Have the Agile Principles endured? An empirical investigation post 20th anniversary of the Agile Manifesto (2001)

Ashish Kakar
Texas Tech University
ashish.kakar@ttu.edu

Akshay Kakar
West Virginia University
akshay.uh@gmail.com

Abstract

This study investigates whether the Agile principles introduced in the Agile Manifesto (2001) have endured today two decades later and whether they are still relevant to software developers. Further, are they positively correlated with work and affective outcomes of software development projects? We find out by conducting an online survey with team members of 58 software development project in one of the largest global IT firms. To our surprise we find that overall, the Agile principles have endured and were positively correlated with team motivation, project effectiveness and project innovation. However, they were negative correlated with project efficiency. As expected, projects using Agile and plan-driven methodologies showed differential findings.

Keywords

Agile Manifesto, Core Values, Work Outcomes, Affective Outcomes

Introduction

For many decades, software engineering was focused on heavy-weight approaches aimed at success in developing increasingly complex business applications speedily, at lesser costs and of higher quality. Formal methods based on scientific management principles using a variety of tools and techniques for measurement and standardization of the software process were adopted in the belief that it would result in success in software development activities (Kakar, 2020). However, in the late 1990s, as disenchantment with the heavy-weight engineering methods grew, suggestions for improvement came from practitioners culminating in the Agile manifesto (Fowler and Highsmith; 2001). The Agile Manifesto caught on quickly with the software development community (Kakar, 2022). By 2007 84% of the respondent organizations were using agile methods within their organizations which rose to an impressive 97% by 2018 (Hoda, Salleh and Grundy, 2018).

The Agile manifesto which represented a paradigm shift in software development from the heavyweight plan-driven methods consists of four Agile Values and ten Agile principles. The Agile principles are derived from Agile values. Over time, the Agile principles and practices to address scalability, global agile development, distributed agile development, Agile-DevOps, agile automation, automated testing, and continuous integration (see Dingsøyr and Lassenius, 2016).

Therefore, the important question to investigate is considering that Agile principles which when introduced in the Agile Manifesto and reflected the fundamental difference and a stark contrast between the two paradigms of software development, have they also diluted overtime or have they endured over the past two decades. We also assess the performance correlates of these 12 principles with work outcomes such as efficiency, effectiveness and innovation and affective outcomes such as intrinsic motivation and work exhaustion of team members. Work exhaustion is an important psychosocial outcome as it is associated with absenteeism and job turnover of high-quality employees.

In this study we also share the salient findings of our previous study (Kakar and Kakar, 2022) which showed that all four Agile values were relevant even today. However, based on participant feedback during debriefing session of the previous study and the interest and feedback received during conference presentation of the previous study we decided to investigate in this study whether agile principles are also relevant today.

URI: https://hdl.handle.net/10125/103420
978-0-9981331-6-4

Literature review

Although introduced in 2000s, the roots of Agile manifesto and principles can be traced to both Lean and Agile manufacturing paradigms introduced in the 1970s and 1990s respectively. Agile manufacturing is a further evolution of production methodology following Lean manufacturing. The term agile manufacturing can be traced back to the publication of the report 21st Century Manufacturing Enterprise Strategy (Iococca Institute 1992). The origins of the "agility movement" stems from US government concerns that domestic defense manufacturing capability would be diminished following the end of cold war in 1989. The following phenomena underscore the reasons for putting agility at the core of manufacturing strategy for the twenty-first century (Goldman et al., 1995):

1. Increasing market fragmentation
2. Growth in the need to produce to order
3. Shrinking product life cycles
4. Globalization of production
5. Distribution infrastructures which support greater customization

Leanness is usually seen as a precursor for fully agile manufacturing (Yusuf and Adeleye, 2002; Narasimhan et al., 2006). While lean production is based on four principles: (1) minimize waste; (2) perfect first-time quality; (3) flexible production lines; (4) continuous improvement (Womack and Roos, 1990), the Lehigh study included four dimensions of agile manufacturing 1.Enriching the customer; 2. Cooperating to enhance competitiveness; 3. Organizing to master change; 4. Leveraging the impact of people and information (Goldman et al., 1995; Gunasekharan and Yusuf, 2002).

