentation to interact synchronously and develop a common understanding. (Korkala & Abrahamsson, 2007) in their research article mentions that distributed teams take more than twice the amount of time required to complete tasks by co-located teams due to communication and coordination problems and that almost three-fourths of their survey respondents indicate inefficient and sporadic communication influenced by rapidly changing requirements cause serious problems in distributed agile projects. A report by (Project Management Institute, 2013) highlights atleast 2 out of 5 projects do not meet the project goals within the intended budget and atleast half of those unsuccessful projects are related to ineffective communication. Continuous interactions and meeting to facilitate open discussion could not be arranged easily with remote members due to unavailability of virtual communication tools, different time zones or atleast the member consider the communication technologies used are less rich than face-to-face communication, which explains the lack of access to all information by the off-site members in distributed teams (Yague, et al., 2016). Although the obstacles are reduced by recent advancements in communication technologies and through efficient usage of combination of tools to manage team, process and product, such as *Video-Conferencing* tools

like WebEx, Microsoft Teams supporting bi-directional face-to-face or one-to-many communications; Project management tools like Atlassian Jira for project planning, managing task-resource dependencies, and progress reporting, but effective transference of right information and overcoming misunderstanding has remained a challenge in distributed agile projects. (Amerongen, 2008) in his article mentions that despite the advancements in communication technologies like IM, web meetings and videoconferencing, there is nothing more effective than a face-to-face communication to ensure a clear and unambiguous understanding of the project requirements and approach. Although communication technologies bridge distributed teams together easily, the main drawbacks of it that (Amerongen, 2008) considers over face-face communication are as follows:

Distractions: Web/teleconference meetings can make participants vulnerable for distractions causing them to lose concentration and focus on the meeting, if they are more inclined to work parallelly on other tasks, email backlogs, and other IM sessions.

Duration: Web/teleconference meeting can take significantly more time to deal with same number of topics when compared to face-to-face meetings as it requires considerable time and effort to make participants understand the proposals and opinions and often participants can lose the context in which who the speaker is and seeks to clarify.

Non-Verbal communication: Non-verbal is the primitive way of ensuring the information is perceived by the recipient. Although video-conferencing can help overcome this obstacle, but it may not provide a complete picture which makes the sender of the information blind to non-verbal signals. Face-Face communication can help alleviate disconnects, challenge opinions, receive instantaneous feedbacks by conveying information that are not purely verbal such as posture, head nodding, inflection of voice, silence, and hand movement etc (Accenture: SolutionsIQ, 2015).

In most cases, the leader of the meeting will be unable to notice the distraction and lack of concentration in the participants and cannot correct or validate if the participant has missed any important information about the project, which eventually can have an undesirable effect on the end-results of the whole project. Hence, (Amerongen, 2008) suggests face-face communication as the most effective in a fast-paced agile environment as it facilitates non-verbal communication, getting participants focused and involved, better clarity on discussion through whiteboards, stakeholder commitment and instantaneous approvals on project approach and solutions, and personal relationship development. (Yague, et al., 2016) based on a recent survey in their article, mentions face-to-face and emails communications are still the most widely used communication channel in agile teams alongside short meeting and audio/video conferencing.

Another common problem in distributed multi-vendor teams is the chance of information distortion as teams don't communicate directly and often the communications happens through the client, mainly due to the lack of trust between different organisations (Korkala & Abrahamsson, 2007). (Pattit & Wilemon, 2005) mentions teams of sheer size can often lead to high communication quality deterioration owing to the number of communication links between team members, and calculates the number of communication links between team members using the formula, $(n(n-1))/2$, where n represents the size of the team. For example, a team with 10 members will have 45 communication links, which is lesser compared to 435 links in a team of 30, thereby the team with fewer links can communicate vital information effectively and with less deterioration. Lack of direct and informal communication can lead to lower awareness of others activities and poor coordination in planning and

executing the work in relation to others activities in a direction towards the project objective (Korkala & Abrahamsson, 2007).

Interruptions in Agile Projects

As emphasised in the agile manifesto that individuals and interactions precede over process and tools, agile projects tend to have changing requirement and increased interactions between team members. While volatile requirement and increased interactions brings with it interruptions that can have positive effects on projects performance such as quicker task completion, improved process flexibility; at times it can also have negative impact on employee productivity. (Wiesche, 2021) mentions that interruptions are common in agile software development teams due to two underlying agile characteristics – one is the need for quicker response to change and high involvement with customer, and the other is the continuous interaction and coordination with team members. Interruptions are intrusions or distractions caused by another individual, external stimuli or secondary tasks that interrupts the flow and continuity of one's primary task by diverting their focus and attention away from it, thereby bringing the primary task to a temporary halt (Jett & George, 2003).

Interruption is a form of impediment, such as extra processes consuming more time and effort without adding value, delays due to resource unavailability or additional features added, context switching where attention paid to more than one activity at a time, that can slow or inhibit a team's ability to enable steady flow of work through the system (Power & Conboy, 2015). (Bruhn & E, 2003) states the interruptions despite viewed as negative can have positive influence as well as interruptions can influence knowledge sharing through interaction between teams, acquisition of knowledge on new approaches thereby creating opportunity for change, switching from routine work to conscious information processing. Besides these diverging perspectives on interruptions in software development projects, multi-vendor distributed agile projects due to its complexity involving multiple organisations and client, can become even more vulnerable to negative aspects of interruptions resulting from high number of meetings and continuous interactions between offsite-onsite, clients-vendors, and vendor-vendor team members; Changing requirements, scope, or priorities by the client or interfacing vendor teams, which can interrupt an individual or team's primary task. (Wiesche, 2021) based on the nature of interruptions classifies it into three categories as follows:

Development-related: Caused by impediments associated with software development tasks such as malfunctioning code or development tools, poorly specified requirements by customers, changes in requirement, backlog, scope, or priorities, missing information or information required from another team that causes the team to postpone the task until the impediment resolved or information received. Even though agile projects embrace changes in the requirements, an un-anticipated change to user story by the customer towards the end of sprint can be impossible for a project team to incorporate in the ongoing sprint and can lead to the current work being discarded or additional rework in successive sprints. (Wiesche, 2021) highlights the most common problem in agile projects as the frequent fundamental changes to user story or reprioritisation of requirements by the customers whilst the team works on it, forcing the team to plan a new sprint either to rework existing code or develop a new code.

Interaction-Imposed: caused by collaborative nature of agile projects such as high number of formal meetings, team members and stakeholders frequently interrupting the work processes. It is too common in distributed agile teams to have continuous interaction between the peers located onsite and offsite, client's business teams, counterparts in vendor teams or project stakeholders over IM or telephone calls to exchange knowledge, discuss work processes and progress. Although these interactions improve team members awareness and help removing impediments, the time spent on these interactions can also have negative consequences on the productivity by delaying their allocated

tasks, forcing the team member to work overtime. (Wiesche, 2021) in his research article mentions agile team members are constantly interrupted by customers and other project stakeholders seeking project status updates and also that daily stand-up meetings are seen as inefficient and focused excessively on status reporting although it's aimed at enabling the team to concentrate on development tasks by encapsulating the team from constant external interruptions such as the stakeholders seeking status update.

Environment-Imposed: Caused by external factors such as office setup, telephone calls, breaks, and team members working in multiple project teams can be interrupted due to the duality of their tasks. Another common problem in agile projects that (Wiesche, 2021) highlights is that a team member working on multiple projects in parallel due to resource constraints, expertise on a specific area, or the project not requiring a full-time resource, can cause interruptions in decision making when other team members require the input of shared resource on overlapping urgent issues. Resources shared across multiple projects are commonly seen in outsourced projects following service-based model where the under-utilised resources are assigned to multiple projects to attain full utilisation.

Conflicts and Lack of Trust

Ultimate objectives of the project are fulfilled mostly in multi-vendor outsourced projects, but it occurs with considerable number of conflicts and friction between vendors, vendor and the client, affecting the quality and speed of the deliverable, and most importantly motivation and productivity of the team members (Banerjee, 2012). Trust and Conflicts are immanent issues of any inter-organisational partnerships where collaboration is required to share their expertise, competencies and skills for mutual benefit (Panteli & Sockalingam, 2005). (Serra, et al., 2021) suggests establishing an environment of trust that is often influenced by organisation culture, as a critical success factor for projects as it improves motivation and team cohesion through collaboration, cooperation, transparency in communication and open-mindedness to relationships. (Panteli & Sockalingam, 2005) defines Trust as a dynamic state of remaining confident and positive influenced by intuitive expectation on actions of member or organisation through different stages of relationship, until the trust shifts to being unconditional which enables synergistic relationship and superior performance. Projects involving multiple stakeholders and vendors may often clash and resist to work together in harmony due to diverse organisational culture and structure, that can be heavily influenced by politics and competitor's actions, ultimately affecting the project performance (Bruyne, et al., 2021). Conflicts are inevitable in projects with team members from different organisations, each with different organisational objectives and interests of business expansion is that makes multi-sourcing vulnerable to such conflicts (Banerjee, 2012). Client might be the unifying factor in multi-vendor teams but a sense of differentiation towards a member of another vendor can be the cause for most conflicts and not willing to co-operate with each other (P.Hui & Beath, 2001). There is more potential for profound and prolonging conflicts with more the number of vendor organisations in multi-sourcing due to the distributed nature of vendor teams each with their own organisational objective and culture. (Panteli & Sockalingam, 2005) defines conflict as an evident struggle exhibited by at least two inter-dependent members or organisations with incompatible goals, scarce benefits, or intervention from other parties in achieving their goals and the sources can be competition over insufficient demand, power differentials, Vagueness over task responsibility or ownership, negative affinity towards other team members, denial of one's characteristics like self-image, values and sensitivities. (Banerjee, 2012) in their research suggest the most common causes and consequences of such conflicts in multi-sourcing as difference in knowledge and skillsets of vendor teams; Competition leading to withholding information to gain rapport over

other vendors and developing knowledge gaps/deficiency through improper knowledge transition, training and documentation; Working in silos; Power struggles, politics and finger pointing by vendors to influence clients preference to their side, which can ultimately lead to de-motivation, reduced productivity, increased defect rates, and lack of trust and rapport between vendor teams. Higher levels of knowledge and organisational diversity will lead to higher levels of relationship conflicts, which in turn leads to lower team performance (Liang, et al., 2007). Conflict is like a twin edged sword that can be both functional and dysfunctional with the negative factors often outweighing the positive factors, where the positive side can give rise to healthy competition and professionalism, and negative side to knowledge gaps, information withholding, reduced trust and collaboration (Banerjee, 2012).

(Rosenau & Githens, 2005) mentions hygiene factors such as achievement and recognition, roles and responsibility, advancement and growth as the most powerful motivators and their absence, especially recognition of achievements could lead to demotivation and hinder employee productivity. In multi-vendor projects, all vendors try to demonstrate themselves and deliver tasks to their best ability, in order to strengthen their organisation's interests in business expansion with the client. However, at times the vendors achievement or dedication towards the work can be left unnoticed by the vendor organisation or the client, most commonly in agile projects where individual performance can be overshadowed by collaborative nature of work, thus leading to employee discontent and demotivation. Micro-management by the client or vendor management and being biased towards certain employees or organisations can lead to difficulty managing teams and recognising their accomplishments, eventually leading to employee demotivation (Banerjee, 2012).

Constantly Changing Resources

Another common problem in large scale multi-sourcing projects is the constantly changing resources. This could be due to many reasons such as large number of simultaneous projects and shortage of resources with niche skillsets or expertise creating the need for vendors to move resources around to staff the projects. Although the approach of constantly moving resources between projects may seem beneficial in traditional projects but not in agile projects where the team is staffed with stability in mind to facilitate continuous collaboration and improvement. (Shrivastava & Rathod, 2015) in their article mentions team continuity or keeping effective teams unchanged as one of essential factors for agile project success because quality of software is not just determined by people's technical skills but by their relationships and ability to accomplish tasks together. (Beck, et al., 2001) quotes the principle "At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly." in their Agile Manifesto, which reinforces the benefits of team stability. (Rajkhowa, 2020) views this principle as a continuous improvement approach by stable teams where the team through empiricism inspects their deliverables and processes retrospectively on a regular basis to identify the areas of improvements and the necessary action items, then adjusts their ways of working in the direction of those improvements identified. The team chemistry and velocity are disturbed and needs to be re-established every time a new member is onboarded into the team, so insulating the agile teams from frequent resource change is one solution to this problem as teams that stays together, unchanged, learns to work best together and support each other to advance towards their team's goal (UpGrad KnowledgeHut, 2019). (Hazrati, 2009) highlights one of the major problems in agile projects is the matrix allocation or moving people between different projects as high performing agile teams are formed by people who go through a prolonged period of forming, storming, norming and performing in a single team and better approach is to maintain the high performing team as intact

and as long as possible. Another problem (Hazrati, 2009) highlights is that instead of assigning people to projects or multiple projects to a team to work on simultaneously, organisations should queue up projects for the team even though it can be challenging considering the business urgency and time to market, but this way projects get better prioritised and enables teams to achieve throughput and delivery on selected projects rather than working on multiple projects and focusing on none. Hence, it is better to work projects around teams rather than people across projects.

Inability to breakdown backlog into smaller user stories or Horizontal planning of user stories

(Verwijns, 2017) mentions the key factor that determines the success of agile scrum projects is the team's ability to perform a just-in-time refinement of product backlogs and breakdown of work into many small functional items rather than a few larger ones that can disrupt the flow of work and increases the risk of failing the sprint. Agility requires "Backlog grooming" which is a skill acquired through learning and collaboration between business and technical people, that enables scrum team to split large epics or requirements into smaller user stories or product features to deliver the most valuable features first, although delivering low value features towards the later stage is likely to involve rework, it is better than no work (James & Walter, 2017). (Rajkhowa, 2020) relates one of the twelve principles of agile manifesto "Simplicity—the art of maximizing the amount of work not done—is essential" to pareto rule that emphasises 80% of output is received with only 20% of work and hence the focus should be on prioritisation and development of extremely crucial product features that produce more value first. Organisations often chose multiple vendors owing to their competency or speciality skillset in specific areas. Hence, the agile scrum teams composed of multiple vendors that are cross-functional and responsible for end-to-end delivery of product features can resemble a component team or a team formed through a combination of sub teams, i.e., design, development, testing team etc. (Rubin, 2012) defines feature team as a cross-functional and cross-component team that can deliver customer features end to end; whereas a component team is a component or sub-system team that can deliver only a part of a customer feature or a community of people who possess similar skills or perform similar roles within a team. This will result in silo thinking by each of the vendors and the work break down into units based on the work that needs to be done such as design, coding, and testing, rather than based on product features, often resulting in teams unable to split user stories into modules that can be completed within a sprint and leading to the end-to-end development activities spanning over more than one sprint. (Verwijns, 2017) suggests horizontal breakdown, i.e., the approach of splitting up user stories horizontally based on the tasks to be performed, doesn't work well in agile scrum teams, and the reasons being: Splitting up product backlog horizontally, would certainly result in smaller individual work items for design, database, coding and testing, but they don't deliver working, demonstrable software and business value on their own. Only the outputs of all the individual work put together generates business value. Horizontal breakdown often results in silo thinking, as every member is resourced from one of the silo teams required for software development, such as design team will be responsible for design, developer writes the code, and testing team performs testing. This results in bottlenecks when one of the preceding tasks is delayed or interrupted, if developer cannot complete their tasks, then the task following it, i.e., testing cannot take place, thus impacting the entire sprint.

Lack of effective governance at agile team level:

(Talby & Dubinsky, 2009) mentions an effective governance process that enables effective project monitoring and control is key success factor for large scale agile software projects.

(Draper, 2017) defines governance as a process of setting directions and making decisions as to who (people/organisations) does what and how, i.e., the process by which they intent to achieve it.

Although organisations with multiple vendors would have outlined in each of the vendor contracts - a RACI that defines the responsibility of each vendor and a dependency matrix that illustrates the dependency between each vendor. However, these governance factors outlined in a vendor contract would usually be at the organisational level and may not be effective at agile team level leading to conflicts and adverse negotiations over contract between vendor teams (Belden, 2019).

Change Resistance to Agile:

(Overby, 2019) mentions transformation of employees and company culture from traditional to agile model as one of greatest challenge faced by any organisation. The reasons for change resistance could be attributed to several factors such as lack of knowledge of agile model; Cultural issues such as Lack of trust, devaluing collaboration, and command-and-control thinking; General fear of change and unknown; Loss of hierarchy, control and authority by managers (Gandomani & Nafchi, 2016). While overcoming this resistance can be challenging at single organisation level, it can be even harder when multiple organisations work together in the transformation towards agile model.

Lack of Specialised Roles

Scrum processes replaces specialised roles such as Testing Manager, Build Lead that are traditionally used for controlling and managing of teams by undifferentiated roles such as scrum master, product owner that can sub-optimize effective functioning and efficiency of agile scrum teams (Overby, 2019). Lack of specialised roles can lead to chaos and challenges managing the vendor teams by client organisation, and leadership/management support by team members when dealing with conflicts and issues with other vendors/clients.

Around 25 problems that can impact project performance in agile projects was identified through literature review and are listed in the table below.

P.No.	Problem	Literature Review Section
P1	Lack of clear boundaries defining scope and tasks often lead to one vendor stepping into other vendors domain	Lack of clear boundaries defining scope and task
P2	Problems such as changing priorities, changing scope and conflicts with resource availability by teams within the project and interfacing teams outside project prevents tasks from being completed within sprint timelines	Poor Project Integration
P3	Difficulties in translating agile process, methods, values and metrics to non-agile or traditionally organised management teams, such as while seeking support for client-vendor, vendor-vendor conflict resolution; status reporting to a non agile stakeholder etc.	Poor Project Integration
P4	Estimation and Planning are mostly performed by co-located teams at client site, usually by experienced team member representing each functional area - build, test etc., and tasks accepted by less experienced or remote team members leads to slippage of schedule	Lack of Realistic Schedules
P5	Joint estimation/planning in agile scrum teams are often influenced by other vendors, management and clients resulting in user stories estimated lower than it would be required results in cost and schedule overrun	Lack of Realistic Schedules
P6	Lack of direct communication between vendor teams and communication often facilitated through clients results in awareness and co-ordination problems	Lack of Effective Communication
P7	Sheer size of the team, i.e., more number of communication links within the team leads to deterioration in communication quality or information exchanged	Lack of Effective Communication
P8	Lack of proper communication tools, less overlapping time due to different time zones, different cultural and language background are obstacles for remote teams to effectively communicate in real time or participate in meetings with counterparts from different locations	Lack of Effective Communication
P9	Online meetings are not as effective as compared to face-face meetings due to distractions from parallel working and communication, considerable time and difficulty to explain project matters and opinions, wait for approval/feedbacks, lack of non-verbal cues	Lack of Effective Communication
P10	Frequent changes in requirements or user story especially towards the end of a sprint causes excessive delays, re-planning and rework	Interruptions in Agile Projects
P11	People often allocated to multiple projects/teams on shared basis results in discontent and stress personally and interruption to team members seeking input on overlapping urgent issues	Interruptions in Agile Projects
P12	Continuous interactions from peers onsite or offsite, clients and vendor teams seeking update on project progress, help or information on work processes, interrupts the primary tasks assigned and eventually leads to working over time to meet the sprint goals	Interruptions in Agile Projects
P13	High frequency and excessive time spent on meetings and agile ceremonies such as daily stand-ups, retrospectives, sprint reviews etc reduces the time available for development tasks and eventually leads to working over time to compensate the time lost	Interruptions in Agile Projects
P14	Sense of differentiation towards members from other vendors, competition, power struggles and politics by vendor teams to influence clients preference to their side often leads to conflicts, lack of trust and collaboration in the project	Conflicts and Lack of Trust
P15	Absence of hygiene factors at work such as lack of advancement, growth, recognition of achievements by management or client leads to demotivation and reduced productivity	Conflicts and Lack of Trust
P16	Constantly changing resources or adding new resources in the project affects team's chemistry and velocity	Constantly Changing Resources
P17	Concerns about sharing tools, knowledge or proprietary information such as development tools, stubs etc., created by one vendor with another	Knowledge Sharing
P18	Inability to breakdown backlog into smaller user stories and prioritise based on most business value that can be delivered within a sprint 'or' horizontal breakdown of stories based on tasks such as design, coding, testing, etc., instead of product features often leads to work spanning over multiple sprints and not delivering working software or business value at the end of each sprint	Inability to breakdown backlog into smaller user stories or Horizontal planning of user stories
P19	Lack of continuous Client/Business User involvement due to distributed nature of the project, such as communicating to business and getting feedback on matters	Lack of Effective Communication
P20	Lack of different skillsets, appropriate tools or restricted access to tools hampers the team to be fully cross-functional and makes the team heavily reliant on component teams in possession of necessary skills or tools	Knowledge Sharing
P21	Lack of clarity around roles or specialised roles such as testing manager, build master, project manager etc., can sub-optimize efficiency, control and smooth team functioning, especially in multi-vendor agile setup where resources directly report to a client manager	Lack of Specialised Roles
P22	Lack of Leadership/management support and commitment at different levels of management - client, vendor etc.	Lack of Specialised Roles
P23	Lack of project management competence or appropriate leadership style for agile projects by project manager or scrum master, i.e., project manager adopts an autocratic leadership characterized by individual control over all decisions and little input from group members	Conflicts and Lack of Trust
P24	Resistance/lack of acceptance of agile principles or wrong understanding/incorrect application of agile approaches, such as, management orders to work agile but the team resists to change, rapid transition to agile ways of working with less time to foster benefits of change/adequate training, or the team does not see a need for agile as the system developed is not modular or suitable for development in iterations	Change Resistance to Agile
P25	Lack of governance and documentation leads to problems with measuring and monitoring the performance, knowledge deficiency etc.	Lack of effective governance at agile team level

Survey of Agile Scrum Practitioners in Software Development

A survey questionnaire was developed using Google Forms comprising of two sections.

First section focused on capturing the background of survey respondents to filter the datasets that meets our research criteria: Globally distributed projects, Multi-Vendor Setup, and Agile Scrum methodology. Following questions were included in this section:

- Demographics of participant – survey participant location and their client location,
- Agile methodology used such as Scrum, Extreme Programming, Feature Driven Development, Adaptive Software Development, Traditional Waterfall,
- Distribution of project team: locally based or globally distributed,
- Number of vendors in development teams
- Role in Project team such as Developer, Tester, Project Manager, etc.,
- Location category such as offsite same country as client, offsite different country, and onsite client location,
- Employee type such as client organisation, vendor organisation or 3rd party contractor,
- Work experience in software development industry and in agile scrum projects.

Second section included the actual survey questions i.e., all the 25 problems identified in literature review. Each problem had two questions:

- First question is to capture the respondent's opinion about the problem in terms of Occurrence, Impact and Time to Resolve on a Likert scale of 1 to 10, with categories 1-3 as low, 4-6 as medium, 7-9 as high and 10 as severe. Reason for using a Likert Scale is that it can capture the agile practitioner's opinion on a problem in a quantitative form that can be further assessed to identify the significant problems
- Second question is to capture the respondent's opinion on the project performance factor impacted by the corresponding problem. One of the question from the survey is included in the figure below and the actual survey questions can be found in the link - https://docs.google.com/forms/d/e/1FAIpQLSdhKQ16ExvYmoc-syZnUQ5WyyvjDkQbir4T_2olr_WOj8uyCBA/viewform?usp=sf_link

Lack of clear boundaries defining scope and tasks often lead to one vendor stepping into other vendors domain *

	1 (Low)	2	3	4	5 (Med...	6	7	8	9	10 (Hig...
Occure...	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Impact...	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Time t...	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐

Which of these Project Performance factors are impacted due to the previously rated problem? *

- ☐ Cost
- ☐ Time
- ☐ Quality
- ☐ Flexibility
- ☐ Morale

Testing Draft Version of Survey Questionnaire

An initial version of the survey was tested by collecting responses from a few selected agile practitioners to assess the efficacy of the developed in gathering all the required data for our analysis.

Initial survey version used a 1:5 Likert Scale and after the analysis of test results, Likert scale was changed to 1:10 as the initial survey test results of 5-point scale indicated neutral point of opinion and less degree of differentiation. Hence, the survey was updated to a 10-point scale, as researchers claim it to provide more accuracy than a 5-point scale (Vidruska, 2014).

A link to the test version of the survey can be found here - [Problems in Multi Vendor Globally Distributed Agile Projects - Google Forms](#)

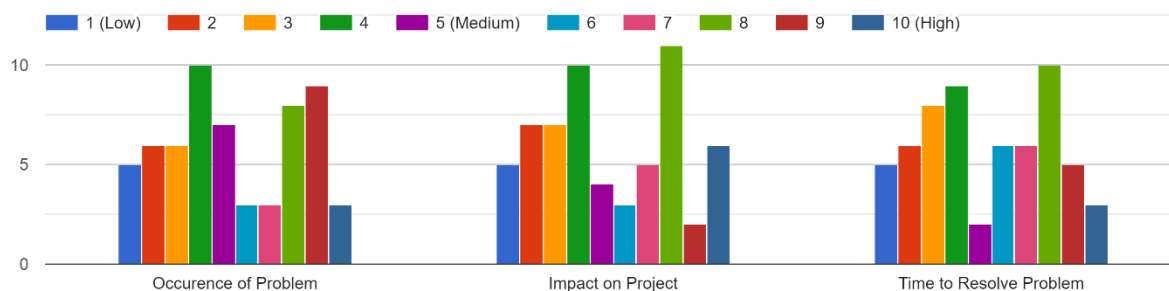
Updated version of the survey was circulated to contacts in software development industry through social media such as LinkedIn and Facebook.

Survey Analysis

A total of 60 responses were received from the targeted 100+ agile practitioners. Statistical analysis of survey responses through mode calculation to rate the occurrence, impact and time based on the most rated point on a point scale was found not to be an ideal approach because a decision cannot be made only based on a point that received most ratings and ignoring the ratings on remaining points. For example, if we look at the occurrence rating of problem P20 in the figure below, it may appear the ideal rating would be 4-medium as it received 10 responses, which is the highest amongst the 10 points, but a rating of 4 does not seem to be appropriate when seen from a bigger picture, i.e., point 7 and 9 received a total of 8 and 9 ratings respectively which means that P20 belongs to a higher rating than 4-medium on a 10-point scale.

