



EXPLORING THE IDEOLOGY OF AGILE FRAMEWORK IN A VERSALITE FIELD THROUGH STATE OF AGILE REPORT – A SURVEY

T. Deepa¹, S.Thaddeus²

¹Research Scholar, Thiruvalluvar University, Serkadu.

²Assistant Professor, Department of Computer Science, Don Bosco College, Yelagiri Hills,
Tirupattur

margaretdeepa23@gmail.com¹, thad@boscoits.com²

Abstract

Software development is considered to be successful if it is completed within time and budget while satisfying customers' needs. The agile methodology is viewed as one of the most outstanding fitting methodology for versatile application improvement. Recently because of the growing competition of software market more flexible methods have been in search by the researchers that can accommodate dynamic things wherever computer code necessities changing dynamically over time. This paper explains agile methodologies, agile fragility, agile maturity and the survey report of the state of agile. Moreover the key insights of the state of agile report clearly stated the current scenario of the agile framework. This research paves way for the new model of the agile based software industry. The gap between the fresh agile employees and the existing employees can be studied with a better solution through this survey.

DOI Number: 10.14704/nq.2022.20.7.NQ33001

Neuro Quantology 2022; 20(7):1-22

I. INTRODUCTION

The evolution of software development methods have been taken place since seventies. Agile techniques have been adopted to put up the active needs of business and for better understanding of software application development. The Agile approach furthermore offer administrations to firms to make the right item which makes them forceful inside the commercial centre. Accordingly the

fundamental point of this report is to full fill the necessities all through the turn of events. It is been respected that agile procedures are thundering in investigating many situations where there has been some inconsistent reports that accepted that coordinated strategy isn't adequate to for complete achievement. This study is a trial in reconsidering the literature published on agile techniques operations for the event of mobile software package



event as the researchers supposed that agile methodology solves the problem of mobile specific applications that needs high quality development processes [1]. The aim of the agile approach is help software development practices where at any point changes are acceptable and inescapable biological process changes may be identified [2]. The agile methods can also facilitate software development processes, where at any stage the changes are accepted. Many unavoidable development changes could also be identified [3]. The agile Methodologies focuses the following methods

- Extreme Programming (XP): A strategy which emphasis on programming improvement and in which society of utilization software engineers is developing to manage the issues of quick conveyance of good programming, so creating it suit the business wishes continually evolving. The majority of the XP highlights comprises little emphases with not many distribution and speedy responses [4].
- Scrum: associates degree empirical approach supported flexibility, ability and productivity during a unstable setting[5]. The specific strategies, strategies for the execution method select by the software engineers for the specific code improvement.
- Agile Modelling (AM): This methodology creates progressive models to help acute style desires and by turning them small [6].

- Crystal: It comprises of multiple ways of selecting the chief fitting one and specialty them for each and every errand [7]. Bigger comes are without a doubt to fuel further administration and troublesome ways than less difficult ones.

- Dynamic frameworks improvement philosophy (DSDM): It change the time and assets to accomplish the reasonableness instead of changing its amount in an incredibly item [8]. It is generally famous for retouch time and assets.

- Adaptive programming bundle Development (ASD): The creation and improvement of programming bundle is finished by ASD philosophy. ASD is a component of expedient programming headway and weight on rapid creation and advancement of programming bundle systems [9]. The emergency of immense and convoluted frameworks can be settled by reformist and unvaried further development.

- Feature-driven turn of events (FDD): This methodology lay weight on the arranging and developing stages, center quality perspectives all around the technique. They help in headway of task with right perceptions by containing fast and substantial conveyance of programming [10].

Agile methodologies is a gathering of programming improvement strategies dependent on iterative and gradual turn of events, where necessities and arrangements advance through cooperation between self-sorting out,



cross-practical groups and client [11]. It is a period boxed iterative methodology, and supports fast and adaptable reaction to the changes. While there are numerous approaches that fit under the standard of lithe like Scrum, XP [12], DSDM [13], Lean [14], Crystal [15] and so on, scrum is one of the most generally utilized agile structure and same has been thought about here too. Scrum is an iterative and gradual lithe programming improvement system with an attention on adaptable and comprehensive item advancement procedure where the advancement group fills in as a unit to arrive at a shared objective [16].

