DOI: https://doi.org/10.46632/daai/2/2/1

Data Analytics and Artificial Intelligence
Vol: 2(2), 2022
REST Publisher; ISBN: 978-81-948459-4-2
Website: http://restpublisher.com/book-series/data-analytics-and-artificial-intelligence/

Examining Effectiveness of Performance Analysis of Agile Transformation

[1]*Parkha Kaul1, [1]Parmanand, [2]Shruti Aggarwal
[1]Sharda University, Uttar Pradesh, India
[2]Thapar Institute of Engineering and Technology, Patiala, Punjab, India
*Corresponding author Email: Kaulparkha@gmail.com

Abstract. The agile manifesto established the agile method as a response to the conventional paradigm in software development. In this manifesto, new ideals for the software industry were examined, as well as new principles for the organization's agile foundation. To keep up with the rapid changes in technology and business, agile methodologies can be a useful tool. Many firms have switched from using plan-based production processes to using agile ones because of the advantages of agility. For this research roughly 10 software and communications firms are identified where agile is being used. All the respondents for interviews and survey's will be completely functioning in agile environment. The entire sample size was aimed at 150 with an emphasis on reaching the proper people to acquire reliable information. This will be done by adopting the single-staged sampling approach where agile organizations are picked. Later SPSS will be used for additional analysis of the data. Surveys will largely focus on questions to identify the efficacy of present yearly performance evaluation techniques/methods.
Keywords: Agile, transformation, software, performance, employee.

1. Introduction

An agile approach to work helps businesses transition from inflexible to resilient. Learn about the advantages of agile and how it can stimulate digital age innovation and growth. Collaboration and adaptability are hallmarks of the Agile method of working. Iterative cycles and cross-functional teams tear down organizational borders, allowing for rapid feedback and progress. When working in an agile environment, you may accomplish tasks more quickly while also narrowing down on a specific problem. Agile is a methodology that originated in software development but has found widespread application. It has been shown to double time to market and boost customer satisfaction by 10% to 20%. Achieving agile transformation requires a fundamental shift in an organization's attitude from the inflexible, bureaucratic thinking that characterized most organizations in the twentieth century to the flexible, post-bureaucratic thinking that will prevail in the future. For this to happen, a culture of speed, adaptability, and employee empowerment must be established. Teamwork is no longer just limited to product managers or owners, but may be applied throughout a huge organization's many divisions when agile transformation is implemented. There have been a variety of approaches to software development during the past four decades. As business practices evolve at a quick pace, so do the programming needs. As a result, old software development methods, also known as heavyweight, are being replaced by more current approaches, known as agile or lightweight. Instead, then focusing on techniques, agile approach focuses on the importance of people as a driving factor for project success. When it comes to Agile, it emphasizes the significance of greater communication between developers and consumers, as well as a stronger working relationship. A portion of the justification behind this is that acquiring entire requirements before to a project is nearly impossible, and that modification and user feedback are essential and should be accepted throughout the project's lifespan. The most significant and most difficult way to maximize the value of knowledge use is through the exchange of knowledge among members of an organization. When working in a dispersed environment, it is impossible to depend exclusively on tacit knowledge, which is the focus of agile techniques. What happens if the company's primary knowledge-bearers leave without having their knowledge and expertise documented and managed explicitly? It's impossible to use agile principles in a dispersed situation without documentation and clear 2 knowledge, thus combining these two issues is a big difficulty. Researchers and practitioners have been concerned about these possible issues and incompatibilities virtually from the inception of the agile movement. "Distributed agile" may be an oxymoron since it represents two competing and conflicting methods," says the author of the article. When used together, however, they provide a "best of both worlds" solution with the potential for higher productivity, considerable cost savings and better IT-business alignment. Distributed agile is intriguing because of the mix of challenges and possibilities it presents. "Agile Information Radiator" will help organizations working with agile methodologies improve their distributed software processes by exploring issues related to distributed agile development from a knowledge perspective in the software engineering environment so as to help the organizations working in agile methodologies improve their distributed software processes, thus bridging the gap between these two incompatible methodologies," the authors say. The purpose of this study is to analyze the effectiveness of annual performance appraisal system in an agile transformation to develop efficient framework.

There are 12 principles of agile software development:

- Ensure client satisfaction by consistently providing high-quality software.
- Regardless of how early or late in the project a change in requirements occurs, always accept them.
- During the course of a project, developers and business experts must collaborate closely on a regular basis. Information is best shared in person.
- By fostering an atmosphere of admiration, trust, and empowerment, you may encourage individuals to work on a project.
- Developing software that really works is the most important metric for evaluating development progress.
- The agile method encourages long-term growth.
- The agility of technological development and design is boosted by a constant focus on excellence and quality.
- Effective agile management necessitates a focus on simplicity.
- The greatest architecture, requirements, and design are produced by self-organized teams.
- To be more successful, teams should reflect and modify via inspection and review.