This led to a change in approach for survey analysis and weighted average method was followed as this method factored into account the responses received from all survey participants rather than only those responses from participants that rated a point on a 10-point scale that received the greatest number of ratings. (Pimentel, 2019) defines weighted average as a weight multiplied to the frequency divided by the total frequency to obtain the weighted average. (Stevens, 2012) in their article defines the formula for calculating the weighted average as $(\text{Number of votes} * \text{Weighting for column 1}) + (\text{Number of votes} * \text{Weighting for column 2}) + \text{Number of votes} * \text{Weighting for Column 3}) + (\text{Number of votes} * \text{Weighting for column 4}) + (\text{Number of votes} * \text{Weighting for column 5}) / \text{Total Number of Votes}$. So, using this formula we can calculate the weighted overage for occurrence factor as $((5*1)+(5*2)+(5*3)+(10*4)+(5*6)+(2*6)+(2*7)+(7*8)+(9*9)+(10*2))/60 = 4.71$. Hence an ideal rating for this problem should be 5 instead of 4 based on the weighted average of responses.

Lack of different skillsets, appropriate tools or restricted access to tools hampers the team to be fully cross-functional and makes the team heavily reliant on component teams in possession of necessary skills or tools



Responses from survey participants that does not meet the research criteria: Globally distributed projects, Multi-Vendor Setup, and Agile Scrum methodology were excluded from the response data and then the weighted mean approach was applied to determine problems that has greater impact. Results of this analysis can be found in the table below and in detail in the spreadsheet attached under Appendix B. Cumulative weighted Score is calculated using the formula Occurrence Weighted

Score*Impact Weighted Score*Time Weighted Score and cumulative on a 1-10 scale calculated using the formula Cumulative Weighted Score/30. Table below is the output of the survey analysis where the 8 problems highlighted in yellow denote those that have cumulative weighted average of above 6, which falls under the high-risk category on Likert scale. So, the 8 problems that we identified as significant problems in globally distributed multi-vendor agile scrum team are P20, P10, P16, P23, P19, P18, P2 and P11.

P.No.	Problem	Occurrence Weighted Score	Impact Weighted Score	Time Weighted Score	Cumulative score	Cumulative on a 1-10 scale
P20	Lack of different skillsets, appropriate tools or restricted access to tools hampers the team to be fully cross-functional and makes the team heavily reliant on component teams in possession of necessary skills or tools	6.153846154	6.025641026	5.897435897	218.6820412	7.289401372
P10	Frequent changes in requirements or user story especially towards the end of a sprint causes excessive delays, re-planning and rework	5.794871795	6.307692308	5.794871795	211.8157083	7.06052361
P16	Constantly changing resources or adding new resources in the project affects team's chemistry and velocity	5.820512821	6.102564103	5.820512821	206.7449215	6.891497384
P23	Lack of project management competence or appropriate leadership style for agile projects by project manager or scrum master, i.e., project manager adopts an autocratic leadership characterized by individual control over all decisions and little input from group members	5.641025641	6.076923077	5.717948718	196.0117332	6.533724439
P19	Lack of continuous Client/Business User involvement due to distributed nature of the project, such as communicating to business and getting feedback on matters	5.538461538	6.179487179	5.666666667	193.9408284	6.46469428
P18	Inability to breakdown backlog into smaller user stories and prioritise based on most business value that can be delivered within a sprint 'or' horizontal breakdown of stories based on tasks such as design, coding, testing, etc., instead of product features often leads to work spanning over multiple sprints and not delivering working software or business value at the end of each sprint	5.820512821	5.820512821	5.666666667	191.9774271	6.399247571
P2	Problems such as changing priorities, changing scope and conflicts with resource availability by teams within the project and interfacing teams outside project prevents tasks from being completed within sprint timelines	5.58974359	5.974358974	5.512820513	184.101384	6.136712801
P11	People often allocated to multiple projects/teams on shared basis results in discontent and stress personally and interruption to team members seeking input on overlapping urgent issues	5.794871795	5.871794872	5.333333333	181.4735919	6.049119731
P12	Continuous interactions from peers onsite or offsite, clients and vendor teams seeking update on project progress, help or information on work processes, interrupts the primary tasks assigned and eventually leads to working over time to meet the sprint goals	5.897435897	5.512820513	5.384615385	175.0619532	5.835398439
P5	Joint estimation/planning in agile scrum teams are often influenced by other vendors, management and clients resulting in user stories estimated lower than it would be required results in cost and schedule overrun	5.41025641	5.564102564	5.743589744	172.9005546	5.763351821
P15	Absence of hygiene factors at work such as lack of advancement, growth, recognition of achievements by management or client leads to demotivation and reduced productivity	5.666666667	5.41025641	5.615384615	172.1571335	5.738571115
P13	High frequency and excessive time spent on meetings and agile ceremonies such as daily stand-ups, retrospectives, sprint reviews etc reduces the time available for development tasks and eventually leads to working over time to compensate the time lost	5.769230769	5.487179487	5.435897436	172.0831437	5.736104789
P9	Online meetings are not as effective as compared to face-face meetings due to distractions from parallel working and communication, considerable time and difficulty to explain project matters and opinions, wait for approval/feedbacks, lack of non-verbal cues	5.58974359	5.615384615	5.435897436	170.6249937	5.687499789
P4	Estimation and Planning are mostly performed by co-located teams at client site, usually by experienced team member representing each functional area - build, test etc., and tasks accepted by less experienced or remote team members leads to slippage of schedule	5.461538462	5.717948718	5.461538462	170.5572751	5.685242502
P22	Lack of Leadership/management support and commitment at different levels of management - client, vendor etc.	5.487179487	5.692307692	5.384615385	168.1869216	5.606230719
P14	Sense of differentiation towards members from other vendors, competition, power struggles and politics by vendor teams to influence clients preference to their side often leads to conflicts, lack of trust and collaboration in the project	5.538461538	5.615384615	5.333333333	165.8698225	5.528994083
P25	Lack of governance and documentation leads to problems with measuring and monitoring the performance, knowledge deficiency etc.	5.512820513	5.564102564	5.384615385	165.1671471	5.505571571
P1	Lack of clear boundaries defining scope and tasks often lead to one vendor stepping into other vendors domain	5.358974359	5.512820513	5.205128205	153.7754345	5.125847817
P8	Lack of proper communication tools, less overlapping time due to different time zones, different cultural and language background are obstacles for remote teams to effectively communicate in real time or participate in meetings with counterparts from different locations	5.435897436	5.435897436	5.179487179	153.0485679	5.10161893
P21	Lack of clarity around roles or specialised roles such as testing manager, build master, project manager etc., can sub-optimize efficiency, control and smooth team functioning, especially in multi-vendor agile setup where resources directly report to a client manager	5.282051282	5.512820513	5.205128205	151.5681316	5.052271054
P17	Concerns about sharing tools, knowledge or proprietary information such as development tools, stubs etc., created by one vendor with another	5.41025641	5.435897436	5.025641026	147.802087	4.926736234
P6	Lack of direct communication between vendor teams and communication often facilitated through clients results in awareness and co-ordination problems	5.230769231	5.461538462	5.128205128	146.5028069	4.883426895
P3	Difficulties in translating agile process, methods, values and metrics to non-agile or traditionally organised management teams, such as while seeking support for client-vendor, vendor-vendor conflict resolution; status reporting to a non agile stakeholder etc.	5.205128205	5.333333333	5.102564103	141.6506684	4.721688947
P7	Sheer size of the team, i.e., more number of communication links within the team leads to deterioration in communication quality or information exchanged	5.205128205	5.256410256	5.051282051	138.2045382	4.606817939
P24	Resistance/lack of acceptance of agile principles or wrong understanding/incorrect application of agile approaches, such as, management orders to work agile but the team resists to change, rapid transition to agile ways of working with less time to foster benefits of change/adequate training, or the team does not see a need for agile as the system developed is not modular or suitable for development in iterations	5.102564103	5.307692308	5.076923077	137.4974966	4.583249886

Interview Agile Scrum Practitioners working in Multi-Vendor Distributed Projects

Interview was conducted separately with 5 agile practitioners working in different multi-vendor distributed projects in roles such as Vendor Manager, Project Manager, Business Analyst, Tester, Infrastructure Support to understand their experience on causes, impact, prevention measures for the 8 problems and the resulting transcript was used to study the inter-relationship between the problems and their impact on project performance.

Interview Agenda

Problems in Multi-Vendor Distributed Agile Scrum Teams

P.No.Problem

- P20 Lack of different skillsets, appropriate tools or restricted access to tools hampers the team to be fully cross-functional and makes the team heavily reliant on component teams in possession of necessary skills or tools
- P10 Frequent changes in requirements or user story especially towards the end of a sprint causes excessive delays, re-planning and rework
- P16 Constantly changing resources or adding new resources in the project affects team's chemistry and velocity
- P23 Lack of project management competence or appropriate leadership style for agile projects by project manager or scrum master, i.e., project manager adopts an autocratic leadership characterized by individual control over all decisions and little input from group members
- P19 Lack of continuous Client/Business User involvement due to distributed nature of the project, such as communicating to business and getting feedback on matters
- P18 Inability to breakdown backlog into smaller user stories and prioritise based on most business value that can be delivered within a sprint 'or' horizontal breakdown of stories based on tasks such as design, coding, testing, etc., instead of product features often leads to work spanning over multiple sprints and not delivering working software or business value at the end of each sprint
- P2 Problems such as changing priorities, changing scope and conflicts with resource availability by teams within the project and interfacing teams outside project prevents tasks from being completed within sprint timelines
- P11 People often allocated to multiple projects/teams on shared basis results in discontent and stress personally and interruption to team members seeking input on overlapping urgent issues

Questions

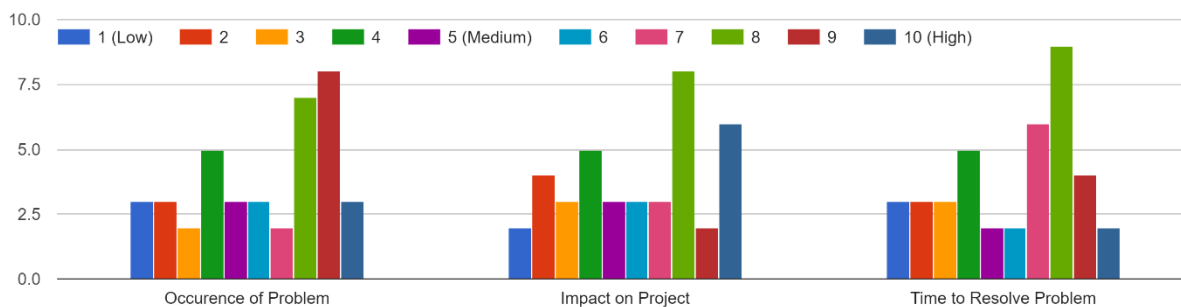
1. Why do you think this is a problem in a multi-vendor project with team members distributed globally?
2. What do you think is/are the causes for this problem?
3. How does this problem impact your project?
4. What can be done to prevent this problem from occurring?

Research Analysis and Findings

Lack of different skillsets, appropriate tools or restricted access to tools hampers the team to be fully cross-functional and makes the team heavily reliant on component teams in possession of necessary skills or tools

Based on the survey analysis, P20 has topped the list of problems with a cumulative weighted score of 7.2 (see table).

Lack of different skillsets, appropriate tools or restricted access to tools hampers the team to be fully cross-functional and makes the team heavily reliant on component teams in possession of necessary skills or tools



Distributed nature of the project causes a huge difference in skillsets between onshore and offshore counterparts because of the access to certain tools restricted to use only at client location, due to reason such as security policies, cost etc. This means the offshore team has less or no knowledge of certain tools that can enable them to understand requirements better and cannot perform certain activities as compared to their onsite teams. This can also be supported by the fact that 13 of 23 survey respondents located offsite have rated this problem as high (above 6 on Likert scale) and Interviewee 1 located offsite mentions the following:

"I've faced this a lot because we have been working from a different geographical location and usually what happens is onshore people have the special privileges than the ones at the offshore. For example, they're (onsite) somebody fully equipped with to do. But because of certain access or certain privileges, we (offshore) couldn't do the task. And that may hamper the entire, you know, the timeline of the project or maybe impact the project overall. So even certain skill sets, for example, certain trainings are being conducted onshore but not to the offshore people, sometimes with the special training. So that may also mean, there could be skill set gaps between the people at onshore and offshore. For example, the people at onshore, they are at the client location and work with clients so closely and sometimes get to see things, like for example, ATM Labs or whatever we use in our testing, they work on the tools that they get to see things so it makes them understanding requirements better. It also helps them to gain more skills knowledge into that particular area than people who are in different location" "even in current product I'm integration testing, vendor side, we are not able to access some of the documents which they have created by their own. So that is creating a problem like we are not able to fully understand end to end because there is access problem. And when we are raising, it takes more than a week sometimes to get those accesses. I think reason may be security, and sometimes it can involve heavy cost when they want to give access to tools for many people. So, they just choose few and just give those people accesses"

Another main factor that most interviewees mentioned as contributing to this problem is the concerns about disclosing proprietary information or assets leading to vendors restricting access to tools and knowledge documents, which (Gogan & Rao, 2008) also sees as major issue unique to outsourced projects with multi vendors where they push hard to protect intellectual property and other strategically-sensitive data from other vendors for competitive advantage adding complexity to projects and tension between competition and cooperation amongst vendors. This can be supported by the fact that most respondents who identified themselves as vendors rated the problem P17 (concerns sharing information or assets) on a higher scale (13 of 29 respondents greater than 6 point) compared to those from client organisations who rated the problem as low (6 out of 10 below 3 point). Interviewee 5 states the following as a reason why vendor organisations restrict access to their tools and information:

"So, it is just because we know, like in any project, there are multiple systems, multiple application, and for each application, there might be different vendors and no vendor wants to disclose what exactly they have in their own application. And why a vendor doesn't want to disclose what exactly they are doing inside the tool, for example, if I take an example of my organisation ABC. There was a product called payment processing engine, which is a payment platform developed by a Vendor X. So, in that case, X doesn't want reveal exactly how they have built it, what kind of services they have integrated and, what all the different services like, they don't want to expose the infra-structure they have created just to safeguard their own expertise so that the client or the company should not throw them away at any point in time, and to restrict their knowledge within their premises itself. And that is the reason they didn't want to disclose everything to the external world. Whatever the information they think is important they don't want to disclose or expose in front of any third-party person, personnel or system or any company they will just restrict it within their own premises doesn't want anybody to intrude inside that."

Segregation of duties or following a RACI (Responsible-Accountable-Consulted-Informed) model to manage multiple vendors is another factor highlighted by the Interviewees as contributing to this problem P20. RACI is a general framework used for third party management to push out responsibilities across different organisations where R represents the organisation responsible for completing a task or making decisions, A represents organisation that is accountable or owns a task, C and I usually representing stakeholder group that needs to be consulted or informed for any decisions made (Rogers, 2019). It is evident through our interview that there are atleast two factors that leads to access restrictions on tools and knowledge that impacts an agile team from being cross-functional and the flexibility towards getting things done quicker – one is segregation of duties or RACI which assigns each vendor or teams as responsible or accountable for certain tasks, and at times, this adds complexity within a project. An interviewee highlights that a simple database update or a file transfer that would not require a special skill or significant time by a developer, usually had to wait few hours

for a component team to action their requests because of the access restrictions imposed by them in the development environments as opposed to only in the production environments, which added an extra layer of dependency and failed co-ordinations often leading to project delays, and the other factor is that team lacking knowledge or skills to use a tool even if they have access to required tools, which again creates dependency on external teams with the required knowledge.

"I would like to add something called a RACI, a kind of duties of segregation between teams. So, let's say there is a developer. He cannot migrate changes to the production and it is something developer can do and may take only half an hour for him. But there is a dependency for him on environment services guy to do that. They also do the same what developer will do. But you know, there is a dependency there because a boundary has been defined. The developer is not allowed to touch anything in the higher-level environment. But you know, that's how the industry works. Let's say you're processing some payments files and sending those files out of your organization, maybe need to use some tool or kind of a gateway, I can say sterling file gateway, which is if these tools are in place and within your organization, but you don't have the right skill set to use it or a right tool, which allows you to connect to that particular data system will definitely hampers your activities. That becomes a kind of dependency where we need to onboard that specific engineer to do the work. Even though it's a small file transfer from one server to another server, if we don't have the right tools, we won't be able to finish the job. Project will get delayed as it becomes a dependency and timelines will change."

"Yes, as you know we have worked on a multi-vendor team structure for example if we have a different team who provide us DBA, they are the only one who has access on different tables both in test and production environment ok so there we definitely have a dependency and if we don't coordinate well and we can always have some kind of trouble in getting the things done in a smooth fashion and it can eventually impact your overall project timeline"

(Gandhi, 2020) mentions in his article that RACI predominantly used in traditional models to define who owns or controls a specific piece of work, so an individual or organisation takes credit of a success or blamed in the event of a failure, has always been overlooked at agile project levels as the responsibility of getting the work done resides on whole team rather than on any individuals/organisations that often leads to voids in collaboration, team cohesion and information exchange in larger projects with multiple organisations working together. We also insist that removing RACI or segregation of duties as not an ideal solution for this problem as it would lead altogether to a different problem, P1, which (Akshanthula, et al., 2019) also in their article mentions not demarcating the scope and activities to be performed by each team member can result in one vendor team stepping into another vendor team's domain. An ideal solution for this would be establishing governance at Agile team levels, (Draper, 2017) defines governance as a process of setting directions and making decisions as to who (people/organisations) does what and how, i.e., the process by which they intent to achieve it. Although organisations with multiple vendors would have outlined in each of the vendor contracts - a RACI that defines the responsibility of each vendor and a dependency matrix that illustrates the dependency between each vendor. However, these governance factors outlined in a vendor contract would usually be at the organisational level and may not be effective at agile team level leading to complexities in terms of processes, access restrictions, and additional layers of dependency, such as a simple database update responsibility that we discussed earlier. Therefore, one possible solution we identify for this problem is to follow effective governance process within the agile teams such as refining the RACI and other policies through identifying day-to-day behaviours that prevents the team from switching to agile practices and ways of thinking and to improve on through continuously shaping the roles and responsibilities that promotes flexibility. (Talby & Dubinsky, 2009) highlights in their article that an effective governance should also focus on the internal governance for a specific agile team by dealing with their current and relevant problems aimed at short term improvements, usually within few sprints, in traits such as flexibility and appropriateness. From this it

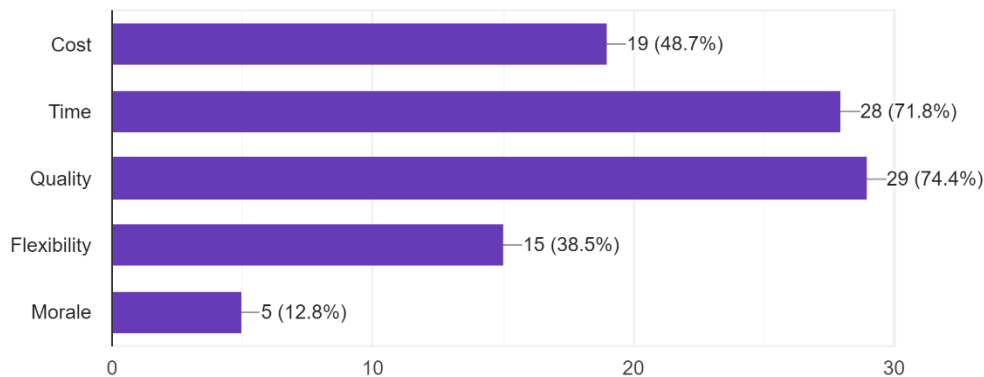
can also be concluded that problem P25 lack of governance in agile projects with a weighted score of 5.5 (medium) may as well be influencing this problem. An interviewee highlights the importance of governance to solve this problem as follows:

So, whenever any company, any project, have multiple vendors set up in the similar way what we discussed. They have a dependency on X for processing the payments, dependency on B to process their card payments, a dependency on C to handle their core banking system. So, while engaging such third-party vendors, they can speak of in their official forum that whatever you do or whatever the product that you customize as part of my project for which I'm paying, it should not be restricted and it should be exposed to everyone. This can be done only when they are setting up any board, meeting, any management meeting where they are deciding what all the vendors will be coming into the picture to have our entire infrastructure ready. So, if we take an example of three vendors above, I'll make sure on very first day I being a CFO, CXO or CTO or whatever I could be that all these three vendors have enough understanding to understand each other. So that they should not hide anything from each other or anyone. And they should have their documentation in a way which should be all stored on the SharePoint so that every other vendor, every third-party vendor can read through it and whatever the tools are necessary to understand each other's system should be open for everyone to access.

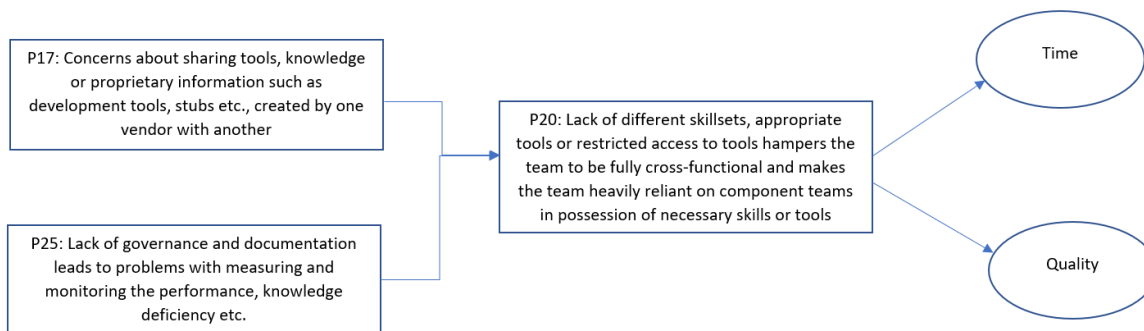
All of the interviewees highlighted 'Time' as the main project performance factor impacted by this problem P20 due to the added complexities and layers of dependency in the process and with some mentioning 'Quality' as another factor impacted. Responses from the survey also highlights Time and Quality as the most rated factors to be impacted.

Which of these Project Performance factors are impacted due to the previously rated problem?

39 responses



Relativity between the problems: based on our analysis, it can be concluded that P17 and P25 are other problems that can have some influence on agile teams leading to problem P20, ultimately impacting Project Performance factors - Time and Quality

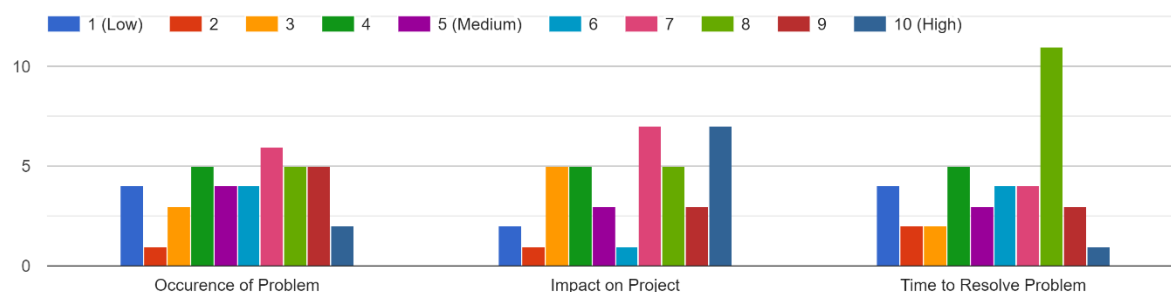


Frequent changes in requirements or user story

P10: especially towards the end of a sprint causes excessive delays, re-planning and rework

Based on the survey analysis, P10 has been ranked second with a cumulative weighted score of 7.0 (see table)

Frequent changes in requirements or user story especially towards the end of a sprint causes excessive delays, re-planning and rework



Most of our interviewees perceived this as not a significant problem when seen from the perspective of requirement changes initiated by the business needs itself. In most of the cases, projects were faced with minor business requirement changes that they were able to accommodate within the current sprint through replanning of the tasks with minimum re-work or waste, and in few occasions, if the change to business need is major, it was either added to the backlog of tasks to be delivered in successive sprints or, in worst cases, leading to the work already performed in current sprint being scrapped and replanning to accommodate the new requirements.

We do see, because in real world, you know, customers or business will never satisfy or while the requirement gathering and all, they'll definitely miss some of the points which are definitely required for entire project delivery. So, we'll see if it can be accommodated within the current sprint. If it can be accommodated like I said, it's a small change, then we'll onboard it. If not, then we'll keep it as a backlog for the upcoming sprint. Hmm. You know, if at all the total functionality is getting impacted and we have no other option other than scrapping the current Sprint and Doing a re-plan. But most of most of the 80 percent cases, what we see is like minor requirement changes. So those will normally accommodate.

However, most interviewees highlighted the fact that changes in business requirements, user stories or tasks planned in the middle of a sprint is still a commonly occurring problem in less mature agile projects, which are new to the way of planning work in an agile way, i.e., when they switch to delivering requirements in an iterative manner as opposed to the big bang approach of defining requirements upfront, which often led to frequent replanning, rework and projects not delivered to the schedule. This view can also be supported by the fact that 9 out of 13 survey respondents with less than 3 years of experience working in agile teams rated high (above 6) for problem P18, which is the inability of team to breakdown work into smaller user stories or prioritisation based on most business value causes the team to revisit sprint goals when any deviation from delivering the intended project

requirements is foreseen and necessary changes are made to user stories already in sprint execution to align with the project requirements.

Yeah, it happens especially in not matured agile teams. It takes one or two years to get used to agile way of working. It happens in the beginning and later when the process is streamlined, product owners, business design and the team follow the process in a disciplined manner. Yes, in our project we faced it initially we started agile in 2018, and in 2019 and 2020 we faced this problem and later it gradually disappeared. Entire team, even not only partner and even client has to make the agile mindset and if everyone is new to agile this problem happens. Definitely this will cause delays in product launch because of change in requirement and we have to go back and redevelop and retest it so anyways its cost and time that will be impacted

This agrees to a certain extent, i.e., for newly formed agile teams, with (Wiesche, 2021) opinion that frequent fundamental changes to requirements by customers is the most common problem in agile projects that often leads to rework or a developing a new code altogether. Our interviews also uncovered two factors as contributing to this problem, which (Wiesche, 2021) also mentions in their article as – development-imposed impediments such as changing scope/requirements or missing information from another team resulting in delays until the impediment is resolved, and interaction-imposed impediment such as lack of effective communication and collaboration between vendor teams. Interviewee 1 mentions that there is a high level of development related inter-dependencies between vendor teams, and changes to their requirements by any of these teams, which at times not communicated to dependent teams at the right time, forced the dependent teams to make adjustments in their sprint goals to accommodate the unprecedented changes from other teams, which often led to significant impact on delivery timelines, quality of deliverables and morale of the team. We identified through our interview that the main reason for this is lack of effective collaboration and communication between inter-dependent teams due to their distributed nature such as different geographical location, work culture, work policies and lack of face-to-face communication leading to delays in communicating requirement changes and misunderstanding of requirements. From this, we can agree that problems with effective communication such as P8 and P9 also has some influence on agile teams leading to P10.