While the proposed definition of leanness is the maximization of simplicity, quality and economy (Conboy and Fitgerald, 2004), agile manufacturing added flexibility and responsiveness to the definition (Gunasekharan and Yusuf, 2002). Various lean approaches, such as mixed model scheduling and level scheduling (also referred to as heijunka), have been developed for flexible production lines, but they work best under stable demand environments (Hines, Holweg and Rich, 2004). As a result, various researchers have favored agile solutions (Goldman et al., 1995, van Hoek et al., 2001).

Agile manufacturing approaches focus on addressing customer demand variability by flexible assemble-to-order systems and creating virtual supply chains (Hines, Holweg and Rich, 2004). Virtual supply chains are independent firms with distinctive core competences which come together to exploit market opportunities and disband when they are no longer valuable to each other. Further, agile manufacturing seeks to achieve competitiveness through rapid response and mass customization. While lean manufacturing methods deliver good quality product to consumers at low prices through removal of waste and excess inventory, agile manufacturing focus on rapidly entering niche markets by developing capabilities to address specific needs of individual customers.

In line with these developments in manufacturing, ASD began as a countermovement to the Taylorist software development processes like the Waterfall Model or the V-Model (Fowler and Highsmith, 2001). There is a sharp contrast between Taylorist and Agile software development approaches. Taylorist approaches are based on the principle that the first step in a product/ system solution is to comprehensively capture the full set of user requirements to address the business problem. This is followed by architectural and detailed design. Coding or construction is commenced only after confirmation of requirement specification by the customer and completion and approval of architecture/ design. The customer is typically involved at the stage of requirements gathering and the final stage of product acceptance. As a result the validation of the product happens only at requirement gathering stage and at the end of the long development cycle.

"On the other hand, agile projects work on minimum critical specification." (Nerur, Mahapatra and Mangalraj, 2005) Agile projects start with the smallest critical set of requirements to initiate the project. They work on the principle of developing working products in multiple iterations. "Users review actual working product at demonstrations instead of paper reviews or review of prototypes done in plan-driven methods." (Nerur, Mahapatra and Mangalraj, 2005) These working products become the basis for further discussions and the team uses the latest feedback from relevant stakeholders to deliver the business solution. As the solution emerges through working products, the application design, architecture, and business priorities are continuously evaluated and refactored.

Table 1 shows how Agile and Lean manufacturing principles have influenced the Agile principles

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

Table 1. The 12 Agile Principles derived from Lean/ Agile Manufacturing principles

From Table 1 we can see that all Agile principles are relevant today through topical values and strategies that have seen sustained acceptability and success.

Table 2 shows the timelines of evolution of software development approaches and the corresponding manufacturing paradigms.

Manufacturing Paradigms	Software Development Approaches
Taylorism and Mass Production (1910s)	Plan-driven approaches such as Waterfall or V Model (1970s)
Lean Manufacturing (1970s)	Lean Software Development (1990s)
Agile Manufacturing (1990s)	Agile Software Development (2000s)

Table 2. Evolution of SDMs

Theory Development

We also developed theoretical perspectives to assess the relevance of Agile principles today. We did an extensive examination of the Agile principles and why they might lead to salutary performance outcomes in software development Research efforts were focused on acquiring an in-depth understanding of what the 12 principles are (Table 3), how they originated, and why they have prevailed. From the agile manifesto and the agile principles 22 key concepts were extracted (Table 4). The 22 concepts are highlighted in bold in the Agile manifesto and in Table 1 above. Although insightful the 22 concepts did not suit the purpose of the study as a theoretical perspective from 22 concepts would be too complex.