Emergence of Agile Methodologies: In recent years, the discipline's identity has become ambiguous as the use of agile approaches has increased. When the methodology movement fails, agile approaches are typically held up as a panacea. Methodologies such as agile aim to change the course of software development, which has been on a downward spiral for the previous 40 years. There has been a long-held assumption in the software development community that strict discipline and rigour are the best ways to ensure that the process is both predictable and efficient. The Standish Group (an American research business) claimed in 1994 and 1999 that the software crisis was still very much alive, therefore all of the CASE tools, techniques, and quality-oriented models sought to solve this (The Standish Group 1994). According to agile developers—proponents of agile software development—projects often go awry despite the deployment of sophisticated managerial and technology breakthroughs because software professionals have sought to foresee an activity that is always evolving and evolving and evolving. A software development project may become more predictable if risks are addressed throughout the project by using ways that effectively enable human adaptability. An SDM is a written set of rules, methods, and procedures that software development teams often employ to better the software development process in terms of greater IT productivity and higher quality final IT solutions" (Chan and Thong, 2009). The debate concerning agile vs conventional methodologies is addressed in a significant segment of the literature (see, for example, Glass, 2001; Beck and Boehm, 2003; DeMarco and Boehm, 2002). However, the parallels between the plan-driven and the agile method can only be observed at the beginning and conclusion of the development cycle. When a problem arises, either within the company's internal systems or as a result of a new business opportunity, the concept for a software solution to address the issue is hatched. The ultimate product of either technique should be a fully functional piece of software. However, the approach used to achieve the intended end-result may differ. In 1998, the term "agile" was first used in conjunction with "software process" (Aoyama, 1998). Working in cross-functional teams that take responsibility for their own organisation is encouraged by Agile Methodologies (AM). Such teams are made up of people who can fill all of the duties. Agile firms routinely employ staff rotations to keep things fresh (Chau et al., 2003). Agile Methods are a response to the "demand for an alternative to documentation-driven, heavyweight software development techniques" that have been in place for decades (Beck et al., 2001). In the past, needs were elicited and documented, and then architectural and high-level design, development, and inspection were carried out. It was very uncommon for practitioners in the mid-1990s to be frustrated and perhaps convinced that these early stages of growth were just not feasible (Highsmith, 2002). A frequent observation among practitioners was that their practise had steadily shifted away from very document- and process-focused development techniques to more people- and less document-cantered approaches, which led to the creation of agile methodologies.

2. Agile Project Management Philosophy

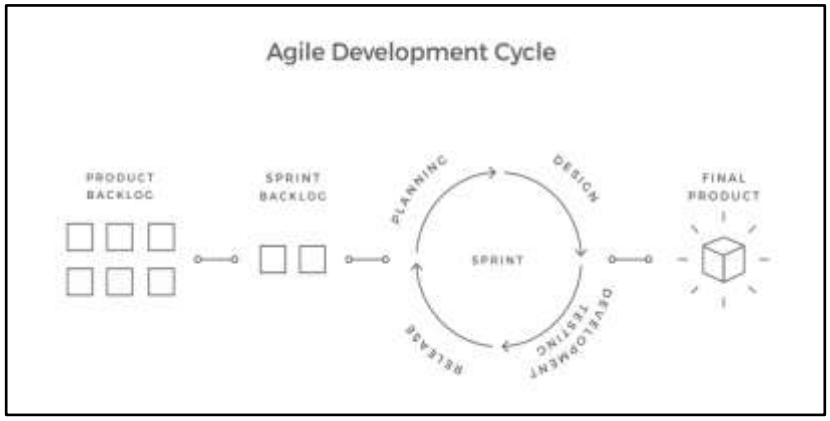
Software engineering may now be made more adaptable and efficient by implementing the Agile project management philosophy instead of the more conventional method. In a short period of time, it has become the de facto standard for managing projects. Agile has been embraced in some form or another by 95% of firms, according to some estimates. In the meanwhile, much effort has to be done in order to bring the practise to maturity. Founded in 1957, Agile has a long and distinguished history. When Bernie Dimsdale, John von Neumann, Herb Jacobs, and Gerald Weinberg were producing software for IBM and Motorola at the time, they were employing incremental development approaches (which are now known as Agile). In spite of the fact that they had no idea how to categorise their new method of software development, they were able to discern that it was distinct from Waterfall in many respects. After 17 software developers convened in 2001 to examine alternate project management approaches, the modern-day Agile methodology was born. Agile software development was outlined in the manifesto for agile software development, which had a clear vision of the technique. The Manifesto, which aims to "discover better ways of producing software," explicitly outlines the new approach's essential concepts:

"*Through this work we have come to value:*

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation

- Responding to change over following a plan."