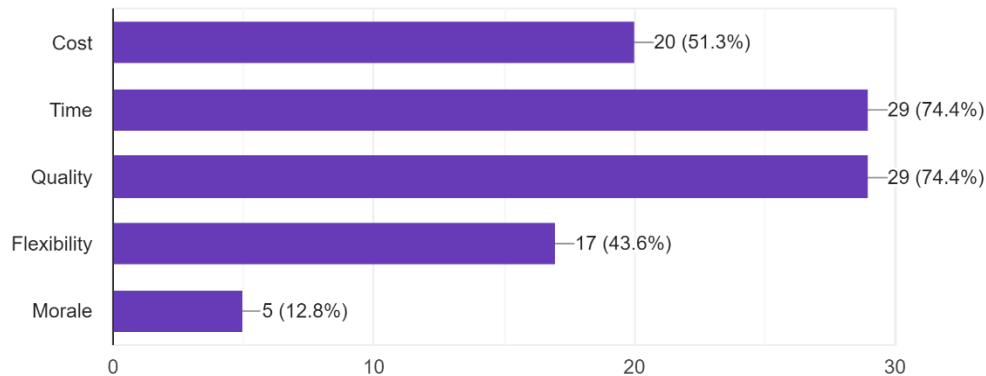
“Basically, when you're working with multi-vendors, so everybody is creating some sort of requirements. For example, in my current project, what is happening, the mobile app team, which creates apps and mobile servers. They have their own requirements, in the integration environment. It becomes very complex. Then, we have tandem side changes that are different than other. But these changes, when they are not synchronised properly, they really hamper the progress of the overall project. And what is happening is, requirement is changing frequently and sometimes lack of access is also there. For example, we have taken requirement documents from them, right? And after some days I mean, when planning they have updated the document, but we are not aware of that paperwork requirement. These are the requirements which are happening at the vendor side. We are not, you know, informed at the right time. And that keeps on going. It should be like whenever there is a change, it should be informed to other teams which are working in integration with the vendors so that there is no gap between them. So, for example, if we got two or three weeks to work on that and, if something gets changed in between or, within one week, it is very difficult to give the quality of that work right. So, quality get compromise in that case. Our mental health also, because when you're working in extreme stress, so it hampers your physical health as well, sometimes and definitely morale goes down when you are not able to when you are trying hard to deliver that piece of, you know, code or whatever work you're doing. I think collaboration there should be extreme collaboration in that case because everybody is working from home, I see there are multiple challenges when everybody is working from home. But still, I think there could be a better way to collaborate and even from the different geographical locations, so the work culture is different, the way of doing things is different. So that makes I mean, I mean, that makes things a little difficult, difficult to collaborate. But when people are at one place, you connect more with people. So, it becomes quite easy to collaborate with other people to understand their requirements in a lesser time. But when you're working remotely, you know, it becomes difficult.”

“So basically, that it might be a possibility that they were expecting something else and the technical team might have understood the requirement in a different fashion ok it's not always change in requirement I would not say it might be a refinement in requirement too which will still be part of the project delivery but then it might take some time of the team effort because they have built something which was incorrect and that needs to be refactored into the upcoming sprints”

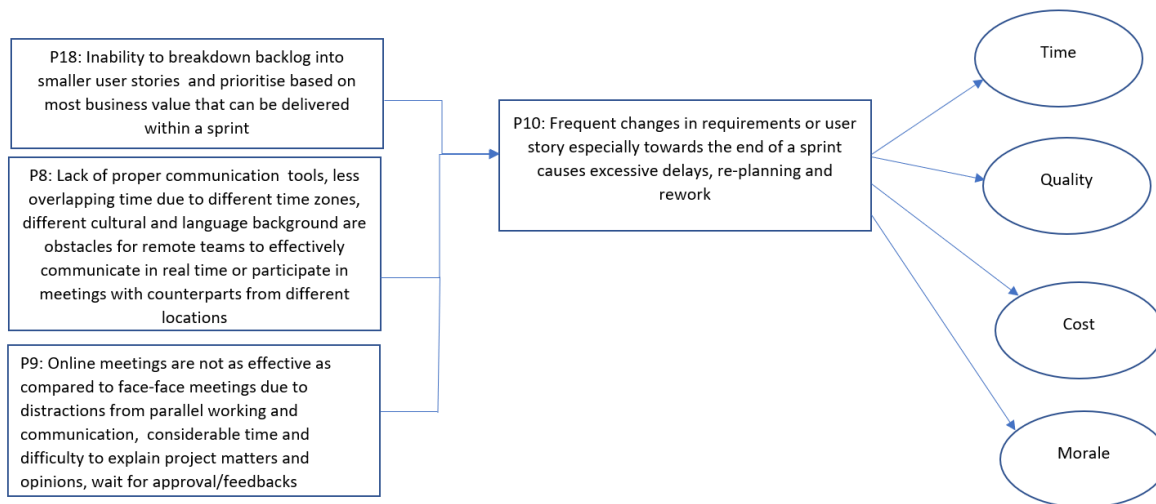
Almost all the interviewees mentioned both ‘Time’, ‘Cost’ and ‘Quality’ as the main project performance factor impacted by problem P10 and atleast three interviewees mentioned ‘Morale’ to be impacted as a result of the wastage of efforts on the work being scrapped and rework. Responses from the survey also highlights atleast 50% rated Time, Quality and Cost as impacted.

Which of these Project Performance factors are impacted due to the previously rated problem?

39 responses



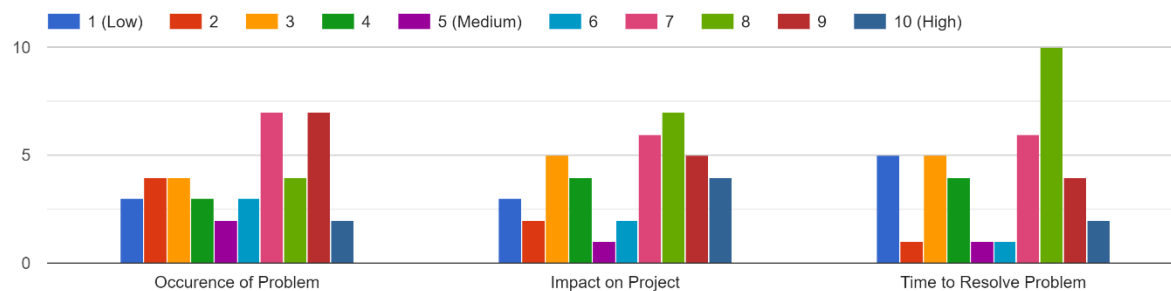
Relativity between the problems: based on our analysis, it can be concluded that P18, P8 and P9 are other problems that can have some influence on agile teams leading to problem P10, ultimately impacting Project Performance factors – Time, Quality, Cost and Morale



P16: Constantly changing resources or adding new resources in the project affects team's chemistry and velocity

Based on the survey analysis, P16 is ranked the third critical problem in agile scrum teams with a cumulative weighted score of 6.8 (see table)

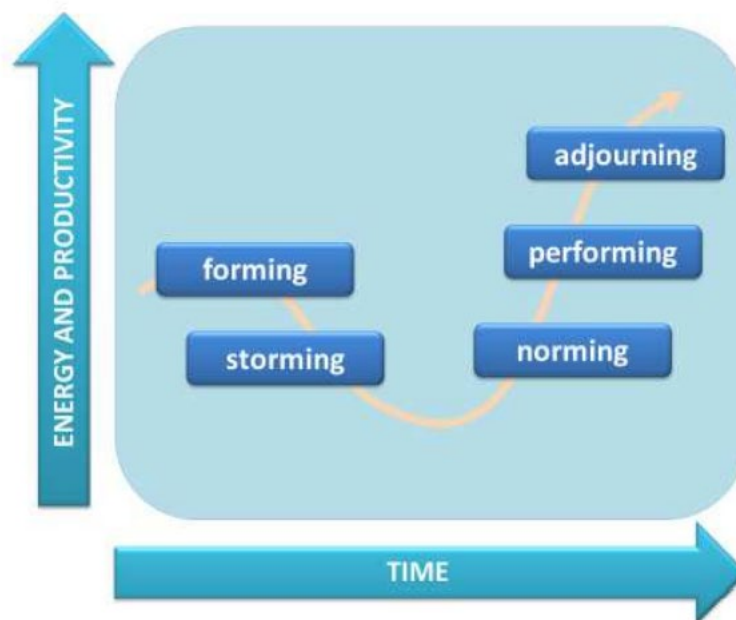
Constantly changing resources or adding new resources in the project affects team's chemistry and velocity



Constantly changing resources is a common problem in any organisation or project regardless of the project management method adopted – agile or non-agile. However, this problem was identified during literature review to study its impact on multi-vendor agile teams due to two reasons –employee turnover has been a challenging problem even at individual organisation levels and agile manifestos emphasises on maintaining a stable team structure in one of its principles “At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behaviour accordingly”. (Hazrati, 2009) suggests the success of high performing agile teams is to maintain the team intact which was formed through a prolonged period of forming, storming, norming and performing in a single team. But our interview with agile practitioners reveals that this is normally not the case in most agile teams. Most interviewees mentioned that changes in agile team structure occurs mostly as a result of employee attrition, i.e., when an employee voluntarily leaves the organisation when their needs are not met or, in some cases due to conflicts or discontent with the management or leadership styles. From this, it is evident that the trigger to this problem is often always due to the problems associated with ‘people’ dimension of critical success factor – P15 which is Absence of hygiene factors at work such as lack of advancement, growth, recognition of achievements by management or client leads to demotivation and reduced productivity and P22 which is lack of Leadership/management support at different levels of management.

“It happens when there is high attrition like people leave when market is open and people are getting good salaries every industry face this challenge. There could be several reasons. There is a slogan right, what I heard is or what we heard every time is people don't leave their organization, they leave their managers. So, there are frequent chances, like when manager is bothering any of the resources or manager is not good or , when the employee's needs are not getting satisfied. They think of leaving the project, leaving the organization. So, if you can mitigate this by understanding what exactly the employee or the resources needs during the course of the project and how you can make them happy so that there will be less chances that they will leave the project. So, manager can play an important role here if you want to really stop the attrition from your project. You need to be the project manager. You need to be good with the resources”

Although the factors that lead to this problem may be common in both agile and non-agile projects, but the impact can be more profound in agile projects as compared to non-agile projects because of the nimble and collaborative nature of agile projects. (Shrivastava & Rathod, 2015) recognises team continuity as one of the contributing factors to agile project performance as they believe it is not just people's technical skills that determines it but their relationships and ability to accomplish tasks together. Our finding from the interview agrees with (Hazrati, 2009) views on agile teams that it take a prolonged period of forming, storming, norming and performing to evolve into a stable team that can perform to the best of its combined abilities. When new members or added into an agile team, the team goes through a learning curve as described in Tuckman's stages of group development (Wilson, 2017) – 'forming' is the initial stage when a new member onboarded meets the team and they begin to learn about opportunities and challenges in their new role such as differences in organisational culture from previous workplace and knowledge regardless of their experience etc. Then they go through the next stage of 'storming' where the new members with the support of experienced team members learn about individual working styles and to work with each other as a team. Storming is the phase where the team's productivity and velocity i.e., the time taken to accomplish tasks will be at their lowest as the new member is still at their learning phase and the experienced team channels some of their efforts away from deliverables, in order to support and development of new members. 'Norming' is the phase where the new member gets accustomed with the team and able to take responsibilities, and work for the success of team's common goal and this is also the phase where productivity and velocity tends to increase towards the team's original level.



Phases in learning curve of newly formed teams and impact on project performance – figure adopted from (Wilson, 2017)

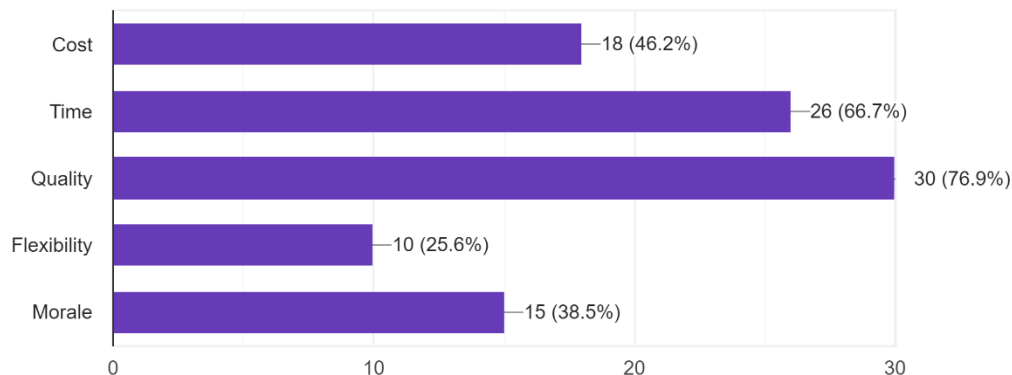
"Usually, it happens because the people move in and out of the organisation, they choose to work with a different organization and to compensate that, you, put some resources in place of that. And when that comes, it becomes difficult for the team to adjust for some time because you have to make that new candidate aware of things. It might be possible that person who has been, you know, hired on that is not from that background, then that becomes a bigger problem. So, it becomes very difficult to adjust for a few months, even if you have hired some somebody who is experienced but with no domain knowledge. So, the new person will definitely take time and it is understandable to everybody. But at the same time, pressure builds on the person who is already working so hard and supporting the new person and doing the project. So, yeah, that impacts the velocity, definitely, because one person has to do all the things, delays and the experienced person might get mentally stressed doing more than what he is supposed to do"

"The new person is also welcomed the way existing team members and we make him feel part of the same team so that will not be an impact but yes the velocity sometimes decreases because it takes some learning curve for the new person to grab the knowledge on the application. When you come out from a different organisation there are always challenges and you have to learn the new processes and the way team works"

From this, we are able to conclude that constantly changing resources in agile teams can reduce the quality and velocity of team for a few months when they go through the learning curve from 'forming' to 'storming', and also impact the morale of existing team members because of the delays in their deliverables and additional effort provided to support the new resources. Responses from the survey also highlights Time and Quality as the most rated and 'Morale' as well has received significant rating as compared with other problems.

Which of these Project Performance factors are impacted due to the previously rated problem?

39 responses

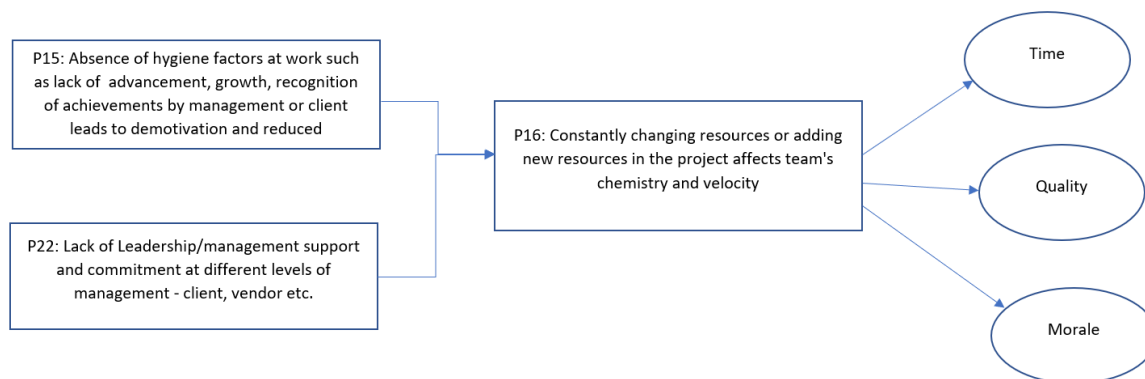


A survey by (Deloitte, 2021) highlights that employee attrition increases when an organisation fails to fulfil employee needs from inclusion to reskilling. (Schaufeli & Bakker, 2004) also mentions that people who are provided with better job resources i.e., physical, psychological, social aspects of the job, develops high levels of motivation – intrinsic (psychologically) and extrinsic (monetarily), and have a tendency to stay longer with their organisation. Maslow's theory states that human behaviour that develops motivation towards their job is influenced by biological, cultural, and situational factors that can be arranged in a hierarchical order, lower to higher from psychological, safety, belongingness, esteem, and self-actualisation needs and the management must begin fulfilling the lowest needs of an individual first before progressing to higher levels in order for it to become a motivator (Kennedy & Daiml, 2010). So, an ideal solution for organisations to prevent this problem is to develop participative leaders or managers who listens, identifies and supports fulfilling employee needs, and also constantly encourages and recognises teams and individual on their achievements, that will resolve the problems

P22 and P15 which we discussed earlier as contributing to this problem. A project manager interviewed mentions how fulfilling self-actualisation needs of an employee helps retaining them:

“So, the best part for this is basically we always connect to the team both on the professional as well as personal front ok we try to keep asking their aspirations and the way they are looking their career to be shaped ok and if there is anything that we can help to mentor them or take them to next level we always try to do that by providing them early or regular feedbacks. we try to ask them if they can do some trainings or certifications that can boost their career ok so by doing all these things and at the end of day, we always keep saying to the team we work as a team and if there any problem they have to come up and they have to ask if there is a solution to that which is not currently with you but may be with the other team members so we try to help each other”

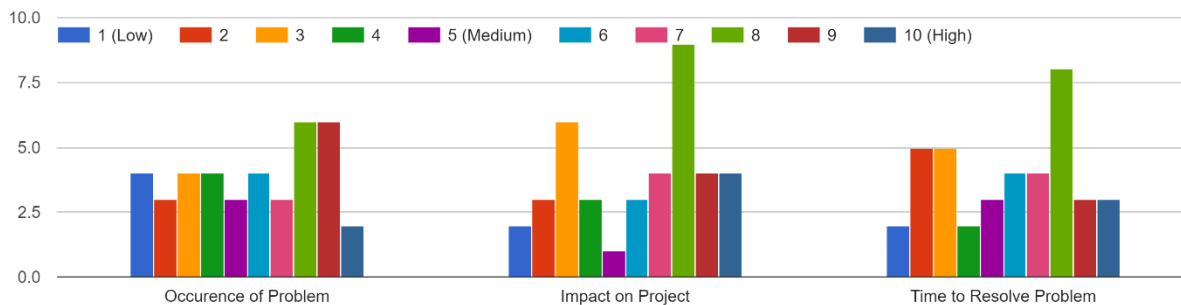
Relativity between the problems: based on our analysis, it can be concluded that P15 and P22 are other problems that can have some influence on agile teams leading to problem P16, ultimately impacting Project Performance factors – Time, Quality and Morale



Lack of project management competence or appropriate leadership style for agile projects by project manager or P23: scrum master, i.e., project manager adopts an autocratic leadership characterized by individual control over all decisions and little input from group members

Based on the survey analysis, P23 is ranked as fourth critical problem in agile scrum teams with a cumulative weighted score of 6.5 (see table)

Lack of project management competence or appropriate leadership style for agile projects by project manager or scrum master, i.e., project manager adopts an autocratic leadership style over all decisions and little input from group members



It is not uncommon in agile scrum teams comprised of multiple vendors to have resources from one vendor organisation report to managers or leadership from other vendor organisations. Hence, this problem was identified with an intent to study the inter-organisational resource and managerial relationships and the impact it has on agile scrum teams. Two interviewees highlighted that such inter-organisational managerial relationships are sometimes influenced by differences in leadership styles adopted by other organisation that they find difficult to adopt, discontent supporting or working together, power struggles, politics, blame-games and biasing towards other vendor resources to influence the preference of client towards their side. This agrees with cases we reviewed during literature study such as sense of differentiation towards a member of another vendor leads to lack of cooperation and conflicts (P.Hui & Beath, 2001); competition over insufficient demand, vagueness over task responsibility or ownership, negative affinity towards other team members (Panteli & Sockalingam, 2005); Power struggles, politics and finger pointing by vendors to influence clients preference to their side (Banerjee, 2012), which can all ultimately lead to de-motivation, reduced productivity, increased defect rates, and lack of trust and rapport between vendor teams. Through this, we can agree that sense of differentiation (P14) or biasing exhibited by project managers towards certain resources within a team can prevent them from demonstrating flexible, adaptive and partnering approach required to pull together a team that functions as one team.

"I would say like whenever, whenever there is a leadership from different organisations, they have different styles of working. And when you're when you're working with different, different organisations, you have to manage with their leadership style and how they want their employees to work. And sometimes it creates like whenever you don't understand leadership behaviour or somebody from different organisation or, you know, vendors, you get a feeling of little. I mean, there could be a little anguish between people when they work together. Also, like some people display some sort of politics and you know, take advantage of or they may harm you for something from you. They may, you know, refrain from working together, getting promoted"

"Yes, we see all those types because you know this business is a dog eats dog business. Definitely everyone looks for an opportunity let's say if a deployment fails in test. For a test manager it looks like a day is wasted, so how he counts is I have 5 resources and one day is wasted for 5 resource so I will say that 5 days wasted for me but as an infrastructure guy I will say it's one working day its 8 hours for me I will still count it as impact but as 8 hours impact but you know as you are in a position you can control so you report it as 5 days so you will say that let me put my resource to control this and all these politics we see day to today"

One interviewee highlights that project managers because of lack of trust on their team exercises too much control and micromanagement over the team's activities that inhibits knowledge sharing, development and decision-making ability of the team, which are integral to effective functioning of

agile teams. (Panteli & Sockalingam, 2005) mentions trust as an immanent factor in any inter-organizational partnership that boosts collaboration, share expertise and skills for mutual benefit.

One thing is if you have too much control on someone definitely you are not letting them grow right. For example, person X is reporting to person Y, person Y says you only do these three things - restart the server or copy files from here to there that's it and if some complex job comes up and he gives it to some other person then how this new person X will learn and his capabilities comes out. You should definitely give a free hand to everyone even though he is new to job or IT industry, definitely we should trust them and allow to do complex stuff and learn so that a team can operate better and better not to do micromanagement so if you give task to some person trust them and you can review task later once, he delivers and you cannot come to his desk and look at what he is doing and all you know.

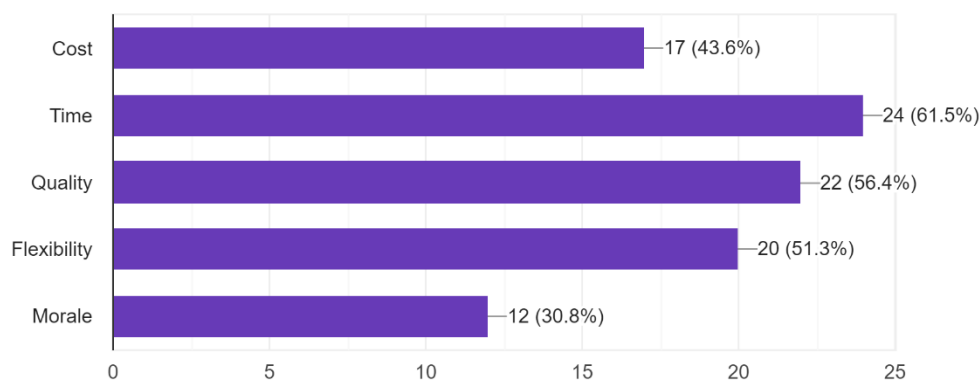
Another notable factor contributing to this problem highlighted during the interview was resistance to change by project managers during the initial stages of transformation from traditional waterfall to agile model. This could possibly due to some level of apprehension and fear of change that causes a resistance within project managers to transform towards agile methodologies. (cPrime, 2021) mentions the resistance are a result of the organizations failing to manage transformation to agile proactively and strategically, leading to reduced morale amongst team members and productivity. Hence, P24, resistance to accept agile principles or wrong understanding of agile approaches, can also be attributed to project manager's incompetence or displaying an autocratic leadership not suitable for agile projects. As resistance to change can be difficult to resolve at individual organization level, it can be even harder in multi-vendor setup with several organizations involved.

"This is a very severe problem with project managers where they don't want to adopt to the new ways of working. It happens initially and then resolved if the project manager gets used to agile way of working. How this is resolved is by agile coaches so the responsibility of agile coach is to change the mindset of organisation. Organisation means everyone and is not just the big people it starts from top to bottom so with the help of agile coaches this problem can be solved. The impact is actually frustration in team members that there is not much communication between team. Even if there is a daily scrum call and if it is dominated by project manager who is not supporting or not listening to the team there is no use of it and it will decrease the morale"

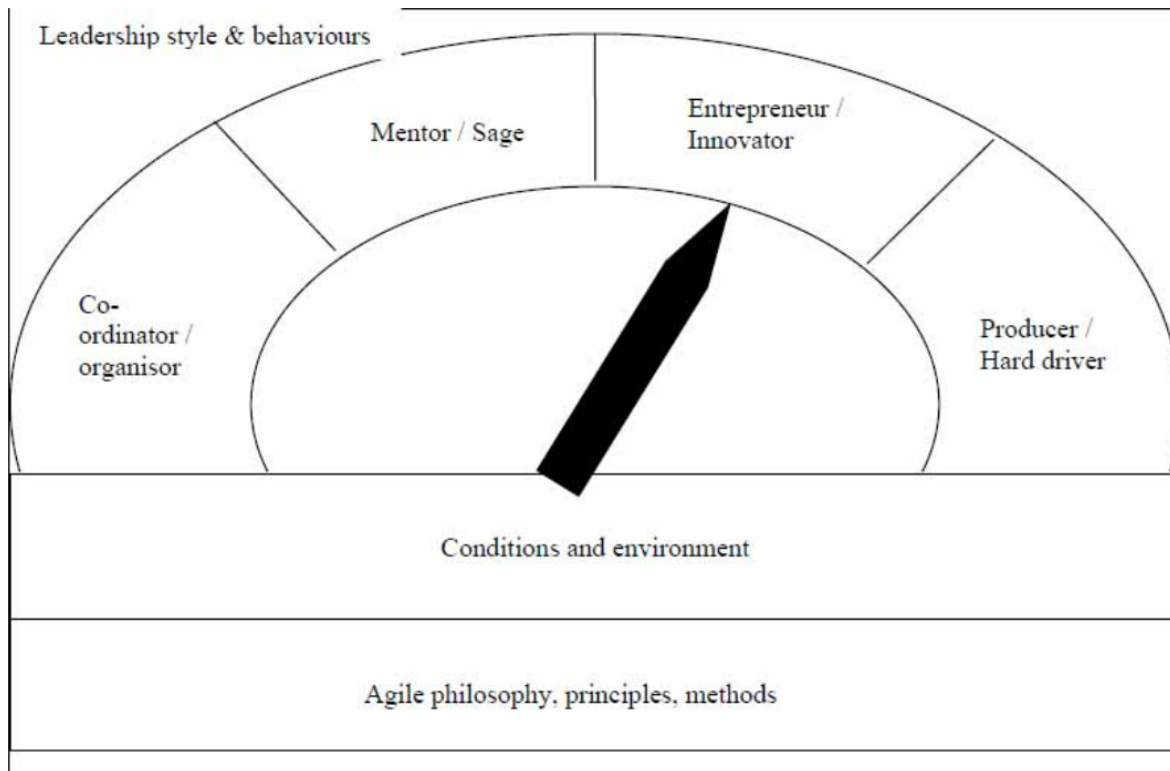
Almost all the interviewees mentioned both 'Flexibility' and 'Morale' as the main project performance factor impacted by problem P23 and survey responses also indicate a high rating on 'Time' and 'Quality' which agrees with (Banerjee, 2012) that a sense of differentiation in inter-organizational resource-managerial relationship can lead to reduced productivity and defect rate.