Therefore the 22 concepts were combined through further abstraction (Table 4). This led to 6 higher order concepts or categories: organization culture, customer focus, self-organizing teams, rapid iterative development, simplicity and waste avoidance, and continuous improvement. For example, the higher order concept "customer focus" is abstracted from concepts in the Agile manifesto and principles such as "responding to change", "customer collaboration" "harness change for the customer's competitive advantage", "highest priority is to satisfy the customer".

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

Table 3. The 12 Agile Principles

	Concept	Agile Principles
1	Collaborative organization culture	(Provide) **the environment and support they need; trust them** (employees); **Customer collaboration (**over contract negotiations**); Individuals and interactions** (over processes and tools)**; face-to-face conversation**
2	Customer focus	(Highest priority is to) **Satisfy the customer; welcome changing** (customer) **requirements; harness change for the customer's competitive advantage; Working software** (over comprehensive documentation); **Responding to change** (over following a plan)
3	Self-organizing teams	**Designs emerge from self-organizing teams; Business people and developers must work together daily**
4	Rapid iterative development	**Deliver working software frequently; early and continuous delivery; working software** (is measure of progress)
5	Simplicity and waste avoidance	**Simplicity; maximizing the amount of work not done**

6	Continuous improvement	**Continuous attention to technical excellence; promote sustainable development; the team reflects on how to become more effective;** (team) **adjusts its behavior accordingly**

Table 4. Key agile concepts derived from agile principles

The shared definitions of the six higher order concepts are shown in Table 5.

1	**Collaborative Organization Culture** A culture promoted and nurtured by the management where team work, participation and consensus is valued; and openness in communication and mutual trust is encouraged.
2	**Customer Focus** A sustained organizational approach where addressing customer needs is central to all its activities
3	**Self-Organizing Team** A group of individuals working towards a common goal and having the ability and authority to take decisions to quickly adapt to changing demands
4	**Rapid Iterative Development** A responsive approach to software development where the product solution emerges through small incremental releases allowing critical product functionality to be released to the customer early and developers getting feedback sooner.
5	**Simplicity and Waste Avoidance** A philosophy where simplicity is recognized as a desirable quality attribute and all activities which do not add value to the customer are not undertaken.
6	**Continuous Improvement** An organizational strategy to pursue sustained incremental improvements of its products, processes and services

Table 5. Key Agile Concepts and their Shared Definitions

After capturing the Agile principles into the six Agile concepts, we can why Agile principles should lead to salutary performance outcomes and therefore be relevant even today. In today's uncertain time when technologies and markets are continuously evolving, a collaborative work culture that promotes internal and external collaboration is considered most suitable - a culture that emphasizes collaboration whether dealing with colleagues, customers or business partners. It is changing organizations where silo mentality had pearlier prevailed.

Customer focus is a core element of the marketing concept (Rosen, Schroeder and Purinton,1998). Theodore Levittt's (1960) seminal statement of the marketing concept argued that customer needs must be the central focus of the firm's definition of its business purpose. The needs of customers evolve continuously in response to changes in environment in which they operate. Software developers with customer focus aim *to provide competitive advantage to their customers* by acquiring the ability to address these customer demands rapidly by developing working products in quick iterations, and with minimal waste. This requires the supply function (in this case the software developer) to not recognize the traditional positions of customer and supplier. Customers too expect that its suppliers are active in their integrated search for the rooting out of all forms of waste (Womack et al., 1990; Womack and Jones, 1996).

Organizations are increasingly using cross-functional teams to enhance their competitiveness (Dumaine, 1990). Projects, especially non-routine projects, require cooperation of individuals drawn from various functional areas (Wind, 1981). Thus, to facilitate the project implementation process, it is often necessary to first foster cross-functional cooperation (Heany, 1989). All players—the sponsor, customer, user, and developer—should be on the same team. Merging their different experiences and expertise with goodwill allows the combined group to change directions quickly to produce more appropriate results and less expensive designs (Highsmith and Cockburn, 2001).