The Twelve Principles of Agile Software have made the idea a ubiquitous and efficient approach to project management. Iterative software development is a hallmark of agile techniques. Agile projects, as contrast to a linear Waterfall methodology, are broken down into smaller cycles called Sprints. They all have backlogs, implementations, testing and deployments within the pre-defined scope of work for which they were created.



Deliverable product increments are provided at the end of each Sprint. Since the product grows incrementally with each iteration, so does the project. The odds of producing a possibly flawed product are greatly reduced if the features are tested early in the development process. Respondents to the 14th annual status of Agile research report on the following advantages of an Agile approach:

- flexibility to adapt to shifting priorities (70%);
- greater team productivity through daily work allocation (58%);
- and improved project visibility owing to the straightforward planning methodology. (65 percent).
- Agile's flexibility and speed of implementation are still seen as important advantages by businesses.

3. The Benefits of Agile Software Development

Agile meetings were created by top software professionals. They intended to build a more effective method for assessing project development after frequently experiencing difficulties and constraints with standard waterfall development in real-life projects. The strategy they employed directly tackles concerns about established techniques' beliefs and procedures.There are numerous possibilities for true involvement between the team and the stakeholders at each sprint meeting in agile. All parties engaged in the project are working together at all times since the client is actively involved. So the team may have a better understanding of what the customer wants to achieve in this way. To build confidence with stakeholders, the team has to consistently produce high-quality, functional software. This also encourages the customer and the team to work together more closely.

Transparency: Customers are involved in all phases of the project — from planning to testing to implementing new features — thanks to the agile methodology. When a project's development is publicly shown, clients should be aware that they are viewing a work in progress rather than the finished result.

Predictable and On-Time Deliver: Every one to four weeks, sprints are held according to a predetermined timetable. Using a time-boxed approach, stakeholders can expect new features to be provided fast and regularly. If the programme has enough economic value, the team can also beta test or release it sooner**.**

Fixed Costs and a Predictable Schedule: As a result of the Sprints' predetermined timeline, the expenses are both set and predictable. The customer will have a better idea of the approximate expenses of each feature if the anticipated costs are combined before each Sprint. The ability to prioritise features and iterate on them in a more efficient manner is provided by this method.

Prioritization that can be changed at any time: By prioritising customer-driven features, Scrum techniques offer for greater flexibility. With each sprint boundary, the team gains more control over the shippable units of work, allowing them to push forward toward the final product milestone. A quick RIO from engineering requires that work be delivered to customers early so that they can see the value of the features.

Allows for Modification: An Agile methodology provides a chance for continuous product backlog refinement while focusing on delivering the agreed subset of features. The additional improvements can be implemented within a few weeks if these changes are incorporated to the next version.

Focuses on the Economic Benefit: Team members are more knowledgeable about the client's business and can supply features that are most beneficial to the business.

Focuses on the Clients: Product features are frequently defined in terms of business acceptability requirements using user stories. Each feature gives genuine value and is not merely an IT component since it focuses on the needs of the user. Beta testing the software at the end of each Sprint is a better way to get important input. As a result, valuable information may be gleaned early on in the process, allowing for necessary adjustments.

Enhances the Level of Excellence: It is simpler for the team to focus on high-quality development, testing, and cooperation if the projects are divided down into manageable chunks. Defects and mismatches may be detected and rectified early by making builds and doing tests or reviews throughout the cycle.

It offers your group a sense of direction.: Creating a shared feeling of ownership and objective for all team members is a common goal in agile procedures. Instead,then generating a fictitious feeling of urgency, this provides your team direction. Teams that have a clear goal are more productive because they always strive to improve their speed and efficiency.

4. Related Studies

Trippensee, Lennard &Remane, Gerrit. (2021). The use of agile approaches in software development projects with smaller teams has been around for a while now. In recent years, numerous companies have begun implementing agile practises throughout their whole business. Agile changes may be difficult and require a holistic perspective of cultural, operational, and strategic issues to be successful, though. Large-scale agile practises are reviewed in this document, which gives a comprehensive overview of many large-scale agile techniques. Eleven practises and 19 sample concepts were discovered in total across four practise areas. To help academics and managers better understand large-scale agile development, the categories have been arranged into a framework and serve as a suitable entry point.