Which of these Project Performance factors are impacted due to the previously rated problem?

39 responses

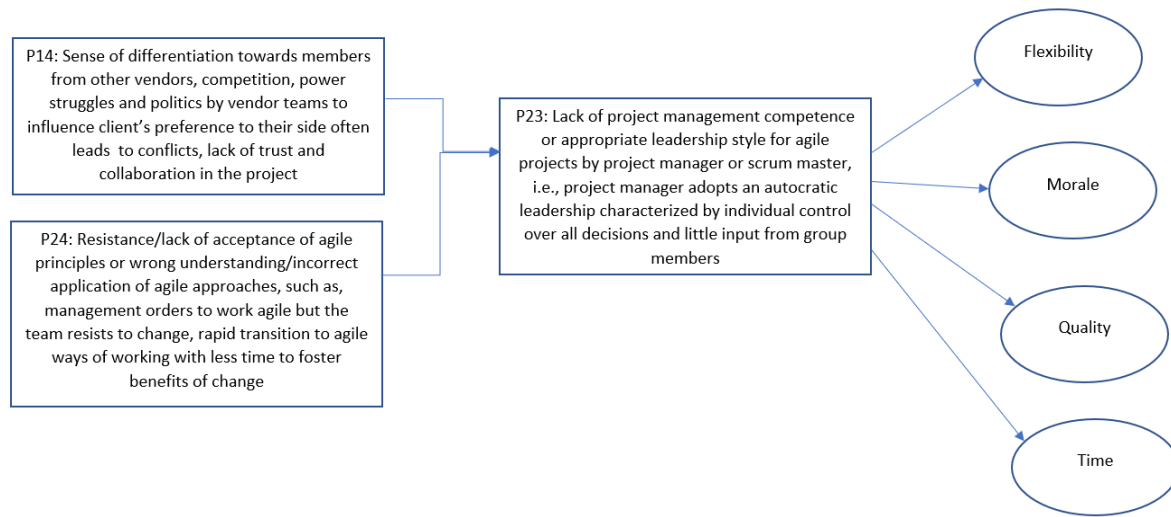


So, it becomes imperative that project managers of agile teams has to adopt - flexible and adaptive leadership style that promotes combined decision making over individual control; Partnering approach that promotes understanding, cooperation and trust between each party in multi-vendor project contribute to client's business strategy; No-Blame approach that encourages to learn from mistakes rather than blame other vendors to attract client attention; One-team approach that promotes working together as one team with freedom of access between each party (Farley, 2005). They also further mention that project managers in agile projects need to be like a chameleon with an ability to adapt to daily or hourly challenges common in fast paced agile projects, while the values and methods remain constant but the leadership approach taken varies based on conditions and people as on a dial shown below.



Agile Project Managers- leadership - figure adopted from - (Farley, 2005)

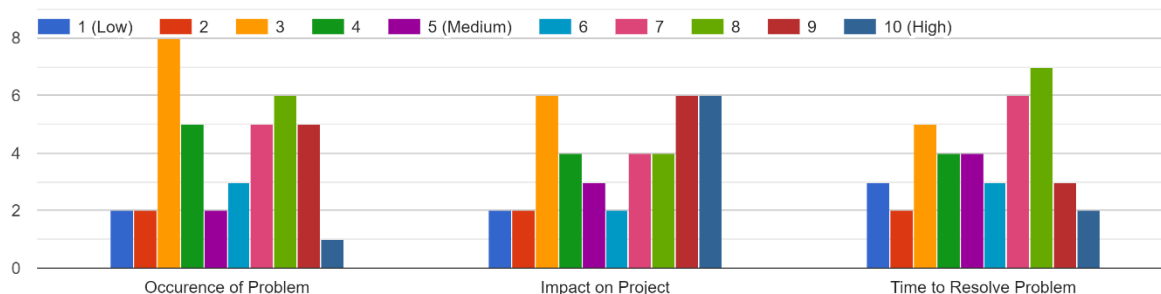
Relativity between the problems: based on our analysis, it can be concluded that P14 and P24 are other problems that can have some influence on project managers leading to problem P23, ultimately impacting Project Performance factors – Flexibility, Morale, Quality and Time



P19: Lack of continuous Client/Business User involvement due to distributed nature of the project, such as communicating to business and getting feedback on matters

Based on the survey analysis, P19 is ranked as fifth critical problem in agile scrum teams with a cumulative weighted score of 6.4 (see table)

Lack of continuous Client/Business User involvement due to distributed nature of the project, such as communicating to business and getting feedback on matters



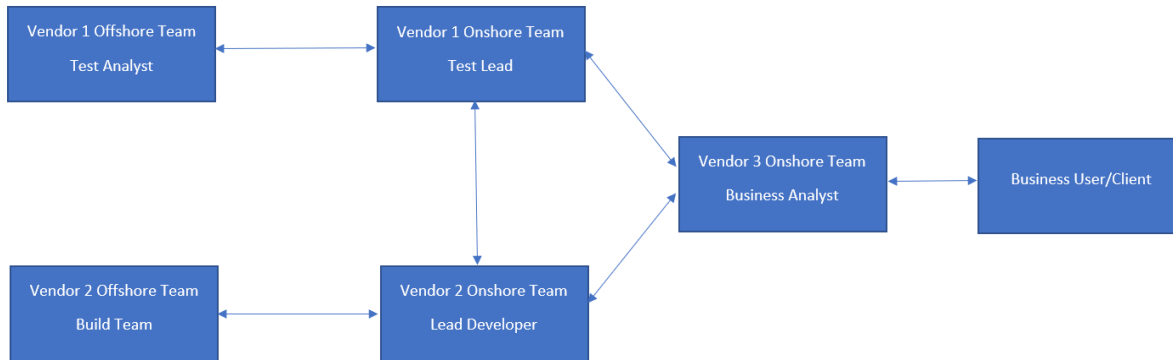
Almost all the interviewees highlighted two causes for this problem – one is the distributed nature of the project and other is the layers of hierarchy within and between the teams. People who worked in locations different from the client faced difficulties and delays in communicating with client or business and getting their feedback because of time zone differences, geographical differences, organisational and cultural differences. This can also be supported by the fact that 14 of the 20 respondents who rated this problem as high (above 6) are located offsite. This proves to a certain extent that problems with effective communication such as P8 and P9 also has some influence on agile teams leading to P10, difficulty in communicating and getting feedback from client or business by resources located offsite than those who are located onsite.

"I think this is more felt with the people who are at offshore than the people who are at onshore because onshore people they daily interact with BA's, developers and other members of the team. But offshore people, they are mostly relying on their leads who are there at onshore. So, whenever they want some critical information, they firstly they usually go to their lead and then to the other members of the team at onshore. But I think sometimes what happens is people at offshore So what makes them a little hesitant to ask directly to the person at the higher position who is at onshore or client site So um, but I think there's a difference when it comes to access of information, critical information or sometimes which may hamper your work. And yeah, maybe the person who is at the power, you sometimes people are hesitant to ask questions directly, and they did take a long route through the lead and that lead will follow the person. So that may also, you know it. It's definitely it's a time-consuming process, but that is something that impacts the project also. And it may take two three days to get that information. But I think I think there should be collaboration. Yeah, that that comes to the collaboration. I think there should be a culture of openness where you can ask question even people at the higher level OK, so. That, you know, people should be encouraged to speak up. And no matter what their position is."

"Yes, this is a common problem. In fact, I am playing a role of business since last many years. So, this is one of the challenges which I see in my job, whether whenever I'm representing service industry or whether I'm representing from offshore perspective. I always face this issue. There are certain restrictions wherein I cannot get in touch with business users directly. It could be because of geographical differences, time zone differences or my organisation differences, like my organization always asked me to follow certain hierarchy to reach to business users. Because as we know all or in many cases, all the businesses with our high-ranking officers and the people working on offshore on a normal job could or may not reach them directly so this hierarchy difference are always preventing me in speaking with business users directly. So, I what I always need to do is I need to speak with my manager first when I'm saying manager it is product owner or any senior person in my project, or it could be my counterpart with from our organisation at the onshore department. And he will then speak with the business users to set up my meeting. So, there is a lot of Gap, a lot of delays are happening in this communication. Rather, if I have direct access to business users, it would have been more efficient and more productive. So, and less error prone free. Impact on overall end to end Productivity is huge"

Another reason highlighted was the hierarchy levels within same team, between vendor teams, or between vendor-client teams resulted in communication delays and information quality deterioration. As mentioned by one interviewee, a model of communication link within their agile team (figure x) depicts the communication between vendor teams, and between vendor and client business team. It can be seen that either of Vendor teams 1 and 2 located offshore and at junior levels does not communicate with each other and communication happens only through their respective onshore counterparts at senior level, and either of Vendor 1 and 2 onshore team does not communicate directly with Business/client team but only through business analyst from Vendor 3 onshore team. This form of communication arrangement could be either imposed by vendor organisation to avoid possible reputational loss that may arise from direct discussion between junior level resources and client or other vendors, or as a result of business/client preference. This can be further complicated as the number of vendors and team size increase and also when layers of management such as Technical Manager, Line Manager etc. are added to this hierarchy as one interviewee mentions. (Pattit & Wilemon, 2005) mentions teams of sheer size can often lead to high communication quality deterioration and delays owing to the number of communication links between team members. Based on this analysis, we can conclude that P7 - sheer size of the team or number of communication links within the team also contributes to this problem. This seems contradicting to one of the four values of

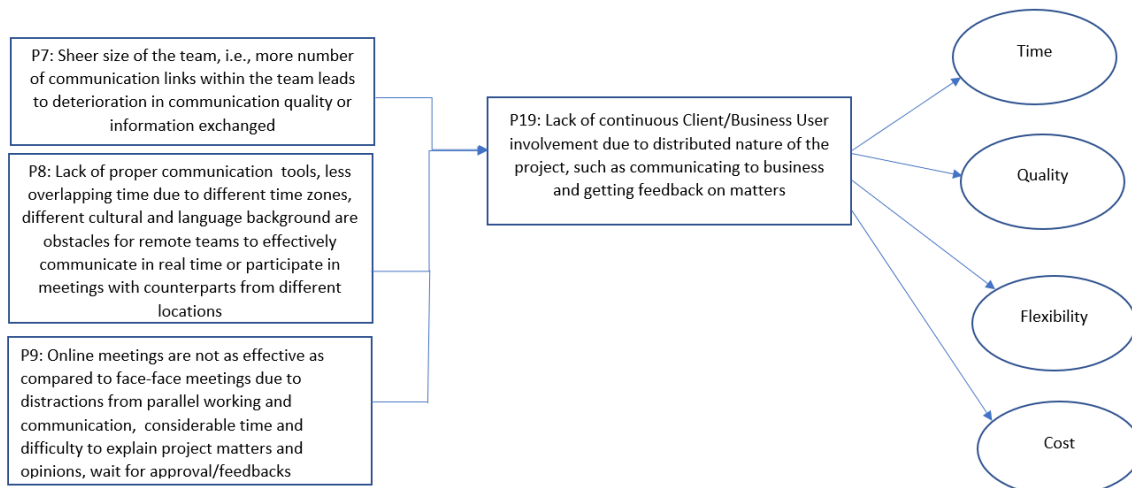
agile manifesto “individuals and interaction over processes and tools” as this form of communication through hierarchy levels hampers interaction and collaboration between individual team members and organisations should remove unnecessary hierarchy levels and facilitate a sense of openness and direct communication between team members at levels to prevent this problem.



Yep, definitely. So, I've seen, you know well of basically this with consultancies. So, you know, they do have the structure where, you know, first you need to ask your team lead, you just need to explain this to your team leader. And then he goes to his next onshore coordinator that he speaks to the business manager in this transmission. You know, I will say 10 things to my lead. He will transmit 8 things to his counterpart he will transcripts 6 things to business user, you know, So, opening up a channel with business usually and if it is not possible every day, better to have a weekly call. And allow the team directly to interact with the businesses will reduce the stress. That's a bread and butter for all those consultants? Isn't it? So, people keep a hierarchy, delay the process and gain the money out of it. So, if you put extra layer there, so why do you need two managers? Why you need technical manager and people, manager, you don't need them right? why you need a team lead in between? So, this is the case for most of the consultancies, and that is already embedded in our system. So, avoiding that could be a very great difficulty I would say. In some cases, to protect the reputation, but sometimes maybe the person might not be having that much experience writing email, so they will change the wording and make the statements nice and will send it forward. So doing that will change the actual, you know, meaning of communications and all those bits of them they impact quality of communication and time is the first one. But quality also and obviously cost as well right like you got too many layers, then you're spending a lot of money there.

Almost all the interviewees mentioned ‘Time’, ‘Quality’ and ‘Flexibility’ as the main project performance factors impacted by problem P19 because of the communication links formed by hierarchy levels and survey response also indicates the same. However, a few attributed the additional management hierarchy levels as contributing to increase in ‘Cost’ of projects.

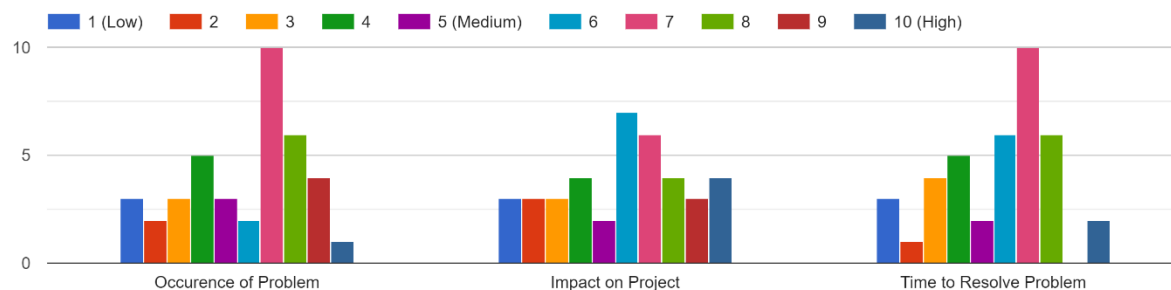
Relativity between the problems: based on our analysis, it can be concluded that P7, P8 and P9 are other problems that can have some influence on agile teams leading to problem P19, ultimately impacting Project Performance factors – Time, Quality, Flexibility and Cost



Inability to breakdown backlog into smaller user stories and prioritise based on most business value that can be delivered within a sprint 'or' horizontal breakdown of P18: stories based on tasks such as design, coding, testing, etc., instead of product features often leads to work spanning over multiple sprints and not delivering working software or business value at the end of each sprint

Based on the survey analysis, P18 is ranked as SIXTH critical problem in agile scrum teams with a cumulative weighted score of 6.3 (see table)

Inability to breakdown backlog into smaller user stories and prioritise based on most business value that can be delivered within a sprint 'or' horizontal breakdown of storie... working software or business value at the end of each sprint



Most interviewees responded that the competency of product owners, project managers or scrum masters play a crucial role in this problem and they should have an expertise in the field of agile project management – planning and estimation of user stories such that a minimum viable product can be delivered within a sprint, and subject matter expertise in their project domain, in order to guide and lead the agile scrum teams. Furthermore, they should also adopt the leadership model that brings together the team as a single unit to support in decision making, planning and estimation of user stories in way that a minimum viable product can be delivered each sprint (P4). (Verwijns, 2017) mentions that splitting up product backlog horizontally results in smaller individual work items for design, database, coding and testing, but they don't deliver working software or business value on their own until the outputs of all the individual work put together generates business value. This view was associated with

multi-vendor setup in a way that each vendor responsible for individual work items such as testing, design etc., could think in silos, leading to a horizontal sequential planning of development tasks spanning over multiple sprints as opposed to all the tasks performed concurrently within the same sprint. However, our interviews identified that this is not the case always, unless the system is complex and not modular in which case a hybrid project management-traditional+agile is followed, but the main reason in most cases is the agile project management competence that fails to facilitate team involvement in planning of MVP's deliverable within a sprint. Based on these points, it appears P23 - Lack of project management competence or appropriate leadership style for agile projects by project manager or scrum master has a direct influence on this problem.

"This is we need to have appropriate business designer or product owner who will like the catalogue or refinement should be such that they should focus on MVPs not the big bang on. this is the major principle of agile and anyone not following that then the project will not succeed. So, the solution for this would be that the product owner should be a SME in that area so if you have one product for example payment and you bring a product owner who has not worked in payments earlier or worked in another domain so this kind of problem will happen so basically the product owner or the business team should be an SME in that area or domain to avoid these issues. And team should also help in the refinements they should also give their inputs for example some team have 2-week sprints or 3-week sprints right so they need to pick MVP or whatever product they are building properly they need to do proper estimation and if estimation goes wrong also it creates issue right"

"It will have a hierarchy like will be there will be a product owner a scrum master, a development team layer, a testing team layer, followed by a business analyst. If we have this set up, I will not care how many vendors are involved in this. For example, if I have a development team of 10 people. There could be a chance like this. Then the 10 developers could be from different organizations. Then I will not care from which organism they are. I will just align my product owner or my scrum master, to assign the user story in a way at the end of each spring, I should be having the end to end working software to be delivered. OK. I don't think like there will be a challenge in how many vendors are involved in this because I treat every person as a different vendor until we understand there are still examples that we are able to deliver a successful product at the end of every sprint. So, I don't think like having multiple vendor setup will have a real impact on delivering a product in a single sprint so that it could be doing, but that can be definitely mitigated by having the right scrum master in place who has good understanding of the product. Or we can have an individual product owner on top of everyone where he can assign the right backlog during each story and they can track the progress."

One interviewee mentions the project manager should have the skills to assess and select appropriate project management method when the system or project is not suitable to deliver in iterations in agile way.

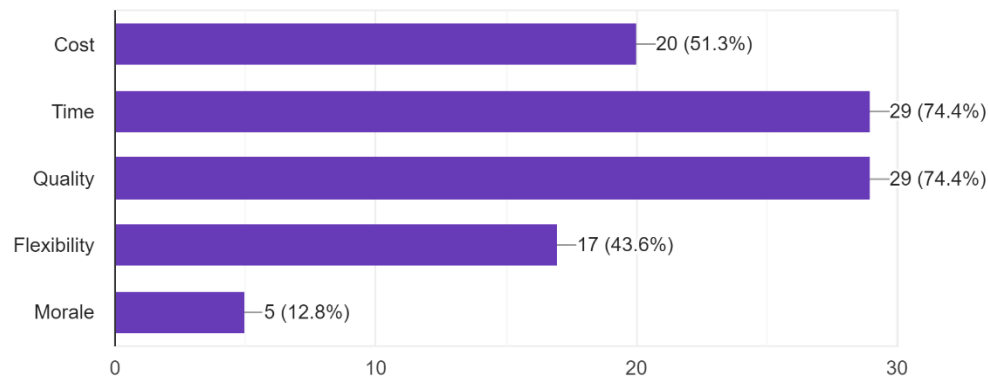
"I have faced a bit of difference over there whenever we talk about the infrastructure project, working in the agile way. Those are kind of a hybrid way of agile as you know the word itself say that we need to be agile, OK? That means we need to be flexible in terms of what the process and other things that we have implemented in one way we don't need to replicate the same into the other one. But what I meant by that is by infrastructure. There are certain things where we have to plan design for the overall infrastructure. OK, so there is nothing like we keep on building something and later on it if the design is changed because infrastructure is something which is very, very costly. You cannot afford to make the changes at the later stage or let say for example. You required five load balancers, which is a F5. OK, the cost of one load balancer is let's say, 10 grand. OK. You don't have a proper design initially, and you work with the same way. OK. And later on, you came to know that you need 8 load balancers. That time it would be too much of the effort. Or you might say that this load balancer that we have, opted, they are not supporting the software or the application. So, we have to replace them. Then replacing at that point of time would make things more complex rather than we doing the complete solutioning part, at least on the infrastructure projects"

Almost all the interviewees mentioned 'Time', 'Quality' and 'Cost' as the project performance factors impacted by problem P18 because inability to plan MVP's that can be delivered within a sprint will eventually lead to the same problems in traditional way of delivering projects and survey response also indicates the same.

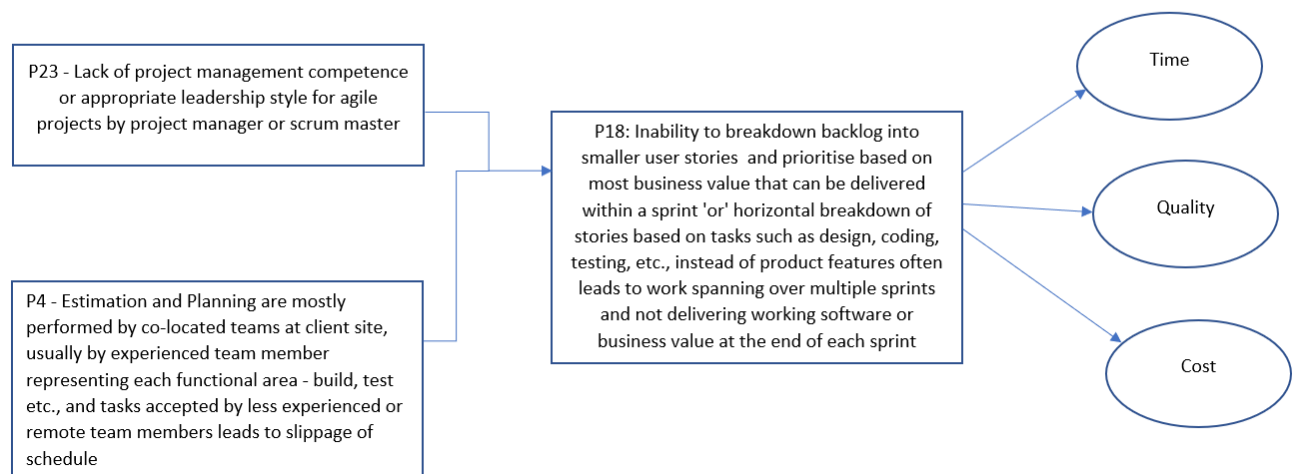
"Impact of the problem is waste of time and waste of effort. you keep people waiting and it is an increase the cost and you won't deliver on time like you might deliver on time because the timeline will get extended because the main thing is you will get more issues because you don't test it early in the lifecycle, it will be again like going to traditional waterfall model and even if you are agile, you are not breaking items into smaller iterations and you are doing the same thing in traditional manner in the new name of agile"

Which of these Project Performance factors are impacted due to the previously rated problem?

39 responses



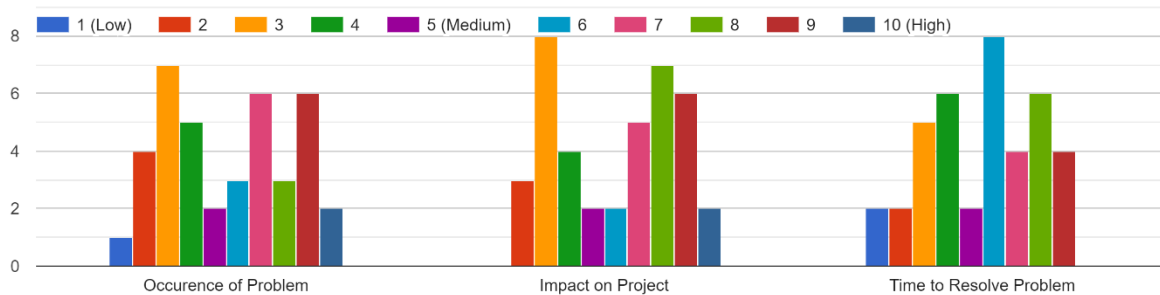
Relativity between the problems: based on our analysis, it can be concluded that P23 and P4 can have some influence on agile teams leading to problem P18, ultimately impacting Project Performance factors – Time, Quality and Cost



P2: Problems such as changing priorities, changing scope and conflicts with resource availability by teams within the project and interfacing teams outside project prevents tasks from being completed within sprint timelines

Based on the survey analysis, P2 is ranked as seventh critical problem in agile scrum teams with a cumulative weighted score of 6.1 (see table)

Problems such as changing priorities, changing scope and conflicts with resource availability by teams within the project and interfacing teams outside project prevents tasks from being completed within sprint timelines



Projects often face the risk of unresolved problems at the integration touch points, especially in projects involving multiple vendors (Akshanthula, et al., 2019) as individual team members focus mainly on their allocated tasks with less or no attention towards their interfaces until or after they have completed their tasks (Rosenau & Githens, 2005). Interviewee 1 & 4 highlighted that lack of synchronisation or effective collaboration between interfacing project entities that are dependent on each other's deliverable can lead to delays in completing the overall project as synchronisation problems can lead to development of independent plans without considering dependencies and lack of collaboration can prevent changes to plan such as scope, priority communicated at the right time to dependent teams. This agrees with (Struckenbruck, 1998) views that failure to achieve project integration that brings together people, organisational and system interfaces together to enable project to function as a whole integrated unit can lead to administrative problems such as changing priorities and resource conflicts, technical problems such as changing scope, and client problems such as interfacing teams not having a common understanding of customer specification. So, it is evident that lack of direct and informal communication (P6) between vendor teams can lead to lower awareness of others activities and poor coordination in planning and executing the work in relation to others activities in a direction towards the project objective as highlighted by (Korkala & Abrahamsson, 2007).

It usually happens. I mean, for example, you have the team has finished their piece of work, but we are also relying on the other team for their work to be completed so that we can test it end to end. So that makes it difficult. Then when there is a there is no synchronisation between two entities that are doing that will only be on upon with the work is working going on. So, I think it's again the collaboration, the time that timelines that has been decided for the for the for the for the piece of work that been, but that has been done from different, different entities for that project. Maybe a daily communication or maybe daily meetings or whatever. Update, even if it is for 15 minutes, we should be doing OK.