Agile projects work on minimum critical specification (Nerur and Balijepally, 2007). Agile projects start with the smallest set of requirements to initiate a project and work on the principle of developing working products in multiple rapid iterations (Kakar, 2015). These working products become the basis for further discussions and the team works towards delivering the business solution using the latest input from customers, users, and other stakeholders. Users review actual working product at demonstrations instead of paper reviews or review of prototypes as in plan-driven

methods. As a result, the project progress is visible and the ability to decide what is to be done next is more complete, thus reducing uncertainty and giving stakeholders more confidence in the state of completion of the project.

The completion effect can be used as the basis for explaining high team member morale and user satisfaction with the iterative development approach of Agile methods. Psychological research suggests that closure, or task completion, is in and of itself a potent influence on behavior (e.g., Katz and Kahn, 1966). The closer one gets to completion the stronger is the motivation to complete a task. This has been empirically supported in various studies (Lewin, 1935; Krech, 1935; Miller, 1944; Brown, 1948; Krech, Crutchfield and Liuson, 1969). If individuals are motivated to complete what they start and if this motive gets stronger as one gets closer to completion, then project completion may be a driving force behind individuals' continuing to invest efforts in projects that are already well under way. It overcomes the costs of persistence, resulting in motivated individuals and teams working towards task closure. This in turn results in greater probability of successful project outcomes and user satisfaction.

The basic principles of lean manufacturing of simplicity, waste avoidance and continuous improvement also align well with the software development (Kakar, 2014). Agile methods apply the lean approach to the overall software development life cycle. These methods focus on providing value for the customer and support requirements variability. Any activity that does not provide value to the customer is simply not undertaken. Moreover, these methods promote the cohesion of team members and developer and customer interaction (Ceschi, Sillitti, Succi and Panfilis, 2005) in line with the predominantly people-oriented rather than process-oriented approach of agile methods. The most commonly observed benefits of lean practices include improvement in quality and productivity, reduction in manufacturing costs and reductions in customer lead time, cycle time. (Schonberger, 1982; White et al., 1999).

Periodic refactoring of code leads to continual improvement in design. Deming believed that people inherently want to do a good job, and managers need to allow workers on the floor to make decisions and solve problems, build trust with suppliers, and support a culture of continuous improvement of both process and products (Deming, 2000). Instead of espousing rigid processes, Agile methods follow the lean manufacturing approach of creating a culture for continuous improvement, enabling processes to improve by learning from mistakes and successes (Poppendieck, 2001; Kakar, 2017). Thus, we expect Agile principles to be relevant even today, 20 years after their introduction and will positively affect key work outcomes positively.

Method

To assess the relevance of the Agile Values and principles today we conducted an online survey to get higher participant response rates from development team members of 58 recently completed software projects. Online surveys are easy to administer and are especially convenient to mobile participants who can respond at an available time and place of their convenience. The developers were employees of a large multinational IT consulting firm with operations across the globe. The 58 projects included software development for 42 companies across 8 countries in North America, Europe, and Asia. The type of projects included 22 which were characterized by Project Managers as Waterfall method, 4 V-method, 17 Extreme programming, 10 Scrum, 1 Crystal methodologies, 2 Dynamic Software development method (DSDM) and 2 Feature Driven Development (FDD). The study included responses from 328 developers who answered a online survey questionnaire and represented the response from 86% of developers who participated in the 58 development projects. The subjects were of average age 29.3 years and included 171 males and 157 females. The average experience of working is software development teams was 5.9 years.

The survey instrument was designed to minimize compounding and order effects. If the respondent is asked in one question about the Agile principle and in the next question about work performance or affective outcomes, his answer to the first may influence his response to the second. By grouping all questions about agile principles in one section and all performance questions in a later section, a temporal separation is created. Past research demonstrates that the temporal separation between measures reduces potential bias due to sequence and compounding effects (Sharma et al., 2009).