Alsari, Abdulrahman (2020) Developers are creating increasingly sophisticated web and mobile applications as processing power rises. Software engineers must adhere to the software design life cycle as a standard practise in order to consistently provide high-quality software deliverables. There are problems with the traditional approach to project management that must be addressed. Team members that adhere to the agile ideals can significantly increase product quality using the more recent technique known as agile methodology. When it comes to educational and industrial settings alike, agile software development approaches like Scrum and Kanban provide a wealth of capabilities and flexibility. Agile approaches, in contrast to conventional ones, are better at responding to change, connecting with customers, and interacting with one other. Based on three essential modules, including design and learning methodologies, team-building, and profiling, we performed a critical survey. The results revealed that the survey raised the present position of the company and opened new research directions to incorporate Agility into existing systems.

Mircea, Elena. (2019). The expansion of the IT sector has brought us complicated technologies that go beyond the processing capability of the human brain. It was necessary for the society to adjust to the new values and principles because of the changes in daily operations and economic activities. Every individual has to learn new skills and look for new methods to execute the task they do every day. Furthermore, "Agile" is presented as an innovative style of thinking in a demanding environment, with all the concepts and practises that characterise it. Scrum and Kan-ban, two of the most prominent agile approaches and methodologies, will be compared and contrasted in this research to see how well they complement one another.

Saleh, Malik. (2011). Keeping software development projects focused and responsive to external circumstances is possible through the application of agile software development approaches. The most critical component in overcoming the difficulties is the techniques followed by developers. When a client is actively involved in the development process, it is said to be an Agile project. Too little client interaction is the biggest challenge faced by agile teams. The absence of collaboration between the developers is a necessary condition for an agile project. Changes in business and technology necessitate the use of agile approaches. In this article, we present a conceptual framework for adapting agile approaches to meet various constraints. Developing processes, customer collaboration, and the ability to foresee change are all part of the framework.

Methodology

For reaching our study aims, we have adopted a mix-mode method to have a comprehensive insight of the topic. By adopting the concurrent triangulation, data gathering will be done at the same time for surveys and interviews. Current performance assessment methodologies will be analyzed from agile environment viewpoint. For this research roughly 10 software and communications firms are identified where agile is being used. All the respondents for interviews and survey's will be completely functioning in agile environment. The entire sample size was aimed at 100 with an emphasis on reaching the proper people to acquire reliable information. This will be done by adopting the single-staged sampling approach where agile organizations are picked. Later SPSS will be used for additional analysis of the data.

Surveys will largely focus on questions to identify the efficacy of present yearly performance evaluation techniques/methods. It will also focus on recommendations for better performance evaluation in agile context. Survey respondents will be mostly working in agile environment and will include:

- **Team members** (Agile team members): Members of the team are the ones that are responsible for completing the duties and are rewarded for their hard work. Selection is based on an understanding of how and when each member of an agile team will be evaluated on an individual basis.
- **Scrum Master:** Keep track of agile techniques to speed the team's progress or motivate them. Keeping an eye out for any roadblocks and ensuring that the group doesn't overcommit itself are other important considerations. We'd want to see how the scrum master sees performance evaluation from his or her perspective.
- **Manager (Immediate):** Keep track of agile techniques to speed the team's progress or motivate them. Keeping an eye out for any roadblocks and ensuring that the group doesn't overcommit itself are other important considerations.

We'd want to see how the scrum master sees performance evaluation from his or her perspective.

- **Product Owner:** As the heart and soul of the project, product owners are responsible for keeping track of all of the project's needs and backlog. Learn about how he or she assesses the work of people and teams.

TABLE 1. Performance evaluation through survey questions and measurement

Survey Question	Measurement
1	Despite the system's shortcomings, there remains room for development.
2	An annual review of one's work helps to increase one's productivity.
3	Collaboration and teamwork are fostered through the performance evaluation process.
4	The company's strategy is supported by the performance assessment procedure.
5	Performance evaluations serve a useful role, and I fully get that.
6	An chance to have a better understanding of what I should be doing is provided through performance evaluation.
7	Growth and learning are made easier thanks to the evaluations.
8	Performance evaluations have given me a boost in confidence.
9	Using performance evaluations to help employees define and accomplish meaningful objectives can be beneficial.
10	The assessment process makes it crystal obvious what success looks like.