It depends if the project is a legacy system where like the technology between the verticals are different and then you are dependent on the system it has to happen. take an example of our organisation where we have different squads and each squad has their own priorities. so, if you have a dependency on them for example if you want them to consume the values which we are sending through webservices ok so you want them to make design and build changes at their end. it may be priority for you but not priority for them so they might say that they have some other priority work that they need to complete first. So, you need to wait for it. So, it happens when the product landscape is very and it interfaces many systems.

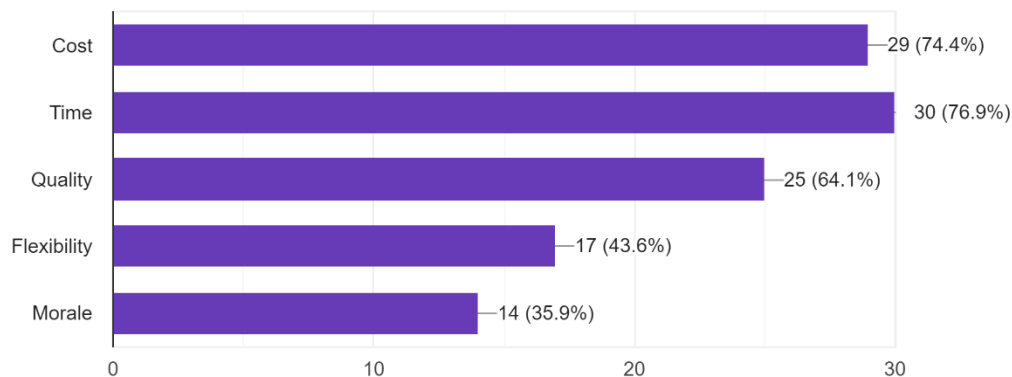
One interviewee highlighted estimates influenced by the vendor management as a reason for the resource unavailability in the project teams. They mention that estimations performed at management

levels sometimes does not reflect the actual effort required to perform a certain task and they suggest the ideal approach is to include the technical resources who actually perform the work to be involved in estimation process to produce more accurate estimates required to accomplish the tasks within certain timeline. This reveals problems with estimation such as P5 can have some influence on this problem.

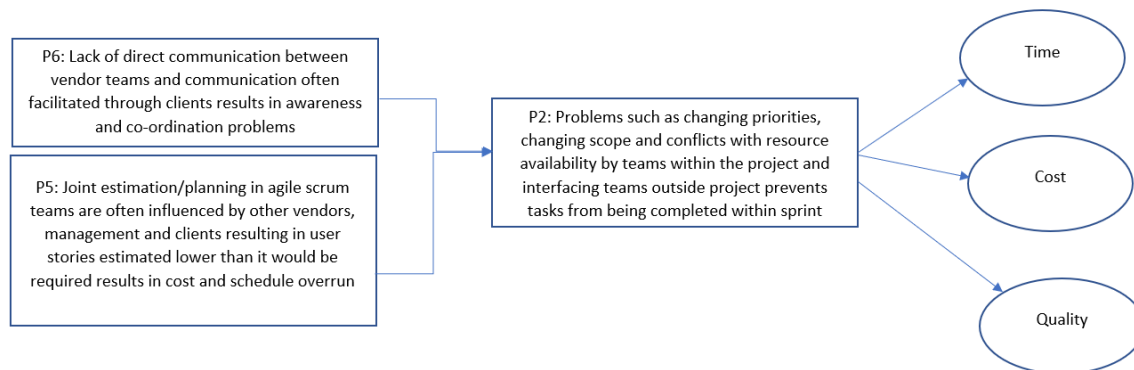
So, the key thing here is when the planning session is going on for a resource let's say if you are going to start and you need these many developers and you know that specific quote whoever look at the statement of work and thinks that it takes 3 developers to develop it right but in our current world that person who has given the quote is not the one who is you know sitting and developing it. That's the big difference you know instead of you asking some other person it is better to onboard or include the developer in stage 1 estimation and let him understand requirement and let him give resource estimates instead of going to their manager or lead just giving random numbers of developers. so, I would suggest include whoever doing the work and let him do the estimates and that will clearly give you a picture how many resources you will need so based on that you can onboard resources at the beginning of the project

Almost all the interviewees mentioned ‘Time’, ‘Cost’ and ‘Quality’ as the main project performance factors impacted by problem P2 because of the schedule overrun resulting from changing priority and resource unavailability and the survey response also indicates the same.

Which of these Project Performance factors are impacted due to the previously rated problem?
39 responses



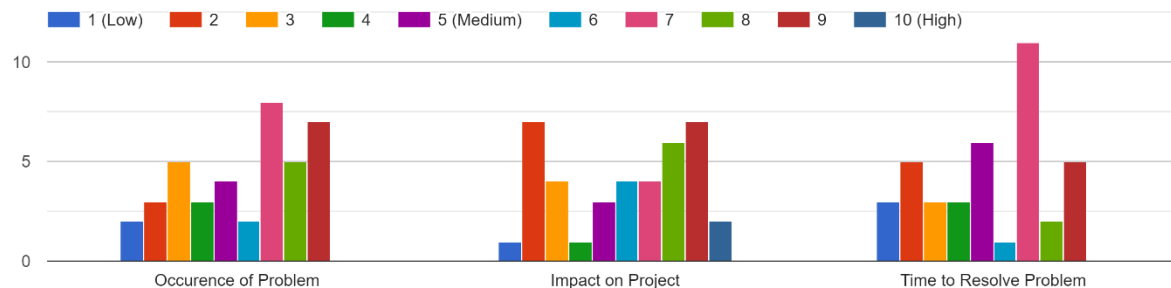
Relativity between the problems: based on our analysis, it can be concluded that P6 and P5 are other problems that can have some influence on agile teams leading to problem P2, ultimately impacting Project Performance factors – Time, Cost, and Quality



P11: People often allocated to multiple projects/teams on shared basis results in discontent and stress personally and interruption to team members seeking input on overlapping urgent issues

Based on the survey analysis, P11 is ranked as EIGHTH critical problem in agile scrum teams with a cumulative weighted score of 6.0 (see table)

People often allocated to multiple projects/teams on shared basis results in discontent and stress personally and interruption to team members seeking input on overlapping urgent issues



(Wiesche, 2021) highlights this problem as an environment-imposed interruption in agile scrum teams and his view is that a team member may work on multiple projects simultaneously due to resource constraints, expertise on a specific area, or the project not requiring a full-time resource, can cause interruptions in decision making when other team members require the input of shared resource on overlapping urgent issues. The information we gathered through our interview agrees with their views and the reasons mentioned by interviews for people shared across multiple projects are shortage of niche skills or dependency on subject matter expertise causes only a few people with those skills or expertise to be shared, project not having requirements or work for the resource to be fully allocated that forces vendors to assign resources to multiple projects for better utilisation, vendor strategy to increase profits by accomplishing tasks with less resources and sometime as temporary measures to cover absenteeism in other teams. Survey analysis also reveals that atleast 60% of the respondents who rated this problem as high are those with an experience of 5 years or more, who would ideally be subject matter expertise or those that possess niche skills. So, with this, we can agree that lack of different skillsets discussed in problem P20 not only hampers the teams from being fully cross-functional but also further leads to people shared between projects to fulfil the deficiency. An ideal

solution to prevent this problem is to promote cross-skilling, knowledge sharing and training to remove dependency on specific resources.

"Yeah, this is a problem because the root cause of it will be the dependency on one person you don't have another team member or only that person or that team knows the functionality or they are SMEs on that. We keep them shared basis and whenever required we take the inputs but it doesn't work very well because if there are more than 4 to 5 scrum team then it is difficult so it is always based on priority so if a person is 50% involved in one project, then only 50% is left and that person is needed in 4 or 5 teams. You can do that and usually that solution architect will be shared that's ok as you won't build and complete system in a sprint but it shouldn't happen with all the team like development team, environment team, testing team then it will create problem. Yeah, it will definitely delay work and the person shared across multiple teams will be too stressed and also proposition of the team also depends for it to be efficient they should be taking care of all like system application area in all perspective like designing, build like everyone should know especially development, testing, environment support like every person should be part of the team I mean the team should be formed of all the required resources"

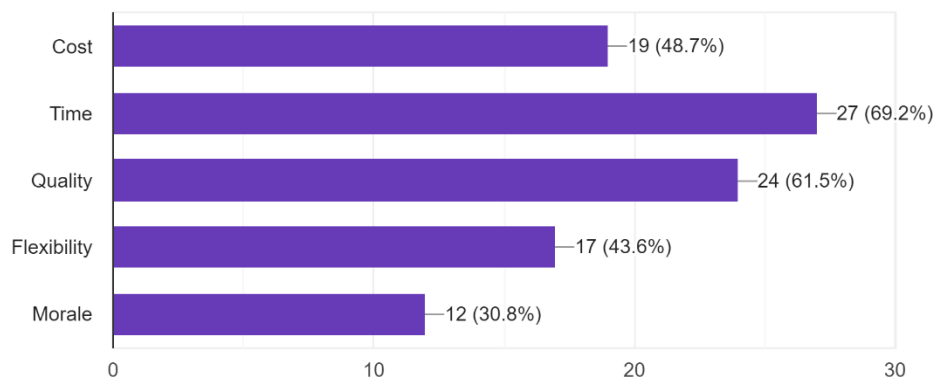
"Main driver for this, I feel, is, you know, to get the best out of less money. But you know, out of less budget you want to get the best coverage of, you know, multiple resources might need to finish a single job. You know, that's what you want to look at your bag as well right so you cannot get a resource whose job is one hour for a month but you cannot book him for the whole month right so a better planning would help here I would say."

Almost all the interviewees mentioned 'Time', 'Quality' and 'Morale' as the project performance factors impacted by problem P11. Although it provides flexibility for vendor management to share resources between multiple teams but it impacts the 'Flexibility' within the agile scrum teams if the resource is not available due to overlapping tasks.

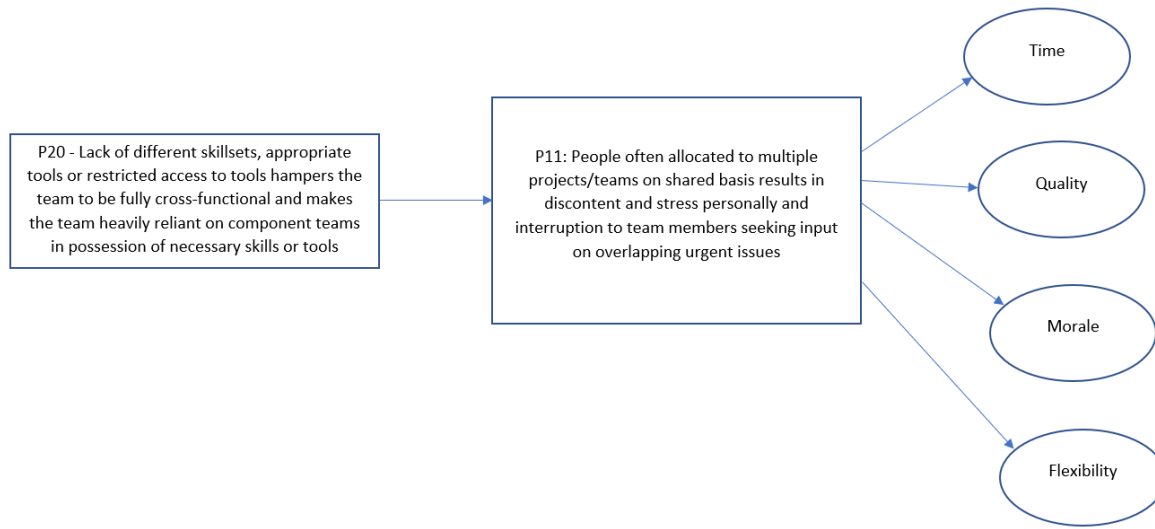
"I think when people are working on shared business or I highly doubt like if the work is being done with the with the highest quality because It's like putting your legs in two different boats, right when you cannot concentrate properly at one piece of. And you can't deliver the highest quality. So, it may well cause stress to the person who is working on a shared basis. And also, it may impact the task itself because sometimes they give more in work the other team and so may even sometimes hamper the overall progress of the project. It will impact the morale of the person, quality and definitely persons work satisfaction will go down because of this"

Which of these Project Performance factors are impacted due to the previously rated problem?

39 responses

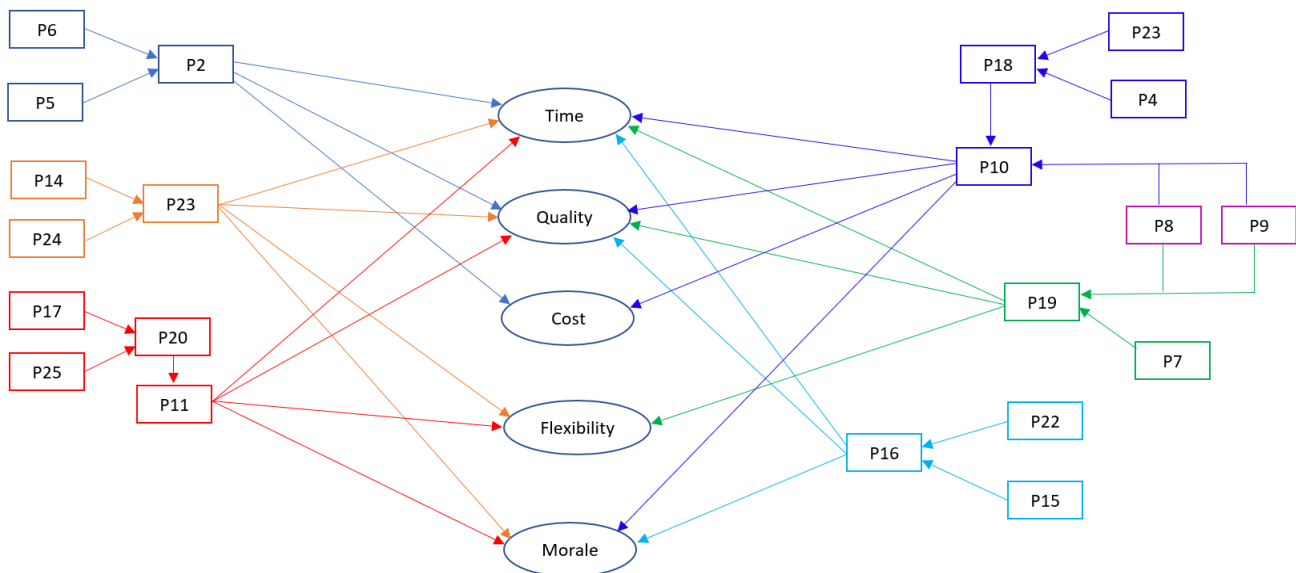


Relativity between the problems: based on our analysis, it can be concluded that P20 can have some influence on vendor teams leading to problem P11, ultimately impacting Project Performance factors – Time, Quality, Morale, and Flexibility



Conclusion

Based on the research analysis and findings, following cause and effect diagram is developed that represents the inter-relationship between the problems and their impact on project performance factors in multi-vendor globally distributed agile scrum projects.



Following propositions developed based on this cause-effect diagram states the failure modes or problems that can impact performance of multi-vendor globally distributed agile scrum projects:

Proposition 1: Lack of direct and informal communication between interfacing teams and between vendors within an agile team can cause lower awareness and synchronisation of activities performed by vendor teams and interfacing teams, Estimation and planning activities performed at the management level without the inputs of team members can lead to inaccurate estimation and resource planning, either of which in the event of changing priority, scope or resource unavailability within an agile scrum team or by interfacing team can impact project performance factors Time, Cost and Quality.

Proposition 2: Sense of differentiation or biasing attitude towards resources from other organisations and resistance to change leadership styles appropriate for agile projects can cause project managers make individual decisions and avoid partnering, no-blame and one-team approach with other organisation members, eventually impacting project performance factors morale, flexibility, quality and time.

Proposition 3: Lack of effective governance process at agile team level and concerns sharing knowledge with other vendors can impair cross-skilling and knowledge exchange within agile teams leading to shortage of skilled resources, whom in the event of increased demand has to work on a shared basis in multiple teams impacts project performance factors flexibility, morale, time and quality.

Proposition 4: Lack of hygiene factors at work such as growth and recognition of achievements by vendor and client management can lead to reduced motivation causing frequent and high number of employee attrition due to multi-organisational team structure, which in turn can lead to changes in the team composition impacting project performance factors time, quality and morale during the learning phase of the newly hired resources.

Proposition 5: Hierarchy levels in vendor-vendor and vendor-client communications causes communication delays and quality deterioration and the underlying issues with communication in distributed projects such as different time zone, lack of face-to-face communication prevents the team from getting quick turnaround on queries and feedback from business and client teams.

Proposition 6: Planning a minimum viable product that can be delivered within a sprint requires involvement of all the team members in estimation and planning activities, Project Managers/product owners with expertise on agile project management and sufficient domain knowledge, and effective communication within the team and with interfacing teams, which when not available in a team can make it vulnerable to frequent changes in requirements, eventually impacting time, quality, cost and morale.

Limitations

There were no major variations in the original plan for this research, apart from the number of survey responses that could be obtained. Atleast 100 survey responses was estimated to provide more accurate results on this research but only 60 responses could be obtained. This was due to the survey being

targeted only towards software development professionals and limited time available for the research, and most people were not available because of the Christmas holiday period. However, the research concludes that most problems are inter-related and lead to problems stated in the six propositions.

Recommendations for Future Work

Our recommendation for future work is that a case study of six propositions identified can be tested against agile scrum teams at different maturity levels to assess the variations in project performance, as our findings indicated that problems such as resistance to change, inability to breakdown backlog into smaller user stories, frequent changes in requirement are more prevalent in newly formed agile teams.

References

Accenture: SolutionsIQ, 2015. *12 Days of Agile Principles! Day 6: Face-to-Face Conversation is Best for Conveying Information*. [Online]

Available at: <https://www.solutionsiq.com/resource/blog-post/12-days-of-agile-christmas-day-6face-to-face-conversation-is-best-for-conveying-information/>

[Accessed 01 December 2021].

Akshanthula, S. et al., 2019. *White Paper on Managing Projects in a Multi-Vendor Environment*.

[Online]

Available at: <https://www.pmi.org.in/EEF/White-Paper-Series/pdf/Multi-Vendor-Environment.pdf>

[Accessed 18 11 2021].

Al-Salti, Z., Hackney, R. & Ali, M., 2011. Factors impacting Knowledge Transfer Success in Information Systems Outsourcing. *Journal of Enterprise Information Management*, 27 09.24(5).

Amerongen, R. V., 2008. *Agile software development, the principles. Principle 6*. [Online]

Available at: <https://technology.amis.nl/it/agile-software-development-the-principles-principle-6-the-most-efficient-and-effective-method-of-conveying-information-to-and-within-a-development-team-is-face-to-face-conversation/>

[Accessed 01 December 2021].

Apptio, 2018. *3 Signs Your Sustainable Pace is Killing Your Developers*. [Online]

Available at: <https://www.apptio.com/blog/sustainable-pace-killing-developers/>

[Accessed 28 November 2021].

Armstrong, M., 2006. *Performance Management: Key Strategies and Practical Guidelines*. Third Edition ed. London: KOGAN PAGE.

Association for Project Management, 2015. *Conditions for Project Success: APM Research Report*, Buckinghamshire: Association for Project Management.

Association for Project Management, 2021. *Becoming Agile*. [Online]

Available at: <https://www.apm.org.uk/resources/member-resources/becoming-agile/>

[Accessed 05 August 2021].

-
- Babaian, A., 2019. Becoming Agile with Scrum Framework. *Software Quality Professional*, December.22(1).
- Banerjee, N., 2012. *Multi-sourcing conflicts: What does a manger focus on? A system dynamics based analysis*, Massachusetts, US: Doctoral Dissertation, Massachusetts Institute of Technology.
- Beck, K. et al., 2001. *Agilemanifesto.org*. [Online]
Available at: <https://agilemanifesto.org/>
[Accessed 07 08 2021].
- Beck, K. et al., 2001. *Manifesto for Agile Software Development*. [Online]
Available at: <https://agilemanifesto.org/>
[Accessed 01 12 2021].
- Belden, J., 2019. *High performing multi-vendor transformation teams – How to make it work*. [Online]
Available at: <https://www.cio.com/article/215782/high-performing-multi-vendor-transformation-teams-how-to-make-it-work.html>
[Accessed December 2021].
- Bruhn, Z. & E, M., 2003. Interruptive events and team knowledge acquisition. *Management Science*, 49(4).
- Bruyne, M.-J. D., Moens, E. & Vanhoucke, M., 2021. Project Recovery: Project Failures and How to get rid of them?. *Journal of Modern Project Management*, August.9(1).
- Carmel, E. & Agarwal, R., 2001. Tactical approaches for alleviating distance in global software development. *IEEE Software*, March.18(2).
- Chow, T. & Cao, D.-B., 2008. A survey study of critical success factors in agile software projects. *Journal of Systems and Software*, 81(6).
- Conforto, E. C. & Amaral, D. C., 2016. Agile project management and stage-gate model—A hybrid framework for Technology based companies. *Journal of Engineering and Technology Management*, 40(1).
- cPrime, 2021. *Resistance to Agile? It Can Be Overcome..* [Online]
Available at: <https://www.cprime.com/resources/blog/resistance-to-agile-it-can-be-overcome/>
[Accessed 1 February 2022].
- Deloitte, 2021. *A call for accountability and action: The Deloitte Global 2021 Millennial and GenZ Survey*, s.l.: Deloitte Touche Tohmatsu Limited.
- Desouza, K. & Evaristo, J., 2004. Managing Knowledge in Distributed Projects. *Communications of the ACM*, 01 04.47(4).
- Digital.ai, 2021. *15th State of Agile Report*, s.l.: Digital.ai Software Inc.
- Dingsoyr, T. H. et al., 2006. *Developing Software with Scrum in a Small Cross-Organisational Project*. Berlin, Hiedelberg, Springer.
- Draper, B., 2017. *Governance: Using a RACI Matrix Tool*. [Online]
Available at: <https://blogs.k-state.edu/its/2017/12/08/governance-using-a-raci-matrix-tool/>
[Accessed 25 January 2022].
- Farley, J., 2005. *Leadership in agile projects -- what makes for success?*. Newtown Square, PA, PMI: Project Management Institute.
- Gandhi, A., 2020. *Should RACI be included in Agile Methodology Projects?*. [Online]
Available at: <https://www.linkedin.com/pulse/should-raci-included-agile-methodology-projects-ankit->

[gandhi/?trk=public_profile_article_view](#)

[Accessed 25 January 2022].

Gandomani, T. J. & Nafchi, M. Z., 2016. Agile transition and adoption human-Related challenges and issues: A Grounded Theory approach. *Computers in Human Behaviours*, 62(1).

Gogan, J. L. & Rao, A., 2008. *Deal or No Deal? Vendor Issues in Two Multi-Organization Pilot Test Projects*. Rochester, NY, IEEE.

Hansen, M. T., Nohria, N. & Tierney, T., 1999. What's your strategy for managing knowledge?. *Harvard Business Review*, 77(2).

Hayat, F., Rehman, A. U., Wahab, K. & Abbas, M., 2019. *The influence of Agile Methodology (Scrum) in software development*. Toyama, Japan, IEEE Computer Society.

Hazrati, V., 2009. *Resource Management in Agile Projects*. [Online]

Available at: <https://www.infoq.com/news/2009/06/resource-management-in-agile/>

[Accessed 16 December 2021].

Holland, S., Gaston, K. & Gomes, J., 2000. Critical success factors for cross-functional teamwork in new product development. *International Journal of Management Reviews*, 2(3).

James, M. & Walter, L., 2017. *Scrum Reference Card*. [Online]

Available at: https://www.collab.net/sites/default/files/uploads/CollabNet_scrumreferencecard.pdf

[Accessed 15 December 2021].

Jett, Q. R. & George, J. M., 2003. Work Interrupted: A Closer Look at the Role of Interruptions in Organisational Life. *Academy of Management Review*, July.28(3).

Kennedy, E. & Daiml, T. U., 2010. A strategy to assist management in workforce engagement and employee retention in the high tech engineering environment. *Evaluation and Program Planning*, 33(4).

Kerzner, H., 2017. *Project management metrics, KPIs, and dashboards : a guide to measuring and monitoring project performance*. Third Edition ed. Hoboken, New Jersey: Wiley.

Korkala, M. & Abrahamsson, P., 2007. *Communication in Distributed Agile Development: A Case Study*. s.l., IEEE.

Kothari, C., 2004. *Research Methodology - Methods & Techniques*. Second Revised Edition ed. New Delhi: New Age International Publishers.

Liang, T.-P., Liu, C.-C., Li, T.-M. & Lin, B., 2007. Effect of team diversity on software project performance. *Industrial Management+Data Systems*, 107(5), p. May.

Nerur, S., Mahapatra, R. & Mangalraj, G., 2005. Challenges of Migrating to Agile Methodologies. *Communications of the ACM*, 48(5).

Oshri, I., Fenema, P. V. & Kotlarsky, J., 2008. Knowledge transfer in globally distributed teams: the role of transactive memory. *Information Systems Journal*, 18(6).

Overby, S., 2019. *Agile Project Management: 10 Challenges you'll face*. [Online]

Available at: <https://enterpriseproject.com/article/2019/10/agile-project-management-10-challenges?page=0%2C1%2C0>

[Accessed 10 December 2021].

P.Hui, P. & Beath, C. M., 2001. *The IT Sourcing Process: A Framework for Research*. Austin, US, s.n.

Panteli, N. & Sockalingam, S., 2005. Trust and conflict within virtual inter-organizational alliances: a framework for facilitating knowledge sharing. *Decision Support Systems*, 39(4).