Measures Used

Subjects responded to all items including the Agile Value question (Do you think Agile Principle "The best architectures, requirements, and designs emerge from self-organizing teams." is still relevant for Software Development) on a scale with anchors

9=Strongly agree and 1=Strongly Disagree. Team innovation was measured using Tjosvold, Tang, and West (2004) scale. A sample item from team innovation scale is: "The team learned new ways to apply their knowledge of familiar products and technologies to develop new and unusual solutions to familiar, routine problems." Team Efficiency and Team Effectiveness were measured using the scale consisting of two sub-scales developed by Hoegl and Gemuenden (2001). The Effectiveness subscale contains items which compare actual versus intended outcomes, while the efficiency subscale contains items related to comparison of intended versus actual inputs. A sample item from team effectiveness subscale is: "All demands of the customers were satisfied." A sample item from the team efficiency subscale is: "The project was completed within schedule." Work Exhaustion was measured using the McKnight, Philips and Hardgrave (2009) A sample item from this scale is: "I felt burned out from my work". The intrinsic Motivation scale was adapted from the Self-Regulation Questionnaire (Ryan and Connell, 1989). A sample item from the scale is "I am glad to have worked on the project."

Method of Analysis

The reliability and validity of the scales were established using factor analysis. MHMR (Moderated Hierarchical Multiple Regression) analysis was used to determine the correlation between Agile Values and work outcomes. Extraneous variables such as Age, Gender and Experience were controlled for in the analysis.

Findings of the study

We first present the findings of our previous study to provide an overall perspective on the relevance of the Agile Manifesto. From Table 6 we can see that all the four Agile values:

Individuals and interactions over processes and tools (Value 1),
Working software over comprehensive documentation (Value 2),
Customer collaboration over contract negotiation (Value 3),

Responding to change over following a plan (Value 4)

were found to be relevant as the values overall were all greater than the mid-point of 5 on the 9 point Likert scale Even for projects that characterized themselves as non-Agile found Agile values to be relevant, although not to the same degree as projects using Agile Methods. From Table 7 we can see that except for efficiency the Agile Values were positively correlated with work outcomes. We can conclude from the findings of the previous study that all the Agile Values are still relevant today for software development and are overall positively correlated with work and psychological project outcomes.

		Agile Methods			Plan-Driven Methods		
Measure	**Overall Mean**	**Mean**	**Standard Deviation**	**N**	**Mean**	**Standard Deviation**	**N**
Value 1	5.53	6.31	0.564	182	4.64	0.597	161
Value2	5.49	5.89	0.781	182	5.04	0.579	161
Value 3	5.66	6.42	0.773	182	4.81	0.741	161
Value 4	6.71	7.37	0.576	182	5.96	0.582	161

Table 6: Descriptive Statistics – Values

Work Outcomes	Motivation	Work Exhaustion	Effectiveness	Efficiency	Innovation
Value 1	.834 **	0.687**	0.441**	-0.501**	0.123
Value2	.782**	0.137	0.150	-0.623**	0.219
Value 3	.821**	0.333*	0.599**	-0.423**	0.475**
Value 4	.925***	0.475**	0.656**	-0.476**	0.524**

*** p < .05 ** p < .01 ***p<.001**

Table 7. Correlation of Agile Values with Work Outcomes

The results of this round 2 of the study show that as expected the agile principles are also considered still relevant today even by developers working in non-agile projects, though to a lesser extent compared with those working on agile projects (Table 8). Except for project efficiency, the Agile principles impact all work outcomes of software development projects positively (Table 9).

		Agile Methods			Plan-Driven Methods		
Measure	**Overall Mean**	**Mean**	**Standard Deviation**	**N**	**Mean**	**Standard Deviation**	**N**
Agile Principle 1	5.599	6.332	0.611	173	4.615	0.591	165
Agile Principle 2	5.564	5.939	0.812	173	4.966	0.547	165
Agile Principle 3	5.68	6.426	0.829	173	4.781	0.779	165
Agile Principle 4	6.645	7.393	0.563	173	5.941	0.547	165
Agile Principle 5	5.606	6.281	0.529	173	4.667	0.539	165
Agile Principle 6	5.465	5.870	0.689	173	5.054	0.582	165
Agile Principle 7	5.648	6.452	0.710	173	4.764	0.689	165
Agile Principle 8	6.655	7.378	0.577	173	5.983	0.591	165
Agile Principle 9	5.559	6.400	0.605	173	4.621	0.552	165
Agile Principle 10	5.529	5.898	0.715	173	5.089	0.530	165
Agile Principle 11	5.62	6.407	0.755	173	4.826	0.818	165
Agile Principle 12	6.739	7.327	0.598	173	5.968	0.584	165