It was necessary to do statistical analysis on the first ten items to test hypothesis 1. The primary goal of conducting separate analyses is to see if we receive different answers for hypothesis 1 in each study. In addition to the closed-ended questions, a few open-ended questions were chosen to gain a deeper understanding of performance evaluation. Interviews with interviewees were recorded if the interviewee's consent was requested and the interviewee's replies were transcribed from the recordings. About 30 to 45 minutes was allotted for each interview.

Findings and analysis

This part will discuss how survey and interview data is analyzed.Figure.1 and Figure.2 shows the descriptive statistics of experience and job role.

TABLE 2. Job Experience of respondents

Choices	Totals
0-5	19
6-10	34
11-15	20
16 yrs	27

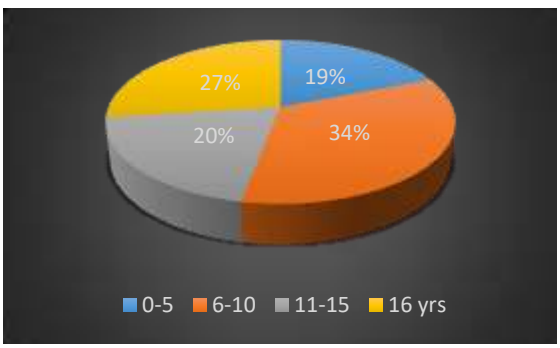


FIGURE 1. Job Experience of respondents

Figure 1 shows how many years of work experience people have gained. Those with less experience (those with 0 to 5 years of work experience) constituted a bare minimum of 19 people, whereas participants with 10 or more years of work experience constituted 35 people. Respondents with 11-15 years of experience have an average age of 20, while those with 16 or more years of experience have an average age of 27.

TABLE 3. Job Role of respondents in Agile

Choices	Totals
Team member	62
Scrum Master	17
Product Owner	7
Manager	15

FIGURE 2. Job Role of respondents in Agile

For job role (Figure 2), majority of respondents work as Team member (Developer or Test Engineer) which represents almost 61% of the population. The lowest population was for product owners and scrum master/manager have 17and16respondentsrespectively.InFigure3, all survey questions related to"Hypothesis1"are plotted to show the respondents responses. From these box plots, it also visible that there were no visible outliers in the data. The average response for each question is represented by is d line.

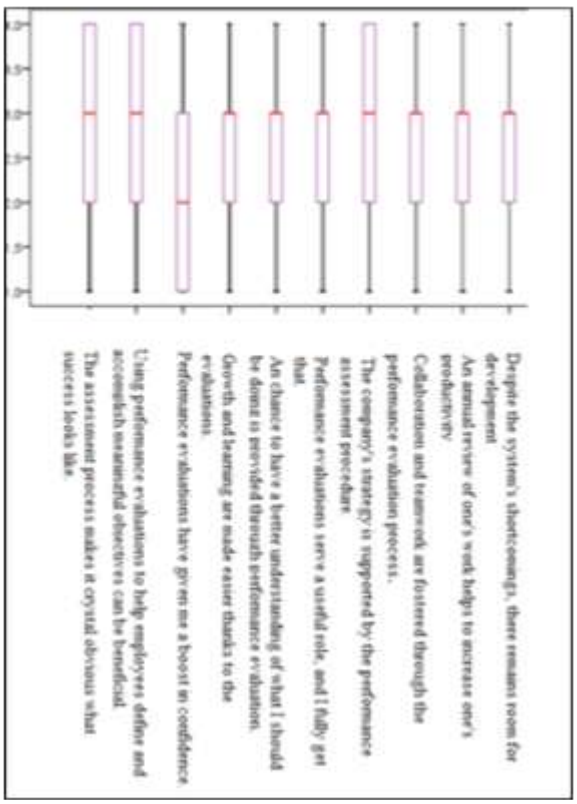


FIGURE 3. Agile Annual Performance Evaluation Representation in Box plot

H1: The current practices of annual performance appraisal are effective when applied to agile work environment.

"Anova (4*4)" was used to assess the effectiveness of yearly performance review in an agile work environment by analyzing data using two factors. Hypothesis 1 was tested using questions 1 through 10 from the survey. Annual Appraisal was the label given to all of these questions. "Annual Appraisal" is a dependent and factor variable in an analysis (experience and job role). Four levels of each factor variable are available. The data in the table below shows the average and standard deviation for each of the experience and job position questions contained in this category.