-
- Pattit, J. M. & Wilemon, D., 2005. Creating High-performing Software Development Teams. *R&D Management*, September.35(4).
- Pikkarainen, M. et al., 2008. The impact of agile practices on communication in software development. *Empirical Software Engineering: An International Journal*, 23 05.13(3).
- Pimentel, J. L., 2019. Some Biases in Likert Scaling Usage and its Correction. *International Journal of Sciences: Basic and Applied Research (IJSBAR)*, 45(1).
- Popli, R. & Chauhan, D. N., 2013. Research Challenges of Agile Estimation. *Journal of Intelligent Computing and Application*, December.7(1).
- Power, K. & Conboy, K., 2015. s.l., IEEE.
- Project Management Institute and Agile Alliance, 2017. *Agile Practice Guide*. First ed. Newtown Square, Pennsylvania: Project Management Institute.
- Project Management Institute, 2013. *The High Cost of Low Performance: The Essential Role of Communications*, Pennsylvania, US: Project Management Institute: Pulse Of The Profession In-Depth Report.
- Rajkhowa, A., 2020. *Blog: Agile Principle#12*. s.l.:Blog:Medium.com.
- Ramasubbu, N. & Balan, R. K., 2012. *Overcoming the Challenges in Cost Estimation for Distributed Software Projects*. s.l., IEEE.
- Richter, I., Raith, F. & Weber, M., 2016. *Problems in Agile Global Software Engineering Projects especially within Traditionally Organised Corporations*. Porto, Portugal , Association for Computing Machinery, New York, United States.
- Rising, L. & Janoff, N. S., 2000. The Scrum software development process for small teams. *IEEE Software*, August.17(4).
- Rockart, J. F., 1979. Chief executives define their own data needs. *Harvard business review*, 57(2).
- Rogers, T., 2019. *Vendor Management Best Practices : Using RACI Charts to Strengthen Third-Party Management*. [Online]
Available at: <https://vendorcentric.com/single-post/using-raci-charts-to-strengthen-third-party-management/>
[Accessed 25 January 2022].
- Rosenau, M. D. & Githens, G. D., 2005. *Successful Project Management: A step-by-step approach with practical examples*. Fourth ed. Hoboken, New Jersey: John Wiley & Sons, Inc.
- Rosenau, M. D. & Githens, G. D., 2006. *Successful Project Management: A step-by-step approach with practical examples*. Fourth Edition ed. Hoboken, New Jersey: John Wiley & Sons, Inc.
- Rubin, K. S., 2012. *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. First ed. Michigan, US: Addison Wesley.
- SAFe Scaled Agile Framework, 2021. *Agile Teams*. [Online]
Available at: <https://www.scaledagileframework.com/agile-teams/>
[Accessed 19 November 2021].
- Schaufeli, W. B. & Bakker, A. B., 2004. Job demands, job resources, and their relationship with burnout and engagement: a multi-sample study. *Journal of Organisational Behaviour*, May.25(3).
- Schwaber, K. & Sutherland, J., 2020. *The 2020 Scrum Guide™*. [Online]
Available at: <https://scrumguides.org/scrum-guide.html>
[Accessed 19 November 2021].

-
- Senapathy, Y., 2021. *How to Drive Project Success in Multi-vendor Environment*. [Online]
Available at: https://www.linkedin.com/pulse/how-drive-project-success-multi-vendor-environment-yad/?trk=public_profile_article_view
[Accessed 18 November 2021].
- Serra, F. R., Scafuto, I. C., Vils, L. & Mattos, M. S., 2021. Skills and Project Managers: Relationship Between Personal Characteristics and Performance Indicators. *The Journal of Modern Project Management*, August.9(1).
- Shrivastava, S. V. & Rathod, U., 2015. Categorization of risk factors for distributed agile projects. *Information and Software Technology*, February.58(1).
- Slack, N., Brandon-Jones, A. & Johnston, R., 2013. *Operations Management*. Seventh Edition ed. Harlow, United Kingdom: Pearson Education Limited.
- Stevens, K., 2012. *Getting weighted averages with the Likert question*. [Online]
Available at: <https://crowdsignal.com/2012/06/12/getting-weighted-averages-with-the-likert-question/>
[Accessed 22 January 2022].
- Struckenbruck, L. C., 1998. *Project Management Handbook: Integration-The essential function of Project Management*. Second ed. s.l.:John Wiley & Sons, Inc.
- Talby, D. & Dubinsky, Y., 2009. *Governance of an Agile Software Project*. s.l., IEEE Computer Society.
- Tarwani, S. & Chug, A., 2016. Agile Methodologies in Software Maintenance: a systematic review. *Informatica (Ljubljana)*, 40(4).
- UpGrad KnowledgeHut, 2019. *5 Problems in Agile and how to solve them | 5 Challenges in a Scrum team | Scrum Implementation*, s.l.: KnowledgeHut.
- Verwijls, C., 2017. *10 Powerful Strategies to Break Down Product Backlog Items: How to improve refinement and deliver more value to stakeholders sooner*. [Online]
Available at: <https://medium.com/the-liberators/10-powerful-strategies-for-breaking-down-user-stories-in-scrum-with-cheatsheet-2cd9aae7d0eb>
[Accessed 15 December 2021].
- Vidruska, R., 2014. *When should one switch to the 10-item Likert scale?*. s.l.:ResearchGate.
- Walsh, K. R., Mahesh, S. & Trumbach, C. C., 2021. *Agile Scrum Implementation and its longterm implication on Organisations*. First Edition ed. Hershey: Pennsylvania: IGI Global.
- Weakliem, D. L. & Frenkel, S. J., 2006. Morale and Workplace Performance. *Work and occupations*, August, 33(3), pp. 8-20.
- Whittaker, B., 1999. What went wrong? Unsuccessful Information Technology Projects. *Information Management & Computer Security*, March.7(1).
- Wiesche, M., 2021. Interruptions in Agile Software Development Teams. *Project Management Journal*, 52(2).
- Wilson, C., 2017. *Bruce Tuckman's Team Development Model*. [Online]
Available at:
https://d1wqtxts1xzle7.cloudfront.net/55992495/Bruce_Tuckmans_Forming_Storming_Norming_Performing_Team_Development_Model-with-cover-page-v2.pdf?Expires=1644322554&Signature=F9ixwPxdBaEdsCiw7O1916GoDb0JPstrJEXpKkzM3-

~nejvP~aJMmWaLFLrjr-uPwgGORTgkPq50nx

[Accessed January 28 2022].

Wysocki, R. K., 2014. *Effective Project Management : Traditional, Adaptive, Extreme*. Seventh ed. Indianapolis, Indiana: John Wiley & Sons, Inc..

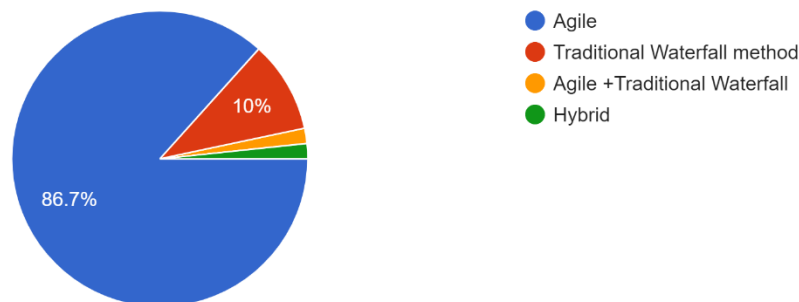
Yague, A., Garbajosa, J., Diaz, J. & Gonzalez, E., 2016. An exploratory study in communication in Agile Global Software Development. *Computer standards and interfaces*, November.48(1).

Appendix A

Original survey response from all the 60 respondents before the responses that does not meet research criteria were excluded

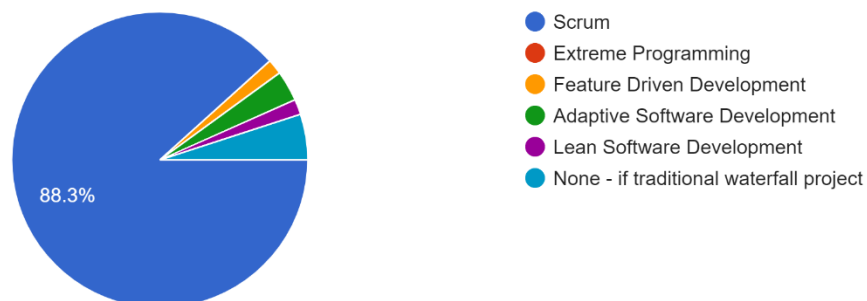
Which project management method is adopted by your project?

60 responses



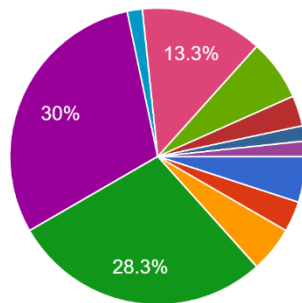
What is the Agile Methodology used in your project

60 responses



What is your Role in the Project Team?

60 responses

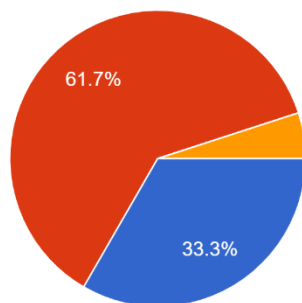


- Project Manager
- Scrum Master
- Product Owner
- Developer
- Tester
- Designer
- Business Analyst
- Other Manager - Test, Build, Design etc.

▲ 1/2 ▼

Which Organisation do you represent?

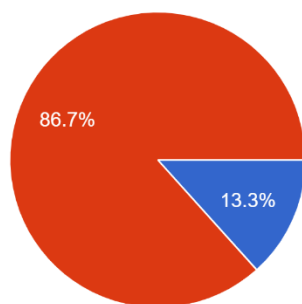
60 responses



- Client Organisation that owns the project
- Vendor Organisation
- 3rd party Contractor

Distribution of Project team

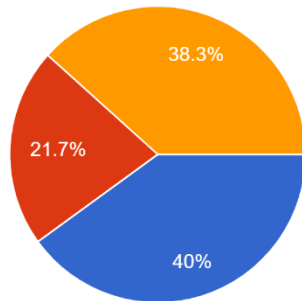
60 responses



- Project team is locally based
- Project team is distributed globally

Which of these location categories do you belong to?

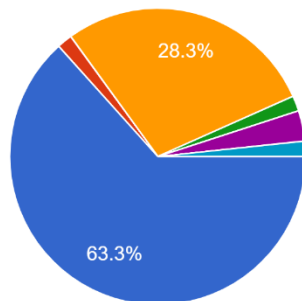
60 responses



- Located onsite same location as client and other vendors
- Located offsite (different location but same country as client/other vendors)
- Located offsite (different country)

Where are you located?

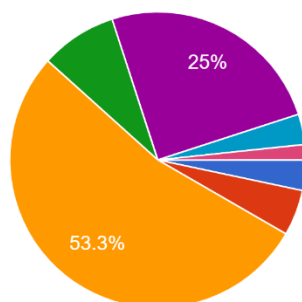
60 responses



- India
- Asia (countries other than India)
- UK
- Europe (countries other than UK)
- North America
- Australia
- Latin America
- African Countries

Where is your client organisation located?

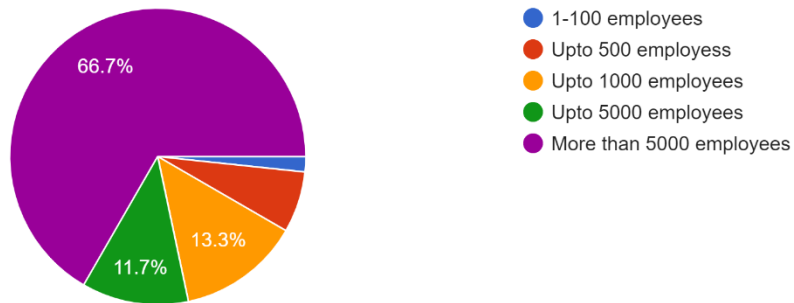
60 responses



- India
- Asia (countries other than India)
- UK
- Europe (countries other than UK)
- North America
- Australia
- Latin America
- African Countries
- N/A (if you belong to client organisation)

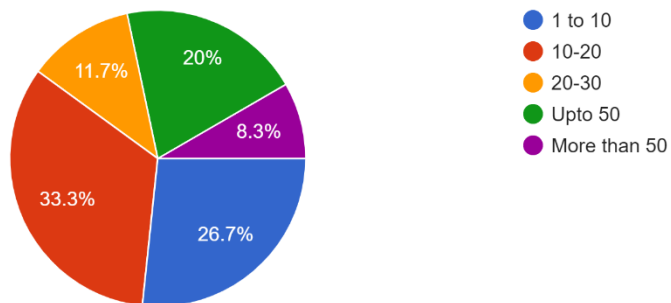
What is the size of your organisation?

60 responses



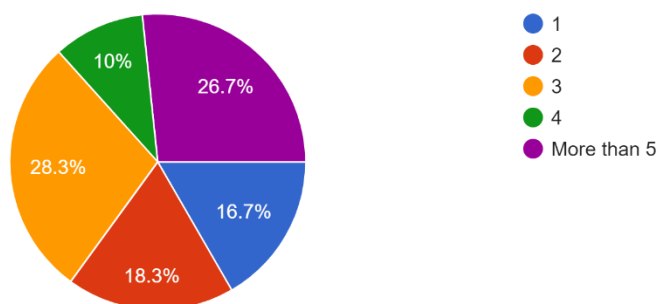
What is the size of your project team?

60 responses



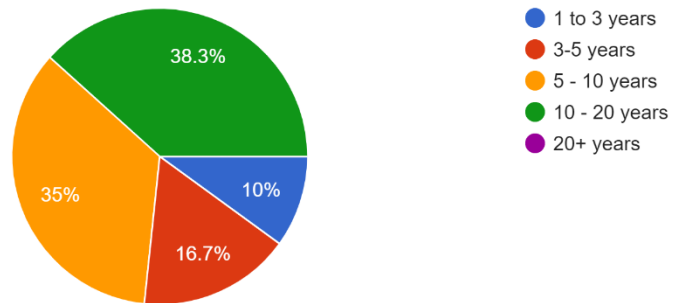
How many vendor organisations work for your project?

60 responses



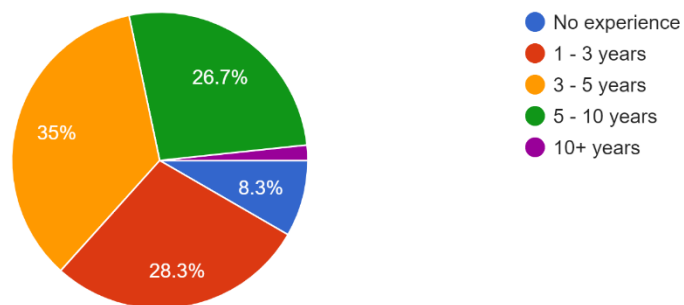
What is your experience in IT industry?

60 responses



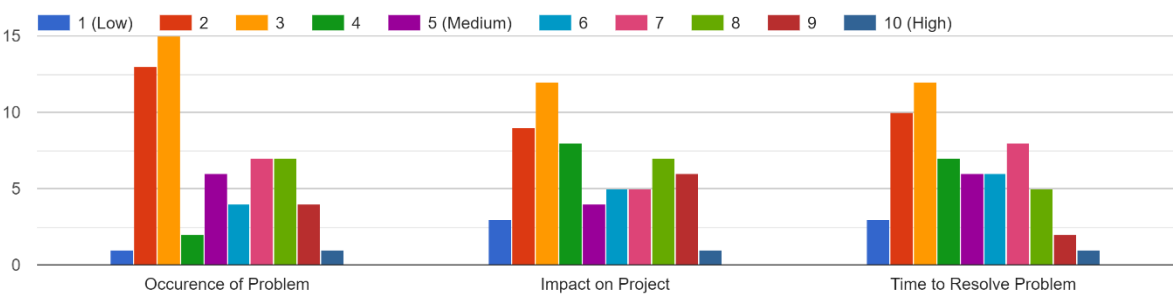
How long have you worked in Agile projects?

60 responses



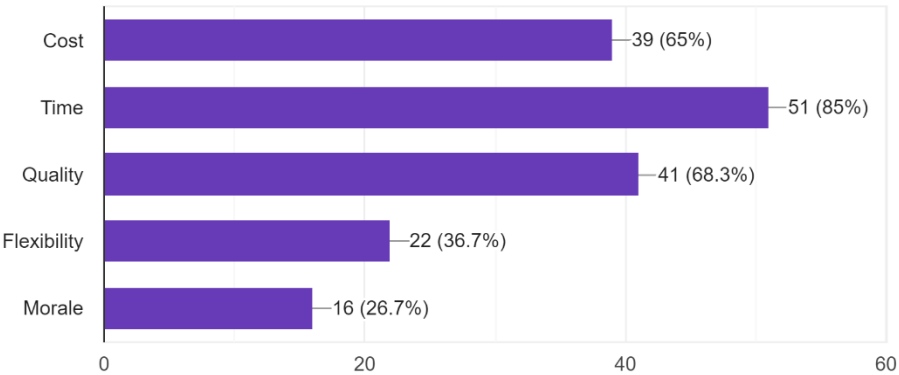
Problems in Agile Projects and its Impact on Project Performance

Lack of clear boundaries defining scope and tasks often lead to one vendor stepping into other vendors domain

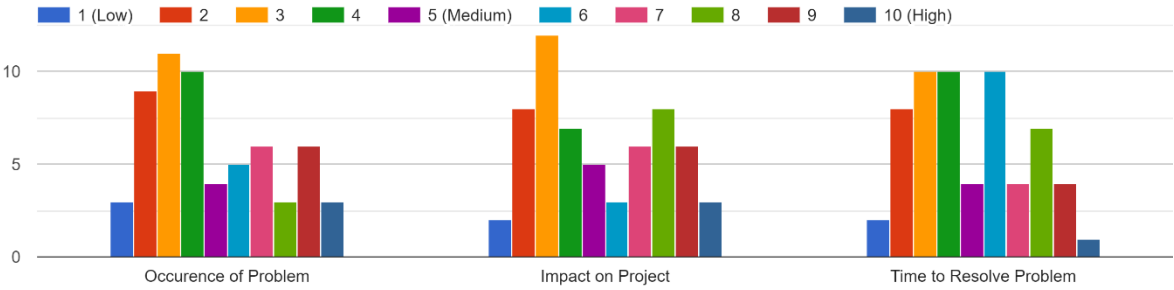


Which of these Project Performance factors are impacted due to the previously rated problem?

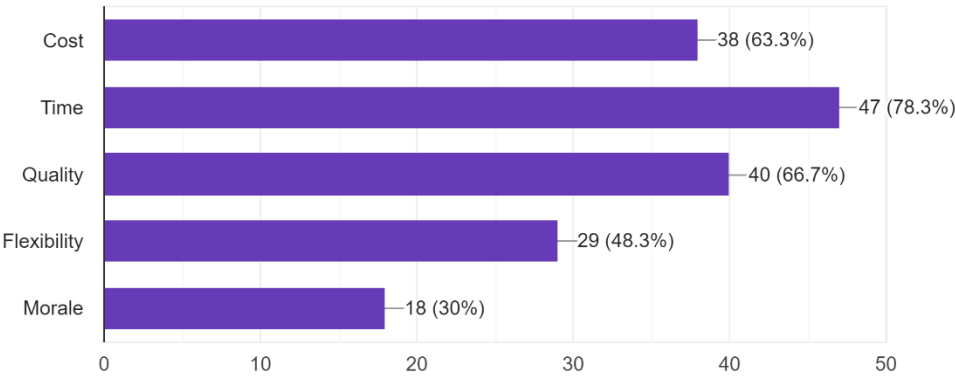
60 responses



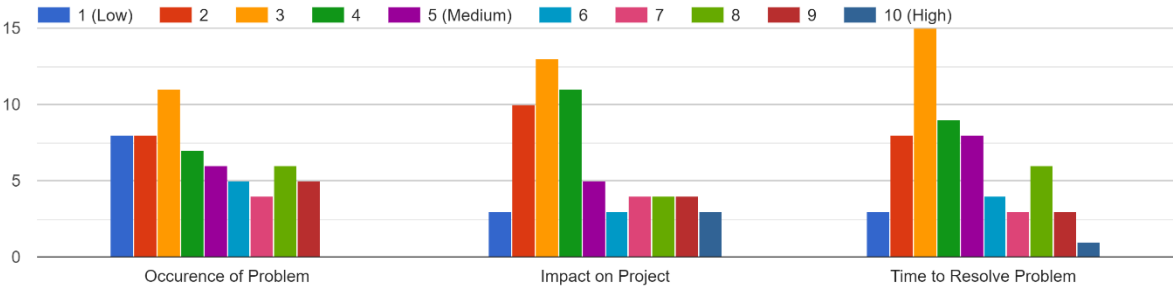
Problems such as changing priorities, changing scope and conflicts with resource availability by teams within the project and interfacing teams outside project prevents tasks from being completed within sprint timelines



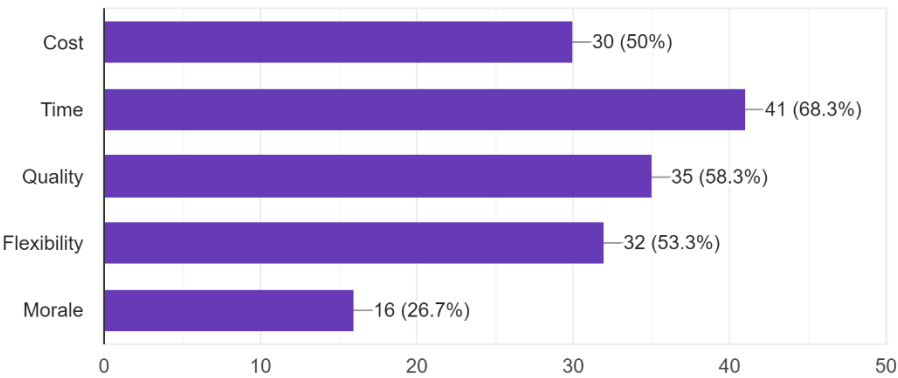
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



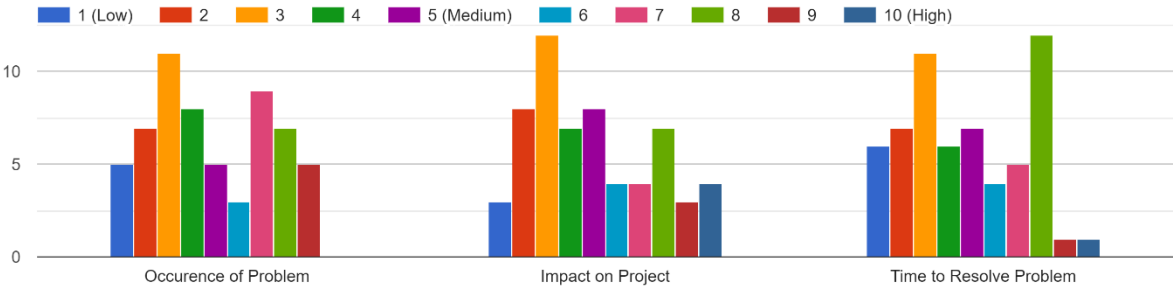
Difficulties in translating agile process, methods, values and metrics to non-agile or traditionally organised management teams, such as while seeking support for client-v...esolution; status reporting to a non agile stakeholder etc.



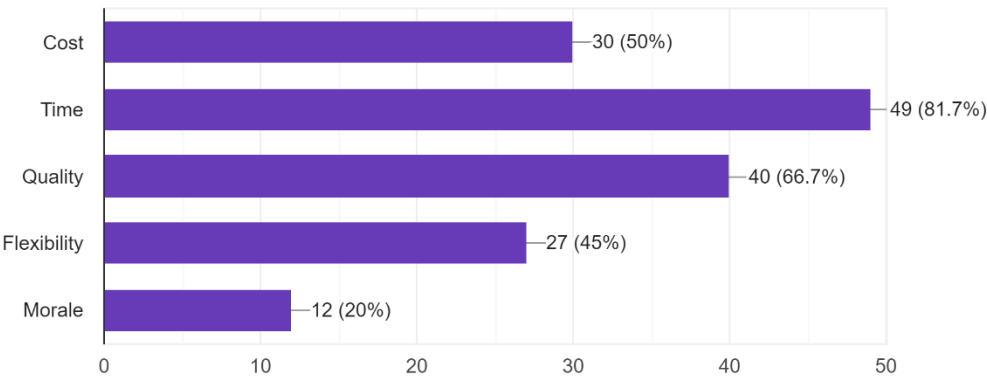
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



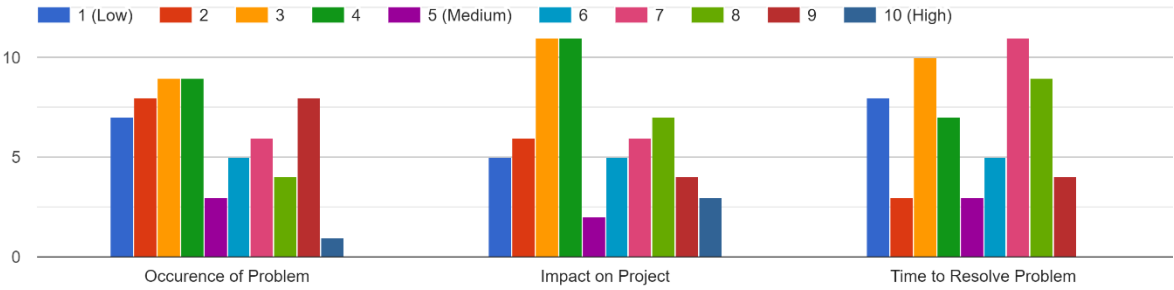
Estimation and Planning are mostly performed by co-located teams at client site, usually by experienced team member representing each functional area - build, test etc...nced or remote team members leads to slippage of schedule



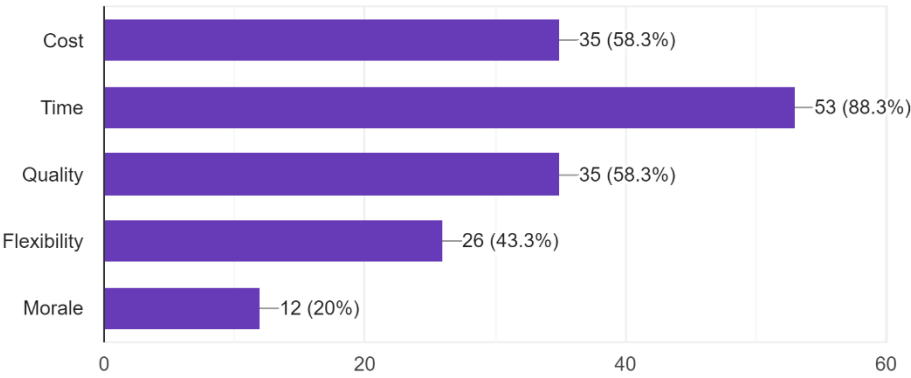
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