The Scrum system fits very well on limited scope improvement projects, where the group is little in size and co found and has a simple admittance to customer. Anyway when we attempt to apply it on after varieties of IT projects, we experience specific difficulties because of the inborn idea of these undertakings.

1. Development projects-The product projects where greater part of the work includes formative exercises like prerequisite examination, plan, coding and testing. They can be comprehensively ordered into two kinds dependent on their prerequisite size, measurements and breadth:

- Little/medium scale projects
- Large Scale projects

2. Testing projects-These are the tasks where greater part of the work spins around testing exercises like discovery testing, coordination testing, framework

testing, relapse testing and robotization of testing measure.

3. Maintenance projects -The undertakings where programming is as of now being used by end clients and the larger part work spins around support exercises relating to client service as well as minor/significant improvements, which need alteration and refinement to the current usefulness.

Software development is constantly connected with ambiguity because of evolving requirements, change in innovation or early delivery, thus plan-driven, designing based approaches are not reasonable for such unique climate [17][18]. Agile development methodologies gives adaptability to deal with evolving prerequisites, further developed correspondence and coordination components, and worked on quality while improving an opportunity to advertise speed [19]. Numerous exact investigations report the effective reception and execution of deft strategies [20]. Software companies are not following any agile method as is; rather they are using the most suitable agile practices. Hence they should have the knowledge of the benefits and issues associated with the use of each practice. Thus they ought to have the information on the advantages and issues related with the utilization of each training. Experimental examinations on lithe strategies are primarily planned towards the discoveries of embracing a particular light-footed technique with the



advantages and difficulties in reception. The effect of applying each training of the technique isn't tended to exhaustively. Likewise the writing tending to the advantages and difficulties of each training is dissipated, thus it is hard to explore each training concerning the encounters related with their use. To address this hole, a modern study is directed to catch the experience of specialists to examine the agile works on with respect to their advantages and difficulties.

II. LITERATURE SURVEY

A large number of studies have been conducted on agile methodologies with a focus on agile transformation; agile adaption, benefits and challenges of applying agile methodologies but there is a lack of empirical studies with a true focus on investigating agile practices in detail. Here is a list of empirical studies evaluating some agile practices exhaustively. These examinations are generally founded on use of coordinated techniques in huge Organisation, thus the difficulties of applying lithe practices is likewise tended to other than their advantages.

Chen et al. [21] played out a contextual analysis at Intel to explore how agile strategies can be utilized everywhere Organisations. The review featured many issues because of applying lithe practices in enormous organizations. **Begel and Nagappan [22]** played out a study at Microsoft to assess their experience of embracing agile strategies. 487

respondents contributed in the review to make it a significant report. It was discovered that Scrum and XP was being utilized at Microsoft. This review gave significant discoveries in assessing the broadly utilized, best and most dangerous agile practices while centering of coordinated reception challenges in huge organizations. The review was a decent drive in guiding the exploration endeavors to zero in on the difficulties of agile practices other than their advantages.

Fruhling et al. [23] researched the difficulties looked in agile change in an administration Organisation. Contextual analysis was directed to investigate the difficulties looked in taking on lithe practices. The review was acceptable drive in investigating the difficulties however the outcomes can't be summed up.

Salo and Abrahamsson [24] zeroed in on applying XP and Scrum rehearses in installed software improvement. The review gives summed up outcomes on the utilization of agilestrategies without nitty gritty investigation of aftereffect of applying each training separately. The outcomes can't be summed up as the review is restricted to the utilization of lithe strategies in inserted industry. **Petersen and Wohlin [25]** researched the experience of applying of XP and scrum in an enormous Organisation. They directed a contextual analysis at Ericsson to explore the advantages and difficulties of agile reception. The review was a decent



exertion towards researching the lithe practices exhaustively by contrasting the writing discoveries and contextual analysis discoveries to sum up the writing disclosures.

M. Laanti et al. [26] led a huge review including 1000 respondents from various geographic regarding the agile adoption experiences and views. The problematic areas and challenges were discussed but the challenges with a focus on adopting each agile practice was not covered in detail. **Bernhard et al.[27]** have of the assessment that ventures running in a virtual climate stay unsafe. Ventures running in eliminate mode regularly fall flat and less useful, notwithstanding an increment in innovation. In the search for answers, the authors focus on team collaboration and virtual project management and argue that new research environments are needed to achieve results with greater external validity. For this paper, genuine situations are utilized for the exact investigation of virtual undertaking group cooperation to explore mechanical and hierarchical situations for living lab setting.