TABLE 4. Results of Mean and Standard deviations

Annual Appraisal					
Experience	Team Member	Scrum Master	Product Owner	Manager	**Total**
0-5	3,51	3,65	.	.	**7,16**
6-10	3,70	3,21	4,00	4,00	**14,91**
11-15	3,78	3,15	3,23	3,00	**13,16**
16+	3,40	3,40	4,10	3,78	**14,68**
Total	**14,39**	**13,41**	**11,33**	**10,78**	49,91

As a consequence of the data, it cannot be concluded that Hypothesis 1 is valid. There was also no significant main impact for the function of the job in the study, as shown by the non-significant values of the test statistics

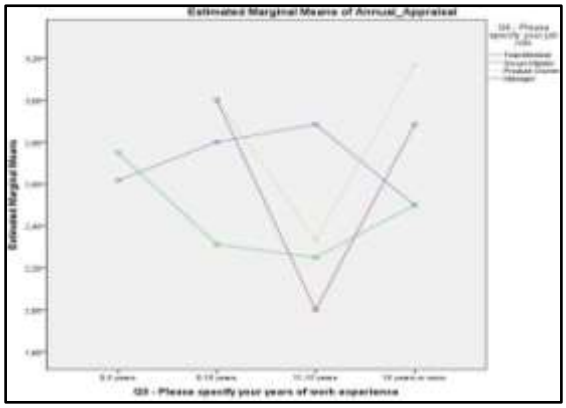


FIGURE 4. Annual Appraisal profile plots with experience

Survey findings (Questions 1-10) were examined one at a time following statistical analysis. As a reminder, the yearly appraisal system's shortcomings cannot be precisely quantified. A detailed review of survey respondents' responses is used to identify the inadequacies of the evaluation method. Some intriguing results from the effectiveness study of the individual questions assist to reveal the gaps in yearly performance evaluations

TABLE 5. Annual Appraisal advantages of employees

Choices	Totals
Strongly Disagree	19
Somewhat disagree	11
Somewhat Agree	30
Strongly Agree	40

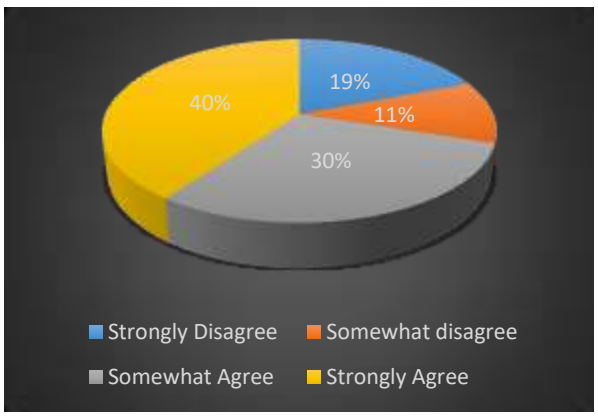


FIGURE 5. Annual Appraisal Limitations of employees.

70% of respondents believe that yearly performance evaluations are effective in an agile setting and have significant role (Figure 6). Thirty percent of respondents believe it is unable to keep up with the fast-paced atmosphere in which they work.

TABLE 6. Co-operation & Team Spirit of employees

Choices	Totals
Strongly Disagree	38
Somewhat disagree	30
Somewhat Agree	20
Strongly Agree	12

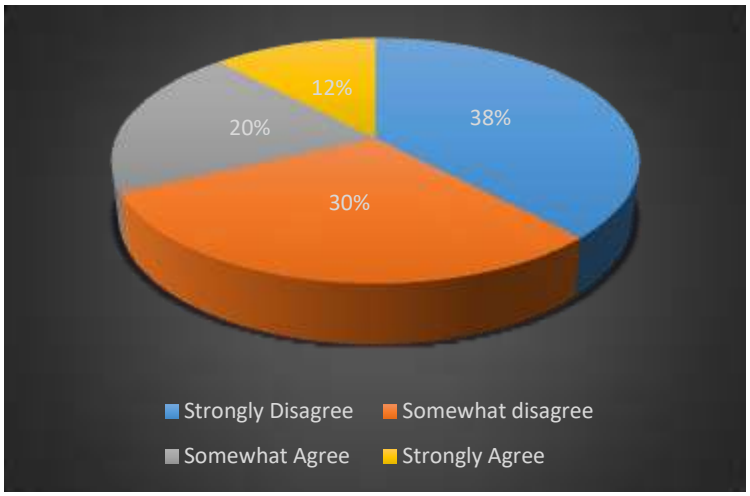


FIGURE 6. Co-operation & Team Spirit of employees

There is a lack of team spirit and collaboration across teams based on the findings of the survey (Figure 6). More over half (68 percent) of the people polled support the idea, while 32 percent oppose it. The personalized focus of yearly evaluations is one of its most distinguishing features. Theoretical stance on performance evaluation states that greater focus should be placed on individual evaluations rather than team evaluations, and this survey results is consistent with that. As a result, it's probable that the team's output may suffer. According to the results of a poll, this is likewise the case.