Table 8: Descriptive Statistics – Values

Work Outcomes	**Motivation**	**Work Exhaustion**	**Effectiveness**	**Efficiency**	**Innovation**
Agile Principle 1	0.840***	0.681**	0.437**	-0.515**	0.139
Agile Principle 2	0.764***	0.126	0.110	-0.590**	0.185
Agile Principle 3	0.794***	0.383*	0.640**	-0.387*	0.479**
Agile Principle 4	0.924***	0.538**	0.673**	-0.558**	0.533**
Agile Principle 5	0.781***	0.741***	0.445**	-0.456**	0.212*
Agile Principle 6	0.777***	0.058	0.153	-0.639**	0.222*
Agile Principle 7	0.831***	0.395*	0.623**	-0.428**	0.404**
Agile Principle 8	0.937***	0.448**	0.692**	-0.429**	0.517**
Agile Principle 9	0.790***	0.622**	0.497**	-0.489**	0.218*
Agile Principle 10	0.773***	0.076	0.087	-0.585**	0.272*
Agile Principle 11	0.883***	0.393*	0.650**	-0.504**	0.492**
Agile Principle 12	0.902***	0.505**	0.624**	-0.413**	0.557**

*** p < .05 ** p < .01 ***p<.001**

Table 9. Correlation of Agile Values with Work Outcomes

Conclusion and Limitations

Agile project management was introduced to mitigate the shortcomings of the plan-driven approach in managing uncertainty and change. In heavy weight methods, upfront planning, defined processes, coding standards, inspections and reviews, productivity metrics and statistical quality control was the norm. Managers not only assigned tasks to the team members but also specified how they should be performed (process) and by when (schedule) they should be completed. However, with increasing uncurtaining and change in the internal and external environment the heavy weight methods revealed various shortcomings. Uncertainty cannot be planned for but must be managed. Therefore, the focus shifted to people over processes, intensive customer collaboration iterative development through multiple prototypes, and agility in addressing change.

This study found the agile values and principles are still relevant today after the proclamation of the Agile manifesto two decades ago and impacts both work performance and psychological well-being of team members positively. Thus, project managers may find it beneficial to adopt them in their projects for superior project performance. Today, there are many agile methodologies and tools that adopt agile principles and values but to varying extents. In choosing a methodology or a tool for their projects, project managers may assess the extent to which they adopt agile values and principles. Overall, not only are team members of projects with who adopt agile values and follow agile principles demonstrate higher motivation and lesser work exhaustion, but also demonstrate higher team effectiveness and innovation. It is not surprising therefore that Agile principles are now increasingly adopted in new product development of physical products (Cooper and Sommer, (2018) and the findings of this study therefore may be applicable beyond software development.

However, the findings of the study should be viewed considering the following limitations. The analysis of the results could be performed at broad level of two categories only - Agile and Plan driven paradigms of software development. These two paradigms represent archetypes and are not seen in pure form but in their various hybrid forms. Within each category today there are multiple methods each with their own characteristic practices and approaches to software development. The sample size precluded statistical analysis at the level of each method of software development. Future studies may test the validity of the findings for specific methods of software development within these two broad categories

Another limitation of the study is the use of self-report of team members to the variables used in the survey. This raises the issue of common method bias inflating the effect size and the bias in subject responses to associated variables due to order effects. However, these biases were mitigated using tested measures and temporal separation in obtaining subject responses on independent and dependent variables (Sharma, Yetton and Crawford, 2009).

References

Cooper, R. G.,and Sommer, A. F. (2018). Agile–Stage-Gate for Manufacturers: Changing the Way New Products Are Developed Integrating Agile project management methods into a Stage-Gate system offers both opportunities and challenges. *Research-Technology Management*, *61*(2), 17-26.