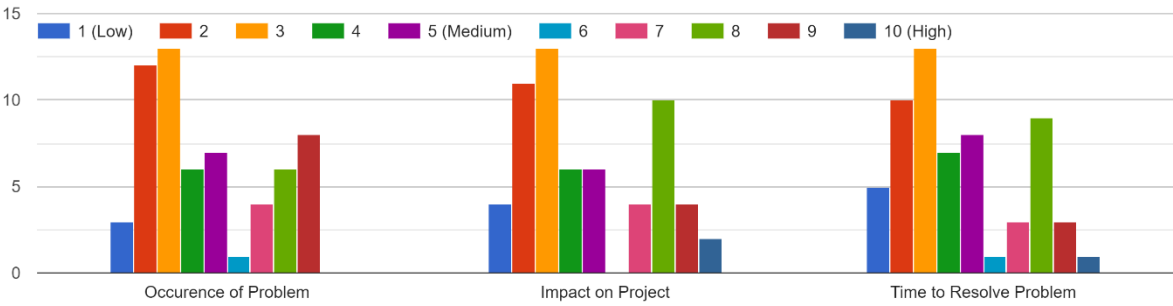
Joint estimation/planning in agile scrum teams are often influenced by other vendors, management and clients resulting in user stories estimated lower than it would be required results in cost and schedule overrun



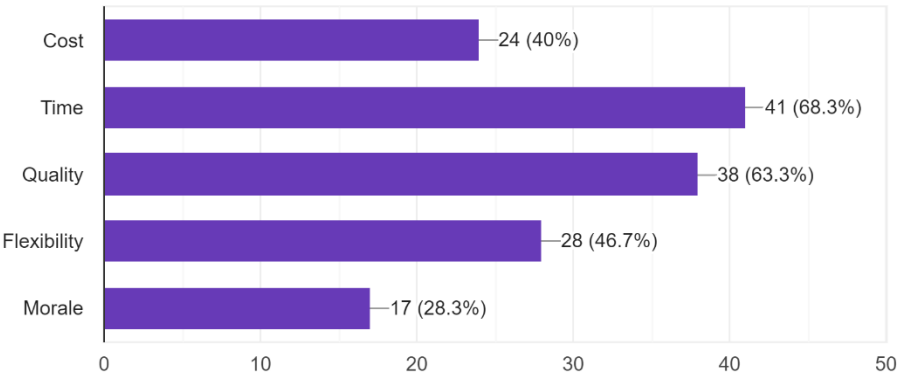
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



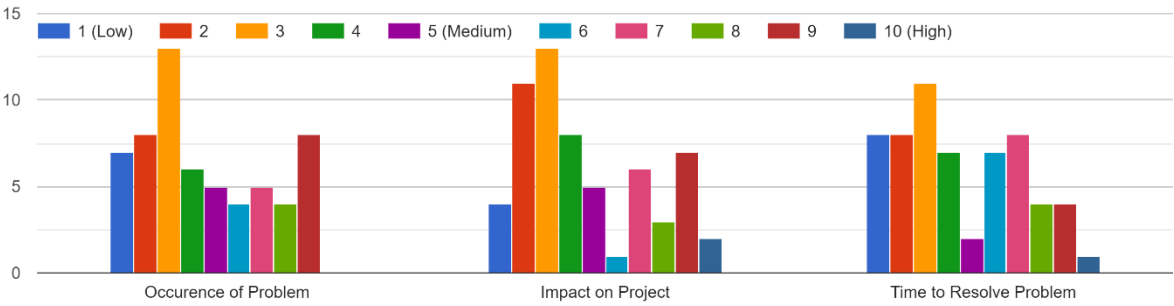
Lack of direct communication between vendor teams and communication often facilitated through clients results in awareness and co-ordination problems



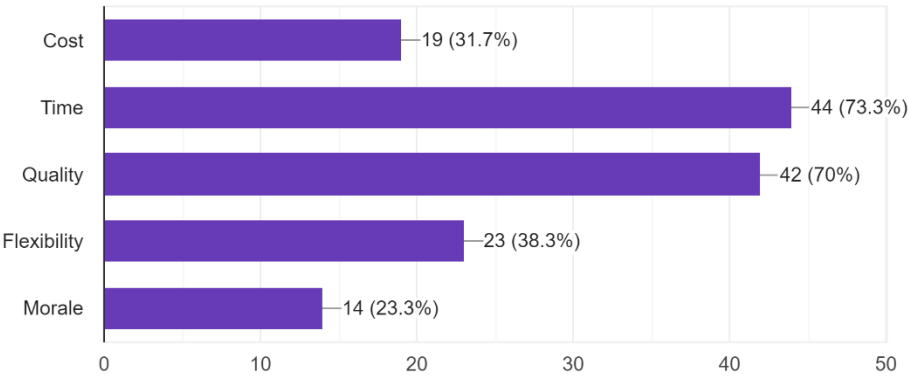
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



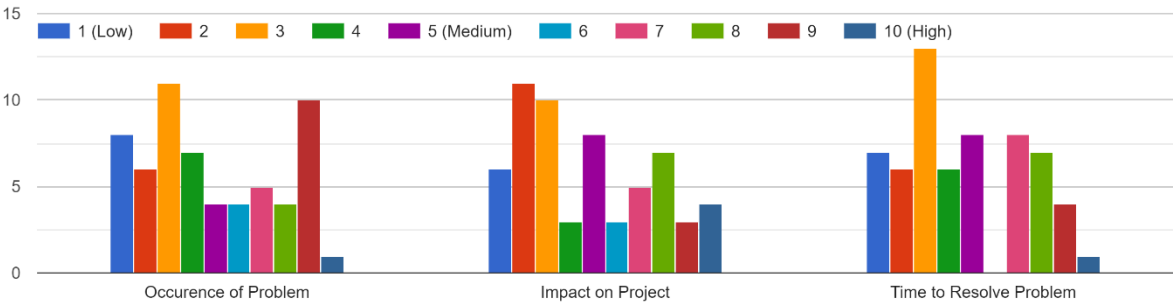
Sheer size of the team, i.e., more number of communication links within the team leads to deterioration in communication quality or information exchanged



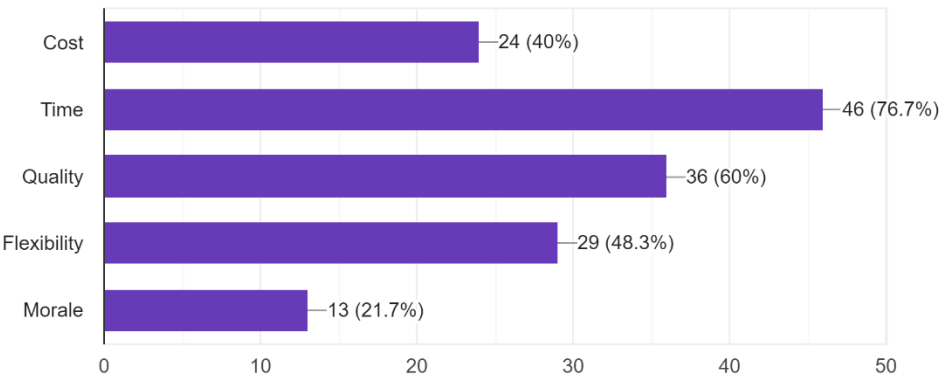
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



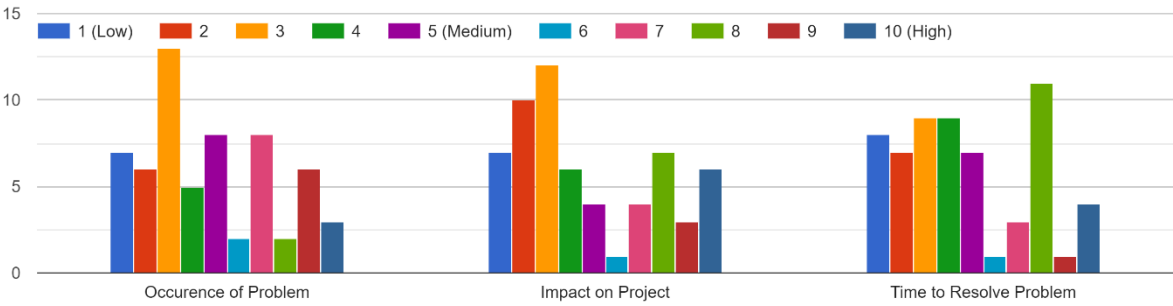
Lack of proper communication tools, less overlapping time due to different time zones, different cultural and language background are obstacles for remote teams to effectively co...ate in meetings with counterparts from different locations



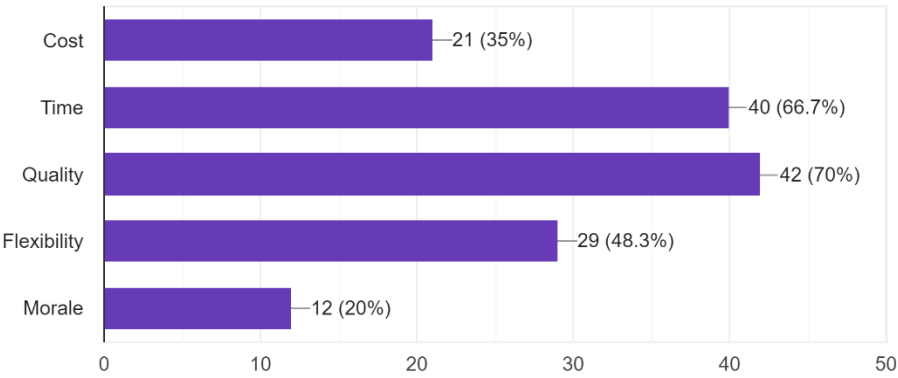
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



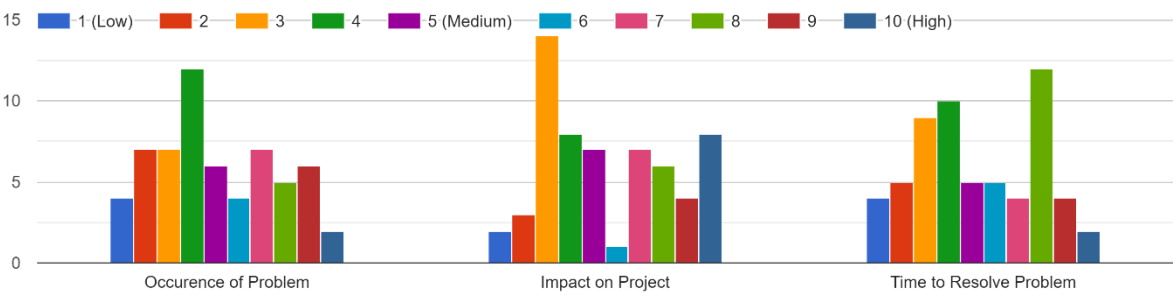
Online meetings are not as effective as compared to face-face meetings due to distractions from parallel working and communication, considerable time and difficulty to explanations, wait for approval/feedbacks, lack of non-verbal cues



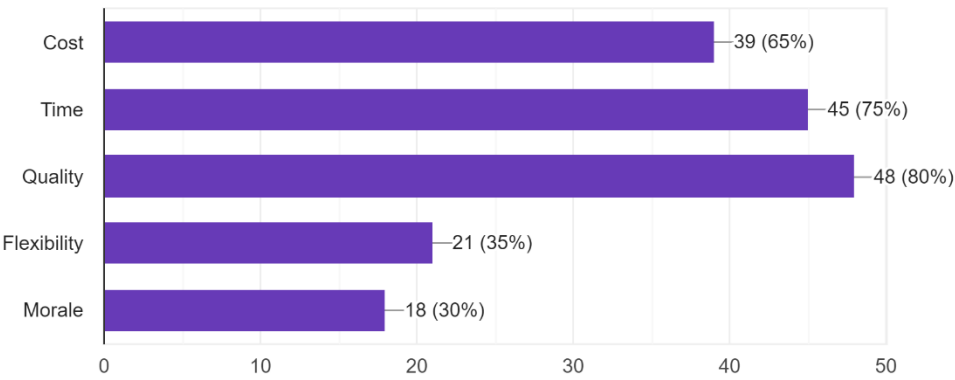
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



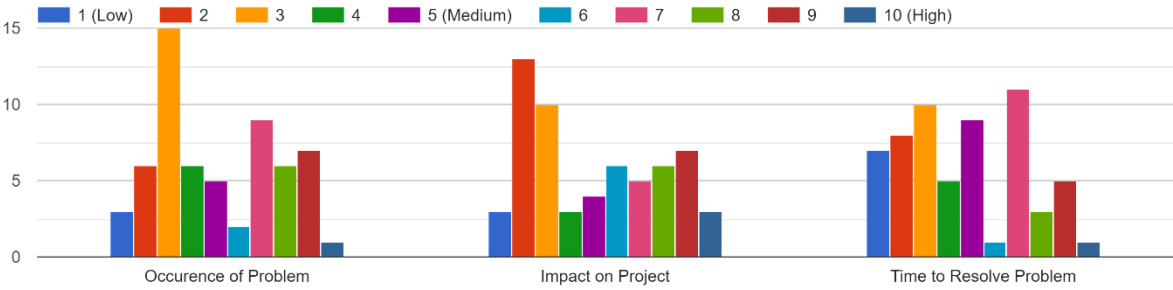
Frequent changes in requirements or user story especially towards the end of a sprint causes excessive delays, re-planning and rework



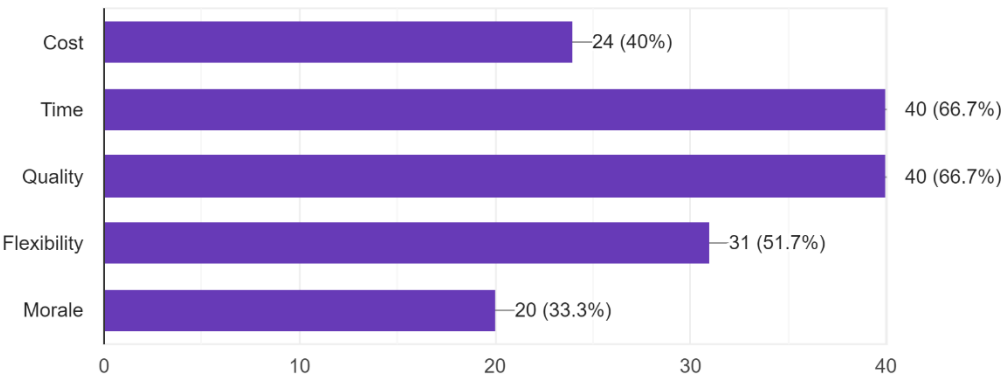
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



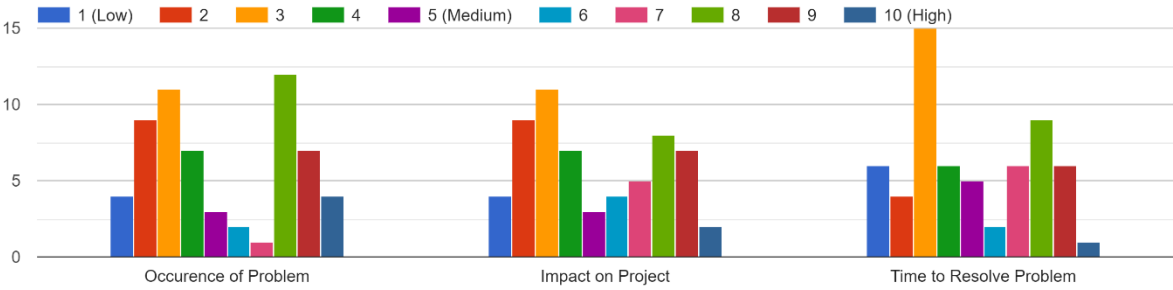
People often allocated to multiple projects/teams on shared basis results in discontent and stress personally and interruption to team members seeking input on overlapping urgent issues



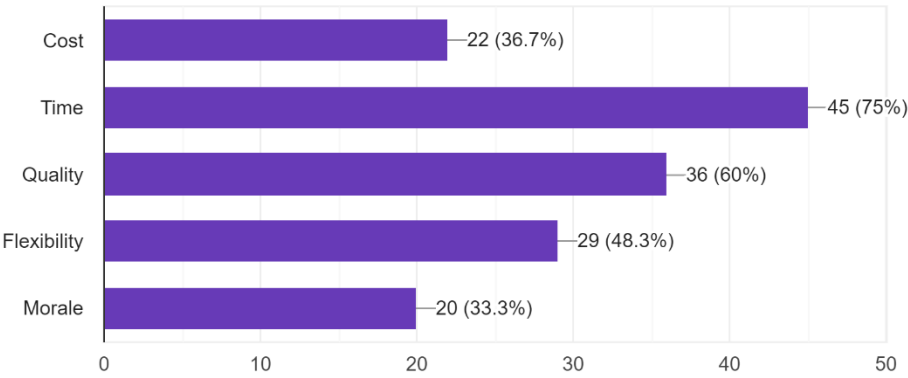
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



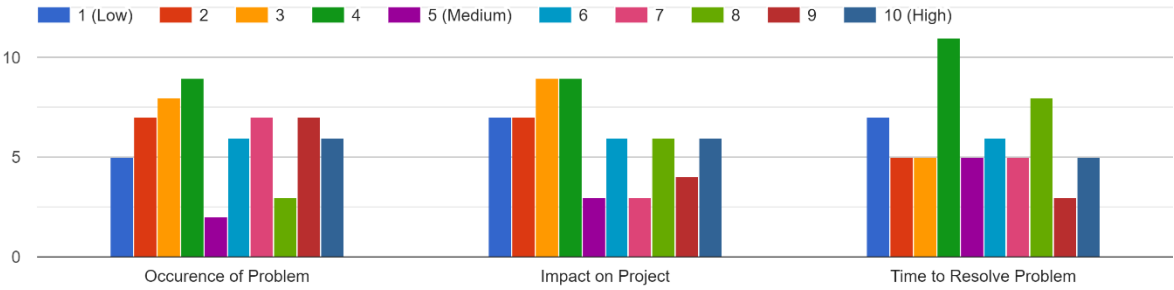
Continuous interactions from peers onsite or offsite, clients and vendor teams seeking update on project progress, help or information on work processes, interrupts the primar...entually leads to working over time to meet the sprint goals



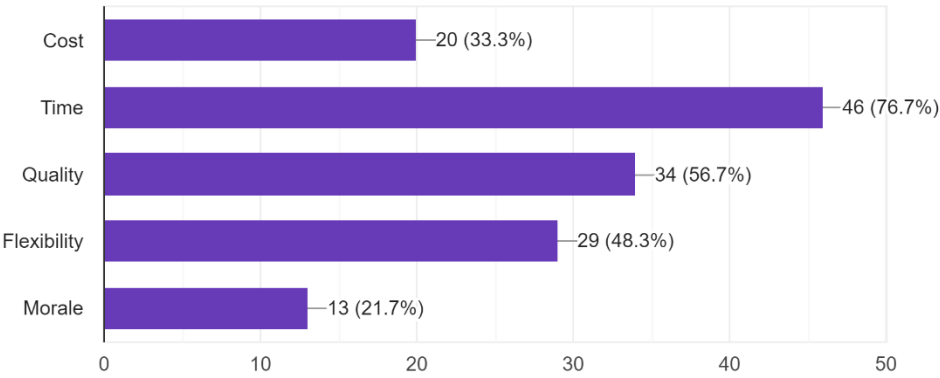
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



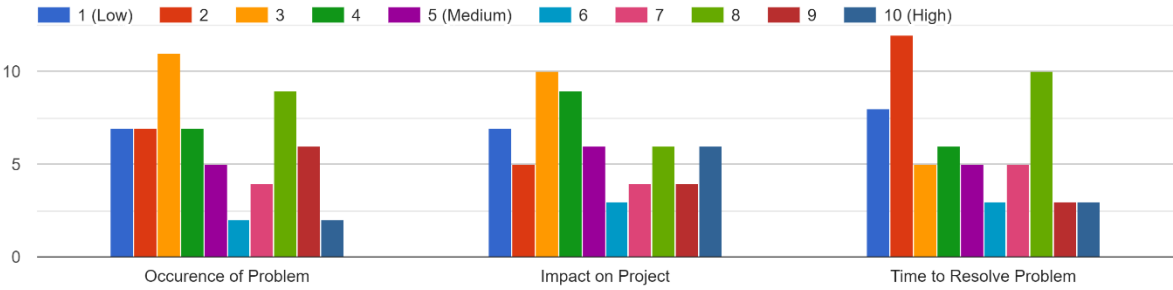
High frequency and excessive time spent on meetings and agile ceremonies such as daily stand-ups, retrospectives, sprint reviews etc reduces the time available for developme...ally leads to working over time to compensate the time lost



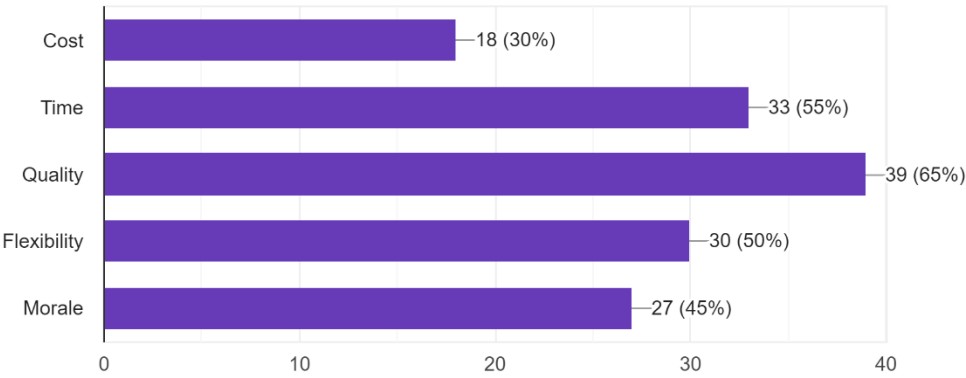
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



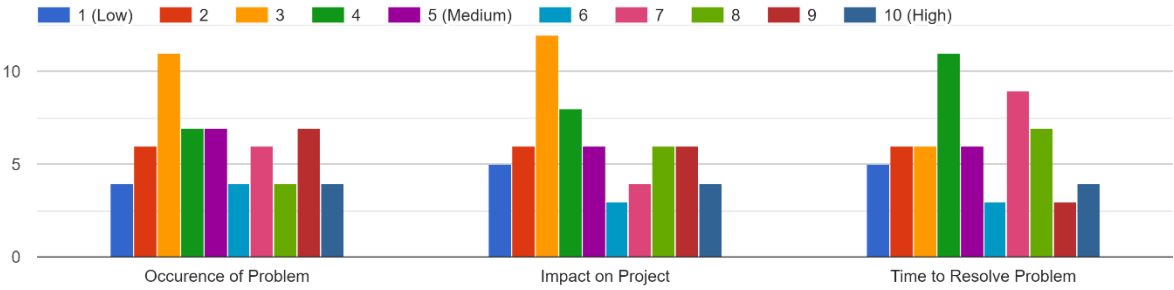
Sense of differentiation towards members from other vendors, competition, power struggles and politics by vendor teams to influence clients preference to their side often leads to conflicts, lack of trust and collaboration in the project



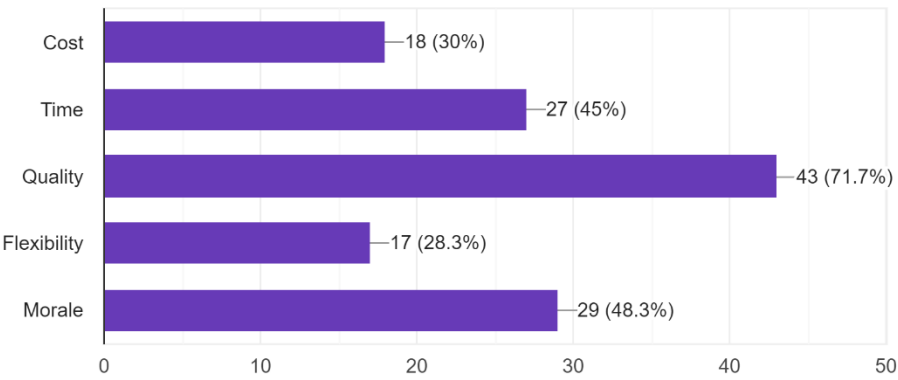
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



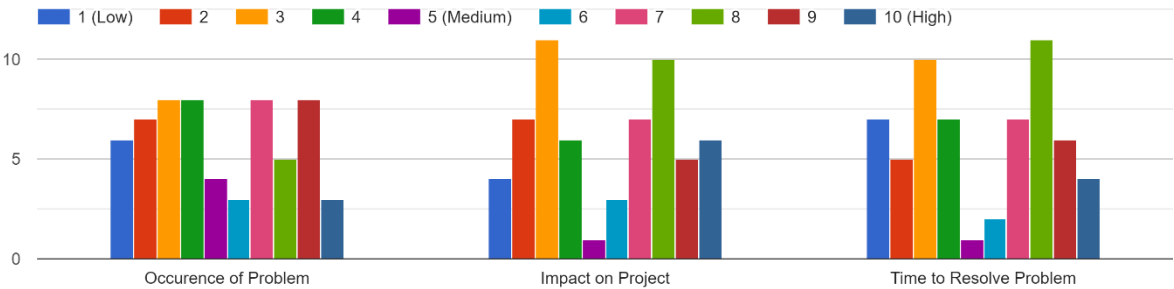
Absence of hygiene factors at work such as lack of advancement, growth, recognition of achievements by management or client leads to demotivation and reduced productivity



Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses

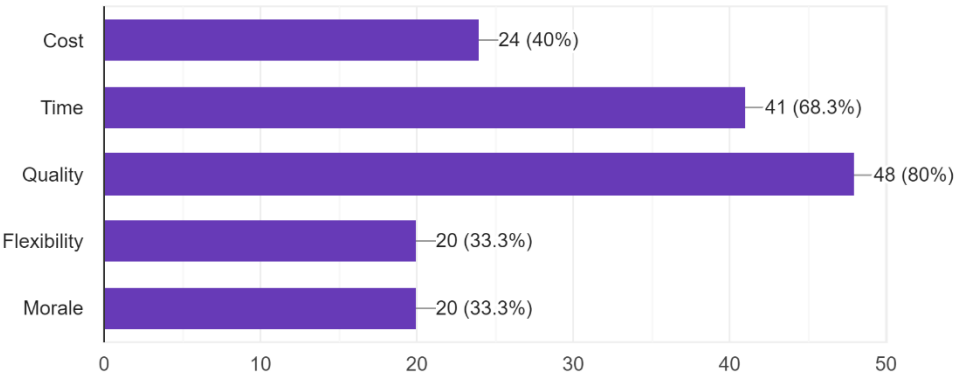


Constantly changing resources or adding new resources in the project affects team's chemistry and velocity

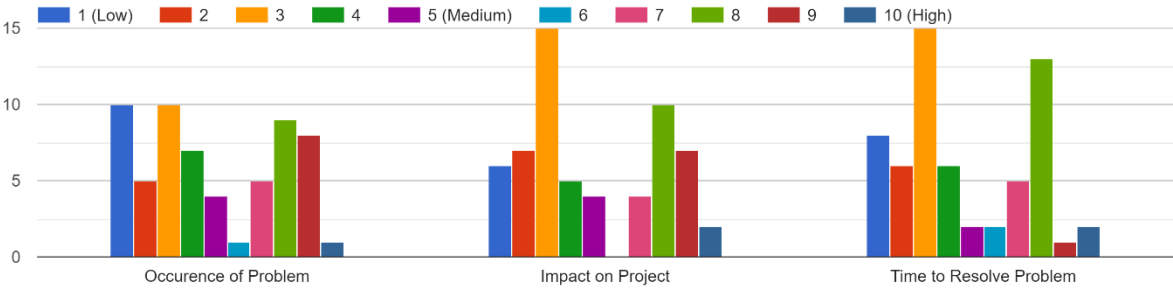


Which of these Project Performance factors are impacted due to the previously rated problem?