TABLE 7. Goals & Objectives with day-to-day tasks of respondents

Choices	Totals
Strongly Disagree	10
Somewhat disagree	30
Somewhat Agree	37
Strongly Agree	23

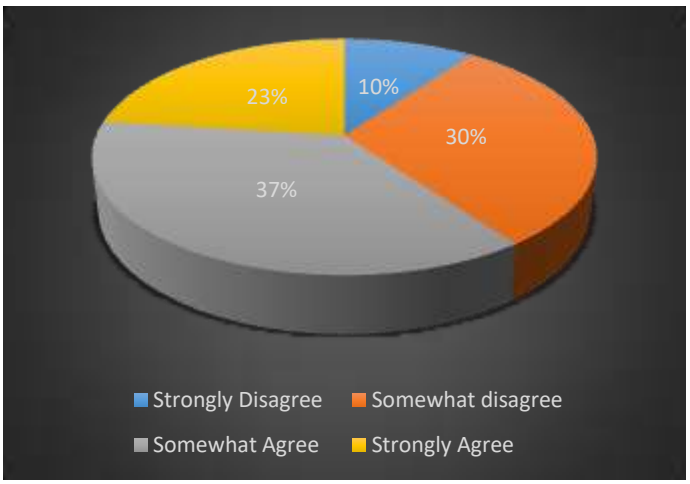


FIGURE 7. Goals & Objectives with day-to-day tasks of respondents.

Figure 7 shows another intriguing fact: there is a link between long-term goals and short-term chores. While 60 percent agreed that it exists, only 40 percent disagreed. The ability to anticipate the unexpected might make aims gainful, according to (Rodney Brim, 2004). Requirements are often revised in agile techniques. So, making goals on a yearly basis is a better utilization of time. During the transition from conventional to agile development, firms may have adjusted their processes or tactics in order to adapt to the new paradigm. Some firms are lagging behind or have restrictions in the technologies they use for performance evaluation, according to interviews. For example, you can only enter data like objectives, self-assessments, manager ratings, and so on during a certain time period each day. The feedback data, on the other hand, is kept separate from the application. As a result, the existing evaluation procedure obscures important information about continual feedback. The existing annual performance evaluation system lacks a robust feedback mechanism that can be used on an ongoing basis. The more input an employee receives, the more likely he or she is to grow as a person. It's critical that the technologies you use can keep up with the constant inflow of data. This will make it easier to keep track of all relevant data as it becomes available. Continuous improvement will be facilitated by this. A continuous feedback approach where "team members assess one another (as opposed to supervisors rating subordinates), therefore capturing volunteer contributions and mentorship" is in accordance with our findings (Kieran, et al., 2011). Getting timely feedback is also critical, as it helps the individual stay in sync with the expectations of the business. It appears that the majority of those surveyed like receiving anonymous comments from colleagues, supervisors or senior executives. There should be a constant and positive exchange of comments. Most consumers, according to a recent poll, prefer a feedback cycle that occurs once a month. The present performance appraisal system focuses mostly on the employee's long-term goals.

5. **Conclusion**

Considering the paper we can say that a better performance evaluation system is must for agile businesses. We've highlighted the suggestions that may be implemented in an agile setting to make it more appropriate. In the process of moving to an agile methodology, the study discovered that firms succeeded to connect their performance evaluation tools with agile's working approach. The purpose of this research was to interview a wide group of people who work in agile teams about their thoughts on the performance evaluation method. Due to a lack of time, the survey's sample size was limited. An effort might be made to expand sampling in the future so that as much information as possible can be gleaned.