Dingsøyr, T. and Lassenius, C. (2016). Emerging themes in agile software development: Introduction to the special section on continuous value delivery. *Information and Software Technology*, *77*, 56-60.

Dumaine, B. (1990). "Who Needs a Boss?" *Fortune*, pp. 52-60.

Fowler, M. and Highsmith, J. (2001). The agile manifesto.

Grant, A. M. (2007). Relational job design and the motivation to make a prosocial difference, Academy of Management Review (32), pp. 393–417.

Heany, D. W. (1989). *Cut Throat Teammates*, Dow Jones-Irwin, Homewood, Il.

Highsmith, J., and Cockburn, A. (2001). "Agile software development: the business of innovation," *IEEE Computer* (34:9), pp. 120–122.

Hoda, R., Noble, J, and Marshall S. (2011). The Impact of Inadequate Customer Involvement on Self-Organizing Agile Teams, *Journal of Information and Software Technology* (53), pp. 521-534.

Hoegl, M. and Gemuenden, H. (2001) Teamwork Quality and the Success of Innovative Projects: Theoretical Concept and Empirical Evidence. Organization Science, 12(4), 435–449.

Kakar, A. K. (2014). When form and function combine: Hedonizing business information systems for enhanced ease of use. In *2014 47th Hawaii International Conference on System Sciences* (pp. 432-441). IEEE.

Kakar, A. K. (2015) Software product features: should we focus on the attractive or the important? *Journal of Decision Systems*, *24*(4), 449-469.

Kakar, A. K. (2017). Do reflexive software development teams perform better? *Business & information systems engineering*, *59*(5), 347-359.

Kakar, A. K. (2020). A Theory of Effectiveness of Agile Software Development. AMCIS (2020) Virtual.

Kakar, A. K. (2022). A Rhetorical Analysis of the Agile manifesto on its 20th Anniversary. Preprint. DOI: DOI: 10.13140/RG.2.2.19207.47529

Kakar, A. and Kakar, A. K. (2022). Have the Agile Values Endured? An empirical investigation on the 20th anniversary of the Agile manifesto (2001). in the proceedings of Southern Association of Information Systems, Myrtle beach, S.C.

Levitt, T. (1960). "Marketing Myopia," *Harvard Business Review*, pp. 45-46.

McKnight, D. H., Phillips, B. and Hardgrave, B. C. (2009). "Which reduces IT turnover intention the most: Workplace characteristics or job characteristics?" Information & Management (46:3), pp. 167-174.

Nerur, S., and Balijepally, V. (2007). "Theoretical reflections on agile development methodologies," *Communications of the ACM* (50:3), pp. 79–83.

Rigby, D. K., Sutherland, J. and Takeuchi, H. (2016). The secret history of agile innovation. Harvard Ryan, R. M. and Connell, J. P. (1989). "Perceived locus of causality and internalization: examining reasons for acting in two domains," Journal of personality and social psychology (57:5), pp. 749.

Rosen, D.E., Schroeder, J.E., and Purinton, E.F. (1998). Marketing High Tech Products: Lessons in Customer Focus from the Marketplace," *Academy of Market Science Review* (98:6).

Sharma, R., Yetton, P. and Crawford, J. 2009. "Estimating the Effect of Common Method Variance: The Method–Method Pair Technique with an Illustration from TAM Research," *MIS Quarterly (*33:3), pp. 473-490.

Schwaber, K. (1995) "Scrum Development Process", *presented at OOPSLA'95 Workshop on Business Object Design and Implementation.*

Shwaber, K., and" Sutherland, J. (2007). What is Scrum" *URL: http://www. scrumalliance. org/system/resource/file/275/howIsScrum. pdf,[Sta nd: 03.03. 2008]*

Tjosvold, D., Tang, M. M. and West, M. (2004). Reflexivity for team innovation in China the contribution Group & Organization Management, 29(5), 540-559.