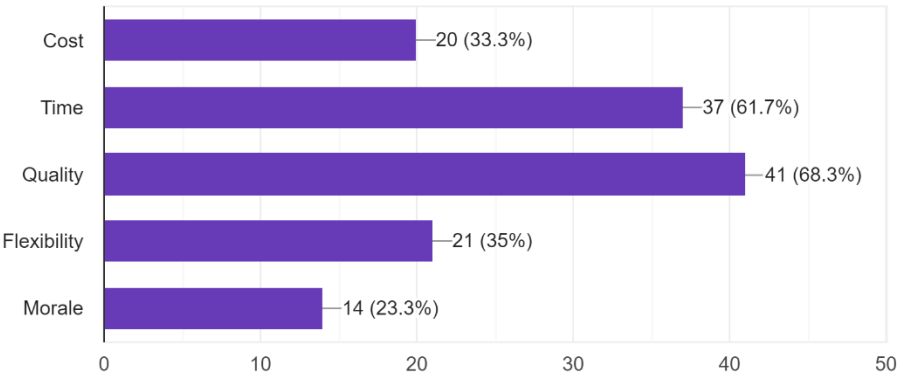
60 responses



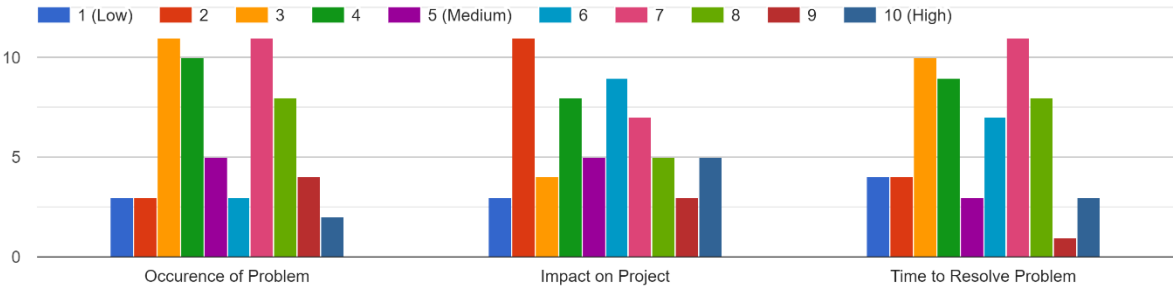
Concerns about sharing tools, knowledge or proprietary information such as development tools, stubs etc., created by one vendor with another



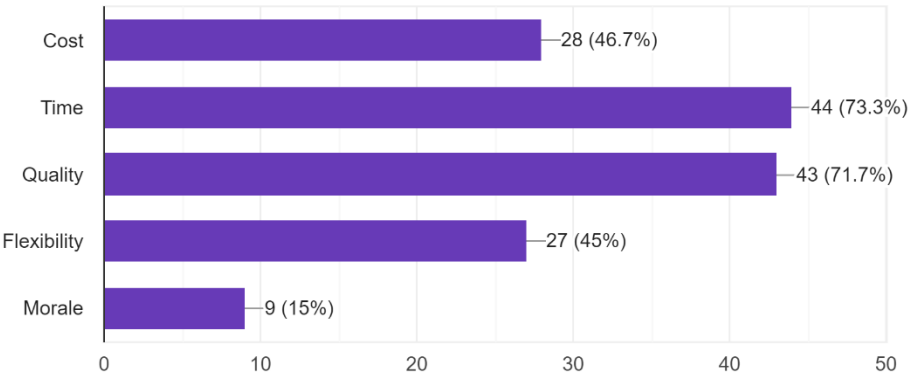
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



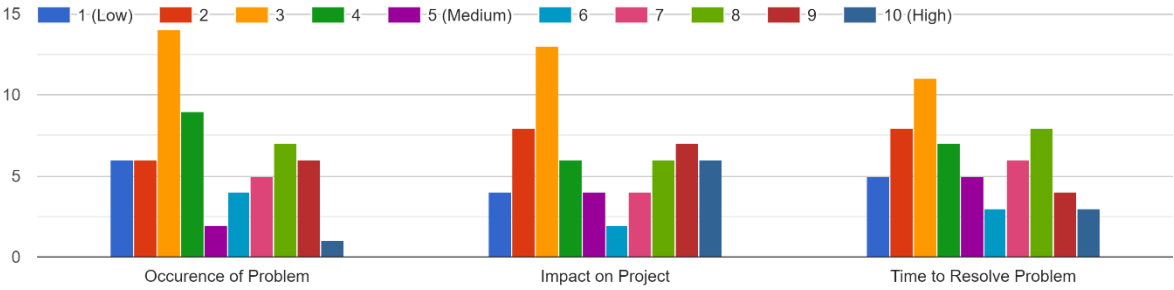
Inability to breakdown backlog into smaller user stories and prioritise based on most business value that can be delivered within a sprint 'or' horizontal breakdown of storie... working software or business value at the end of each sprint



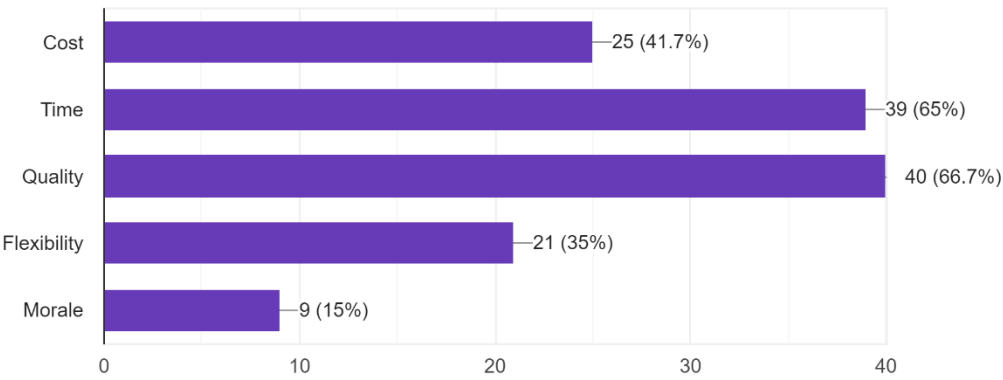
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



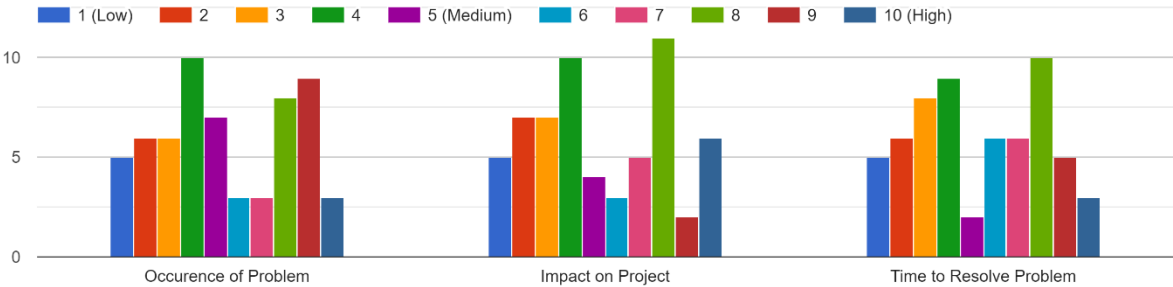
Lack of continuous Client/Business User involvement due to distributed nature of the project, such as communicating to business and getting feedback on matters



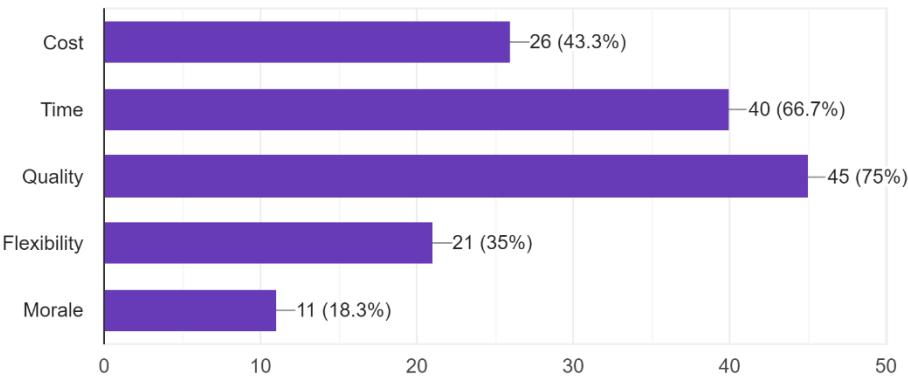
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



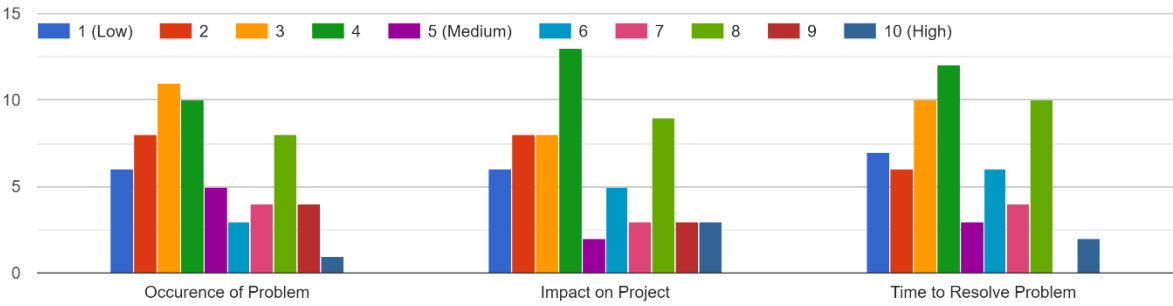
Lack of different skillsets, appropriate tools or restricted access to tools hampers the team to be fully cross-functional and makes the team heavily reliant on component teams in possession of necessary skills or tools



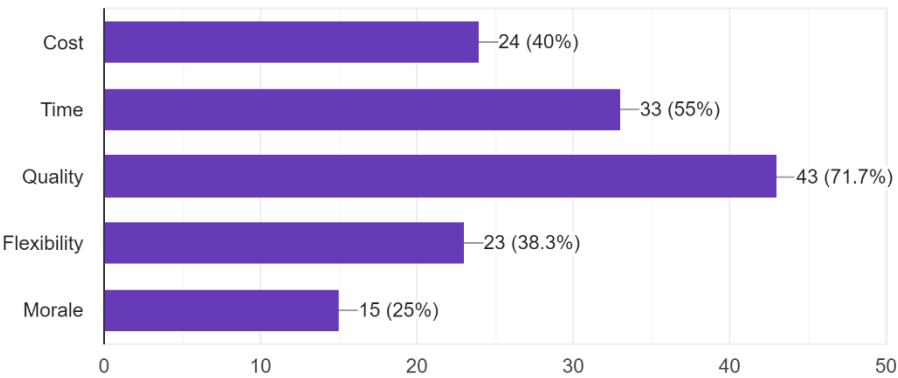
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



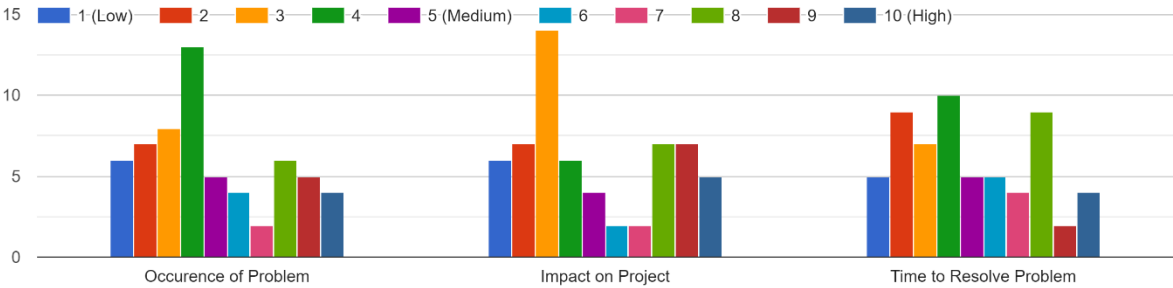
Lack of clarity around roles or specialised roles such as testing manager, build master, project manager etc., can sub-optimize efficiency, control and smooth team function... setup where resources directly report to a client manager



Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses

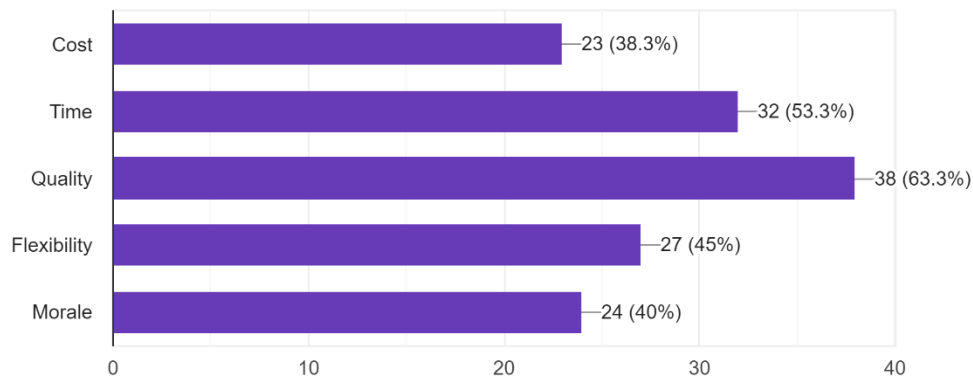


Lack of Leadership/management support and commitment at different levels of management - client, vendor etc.

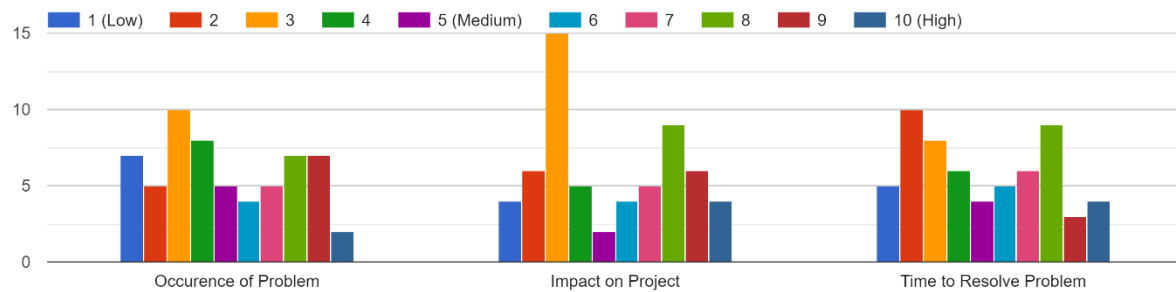


Which of these Project Performance factors are impacted due to the previously rated problem?

60 responses

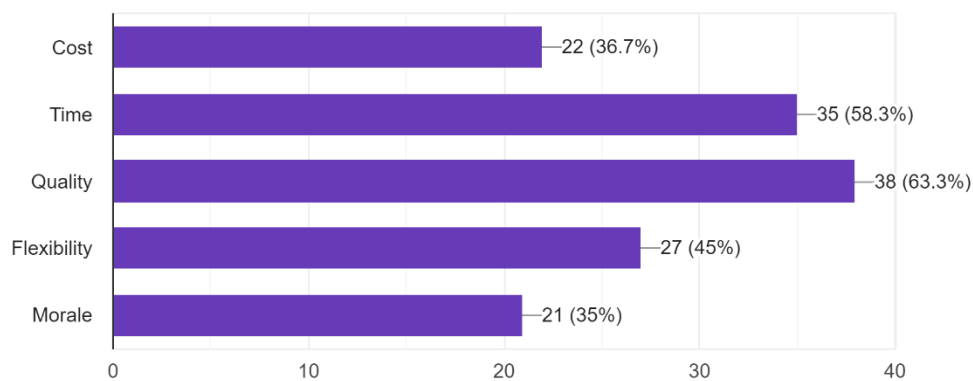


Lack of project management competence or appropriate leadership style for agile projects by project manager or scrum master, i.e., project manager adopts an autocratic leadership style over all decisions and little input from group members

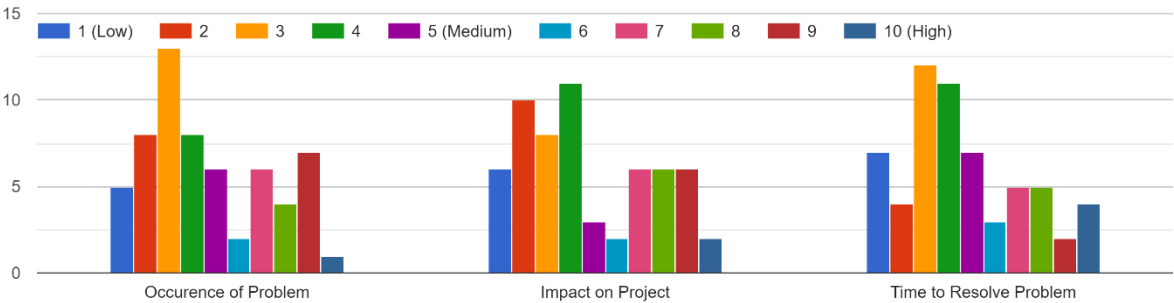


Which of these Project Performance factors are impacted due to the previously rated problem?

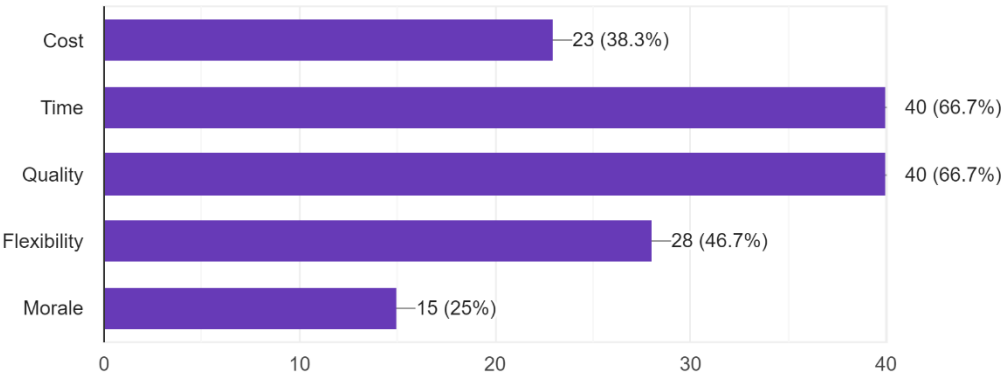
60 responses



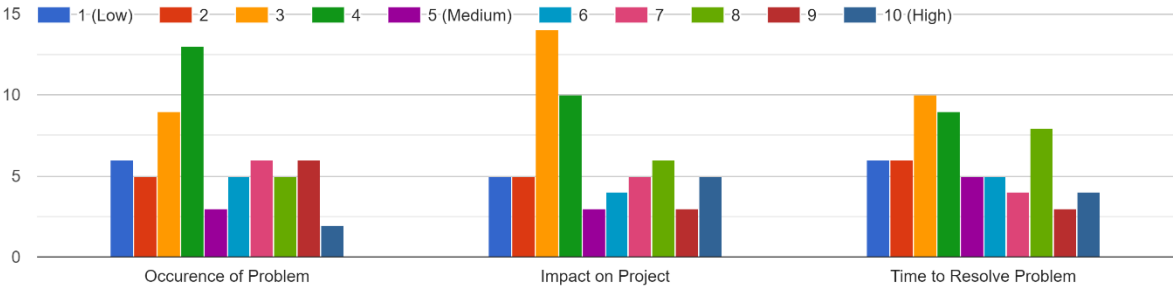
Resistance/lack of acceptance of agile principles or wrong understanding/incorrect application of agile approaches, such as, management orders to work agile but the team resist...ed is not modular or suitable for development in iterations



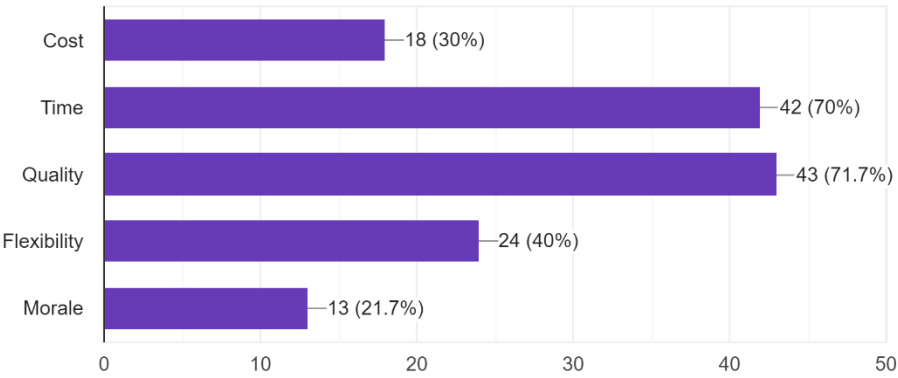
Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



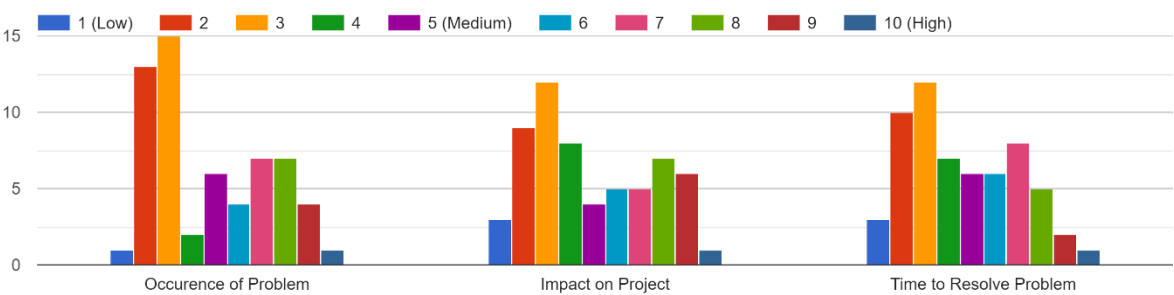
Lack of governance and documentation leads to problems with measuring and monitoring the performance, knowledge deficiency etc.



Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses

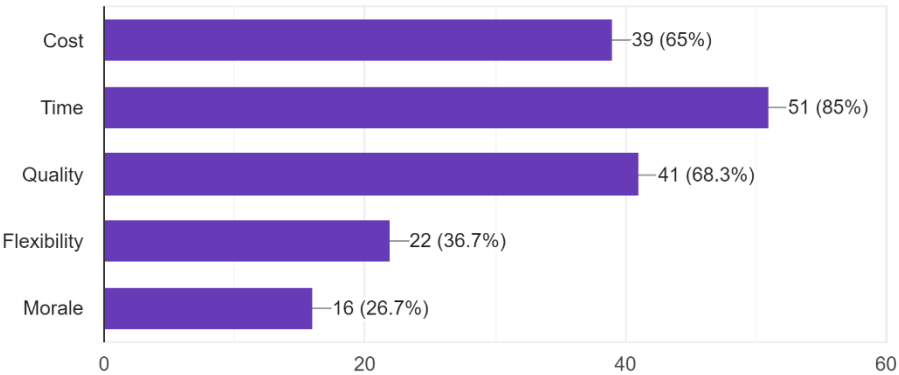


Lack of clear boundaries defining scope and tasks often lead to one vendor stepping into other vendors domain

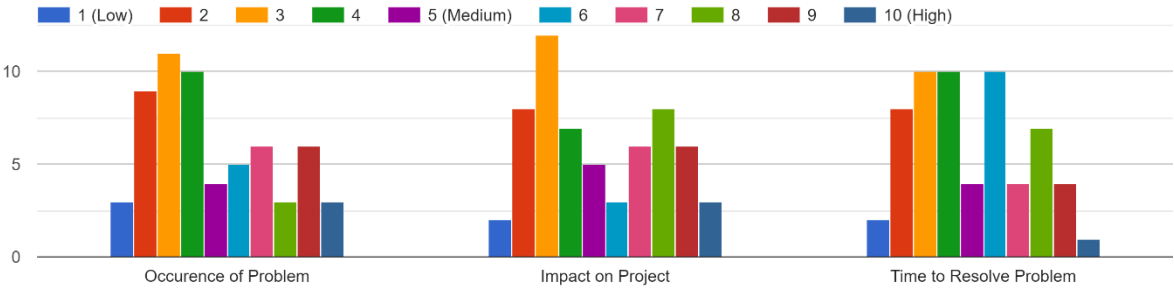


Which of these Project Performance factors are impacted due to the previously rated problem?

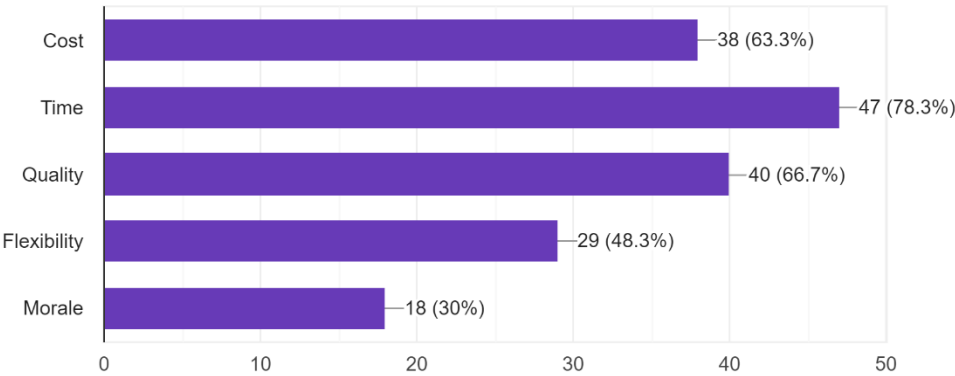
60 responses



Problems such as changing priorities, changing scope and conflicts with resource availability by teams within the project and interfacing teams outside project prevents tasks from being completed within sprint timelines



Which of these Project Performance factors are impacted due to the previously rated problem?
60 responses



Appendix B



Survey Analysis.xlsx