References

1. A. Alorwu, N. van Berkel, J. Goncalves, J. Oppenlaender, M. B. Lopez, M. Seetharaman, and S. Hosio. 2020. Crowdsourcing Sensitive Data using Public Displays: Opportunities, Challenges, and Considerations. Personal and Ubiquitous Computing (2020).
2. A. Barcomb, K.-J. Stol, D. Riehle, and B. Fitzgerald. 2019. Why do episodic volunteers stay in FLOSS communities?. In International Conference on Software Engineering. IEEE, 948–959.
3. Alsari, Abdulrahman & Qureshi, M. Rizwan &Algarni, Abdullah. (2020). Agile Framework To Transform Traditional Team. 10.1109/FIE44824.2020.9274240.
4. B. Boehm and R. Turner. 2005. Management challenges to implementing agile processes in traditional development organizations. IEEE Software 22, 5 (2005), 30–39.
5. B. Boehm. 2002. Get ready for agile methods, with care. Computer 35, 1 (2002), 64–69.
6. C. Tolfo, R. S. Wazlawick, M. G. G. Ferreira, and F. A. Forcellini, "Agile methods and organizational culture: Reflections about cultural levels," Journal of Software Maintenance and Evolution, vol. 23, pp. 423-441, 2011.
7. Vikrant Sharma, M. Ramachandran, Sathiyaraj Chinnasamy, Vimala Saravanan, "A Review on Structural Equation Modeling and Its Classification", REST Journal on Emerging trends in Modelling and Manufacturing, 7(4), (2021):135-142
8. F. Chowdhury and M. N. Huda, "Comparison between adaptive software development and feature driven development," in International Conference on Computer Science and Network Technology, ICCSNT 2011, Harbin, China, 2011, pp. 363-367.
9. Shaik Mullan Karishma, Upendra Rajak, B. Kiran Naik, Abhishek Dasore, Ramakrishna Konijeti, (2022), Performance and emission characteristics assessment of compression ignition engine fuelled with the blends of novel antioxidant catechol-daok biodiesel,Energy,Volume 245,2022,123304.
10. K. Dikert, M. Paasivaara, and C. Lassenius. 2016. Challenges and success factors for large-scale agile transformations: A systematic literature review. Journal of Systems and Software 119 (2016), 87–108.
11. M.P. Jenarthanan, N G Ramkhi, M. Ramachandran, Vimala Saravanan, "Mechanical, Morphological and Water absorption properties of Polypropylene based Composites", Materials and its Characterization, 1(1), (2022):48-52
12. K. N. Rao, G. K. Naidu, and P. Chakka, "A study of the Agile software development methods, applicability and implications in industry," International Journal of Software Engineering and Its Applications, vol. 5, pp. 35-45, 2011.
13. L. Benedicenti, A. Messina, and A. Sillitti. 2017. iAgile: mission critical military software development. In International Conference on High Performance Computing & Simulation. IEEE, 545–552.
14. M. Levy and O. Hazzan, "Knowledge management in practice: The case of agile software development," in 2009 ICSE Workshop on Cooperative and Human Aspects on Software Engineering, CHASE 2009, Vancouver, BC, 2009, pp. 60-65.

15. R. Dhaneesh, Iswarya V.S, D.R. Pallavi, Ramachandran, Vimala Saravanan, "The Impact of Self-help Groups on the Women Empowerment in Tamil Nadu", Trends in Banking, Accounting and Business, 1(1), (2022):1-5
16. Mircea, Elena. (2019). Project Management using Agile Frameworks. ECONOMY INFORMATICS JOURNAL. 19. 34-43. 10.12948/ei2019.01.04.
17. N. Ganesh and S. Thangasamy, "Lessons learned in transforming from traditional to agile development," Journal of Computer Science, vol. 8, pp. 389-392, 2012.
18. S. Baltes and P. Ralph. 2020. Sampling in Software Engineering Research: A Critical Review and Guidelines. arXiv:2002.07764 (2020).
19. S. W. Ambler and M. Lines. 2012. Disciplined agile delivery: A practitioner's guide to agile software delivery in the enterprise. IBM Press.
20. N. subash, M. Ramachandran, Vimala Saravanan, Vidhya prasanth, "An Investigation on Tabu Search Algorithms Optimization", Electrical and Automation Engineering, 1(1), (2022):13-20
21. Dasore, A., Konijeti, R., Tarun, P. N. V., &Puppala, N. (2020). A novel empirical model for drying of root vegetables in thin-layers. International Journal of Scientific & Technology Research, 9(1), 2639-42.
22. T. Anning-Dorson. 2018. Customer involvement capability and service firm performance: The mediating role of innovation. Journal of Business Research 86 (2018), 269–280.
23. T. Dybå and T. Dingsøyr. 2008. Empirical studies of agile software development: A systematic review. Information and Software Technology 50, 9-10 (2008), 833–859.
24. Krishna, M.H., Dasore, A., Rajak, U., Konijeti, R., Verma, T.N. (2022). Thermo-Economic Optimization of Spiral Plate HX by Means of Gradient and Gradient-Free Algorithm. In: Verma, P., Samuel, O.D., Verma, T.N., Dwivedi, G. (eds) Advancement in Materials, Manufacturing and Energy Engineering, Vol. II. Lecture Notes in Mechanical Engineering. Springer, Singapore. https://doi.org/10.1007/978-981-16-8341-1_48
25. Trippensee, Lennard &Remane, Gerrit. (2021). Practices for Large-Scale Agile Transformations: A Systematic Literature Review.