Deniz Kasap et al., [28] assumed the probability of risk in R&D projects is high when contrasted with different ventures. The authors analyzed different types of risks in this paper including marketrelated risks, technology-related risks, environmental risks and organizational risks. The authors accepted that legitimate danger examination prompts proficient dynamic

in various strides of the R&D projects. The authors proposed a danger the executives system which can be utilized in R&D activities to successfully oversee risks.

Katrina D. Maxwell et al.[29] have mentioned that there is a need to benchmark software improvement usefulness. The authors have thought of a technique called "Experiment 2.0", which estimates usefulness in work focuses each hour. There are eight usefulness factors have recognized as a feature of this examination are Customer Participation, Staff Availability, Standards use, Tools use, Logical Complexity, Requirements infringement, Efficient necessities, and Staff devices abilities. However, it is not very reliable to measure projects using the average output of one "slice" or data dimension. In addition, some of the factors do not even affect productivity and a few variables in each sector are important. Organizations therefore have to statistically analyze the data to develop benchmarking equations based on key factors in productivity

Suzette Schulze et al.[30] clarified that AGCS R&D Organisation has set vital targets to work on quality and efficiency. To measure the customer satisfaction, measurements are created to follow the advancement of the recognized objectives. This also produces the Customer Satisfaction Report. In addition, the authors have introduced a yearly consumer loyalty overview to survey on a yearly premise the objectives, pointers, needs, and approaches that are



executed to determine shopper issues and needs.

III. AGILE FRAMEWORK AGILE IN EDUCATION SCENARIO

Designing and developing curricula in software engineering and technology development can be challenging, particularly because specific technologies can change rapidly. If the functional skills to which a student is introduced in the university can quickly become obsolete, how can educator's best prepare students for software development jobs? One potential methodology is to make a study hall experience in the soul of spry programming improvement standards, trying to imbue students with the capacity for creative and versatile group based critical thinking so frequently saw in agile software development team. It is this potential that the creators at first meant to investigate, and that all the while prompted their premium in the Agile Organizing Framework (AOF) created by [31].

Although agile methods have been brought into programming educational plans [32], there is no proof that they have been applied in courses or projects outside innovation improvement. By utilizing the AOF, there could be considerably more noteworthy potential for illustrations learned by the agile software development to apply wide reaching impact on the improvement of reformist teaching method in many fields. In light of contextual analyses inspected

with regards to complex versatile frameworks hypothesis,[33]recognized the developing capacities of deft groups in their definition of the AOF.

Its three characterizing standards, in light of work by [33], are to

1. ***Match the co-transformative change rate between the advancement group and the client.***
2. ***Streamline self-association***
3. ***Synchronize double-dealing (enhancements in efficiency, measures, and existing items) with investigation (development and the production of new information).***

These standards reflect key components of the social constructivist instructional method, to be specific the quest for shared significance and advanced convictions, arranged and spontaneous investigation of ground breaking thoughts, and attention to and reflection on one's own learning cycle.

The abilities, empowering agents and inhibitors distinguished by the AOF arose out of examining the practices of behaviours of agile and non-agile development teams in industry. AOF to be meant the educational environment, we depend on the social constructivist view to recognize how the cycles and practices of a agile software development team mimic the learning process in the classroom. In an agile development team, people meet up to accomplish a substantial improvement target which expects them to investigate and comprehend client prerequisites, learn



new business and specialized ideas and abilities, and reshape their methodologies dependent on new experiences. All the time, they should likewise explore complex information conditions where they should depend in different individuals from their groups for extra data and comprehension. Projects of great complexity require that the development team distribute and use their collective knowledge based on the specific interests, aptitudes and capabilities of various team members.

Likewise, in the classroom setting, understudies should investigate and comprehend the teacher's necessities for the course, including details for tasks and projects, and learn new ideas, subjects and abilities to fulfil those prerequisites. Since every understudy has various aptitudes, interests, and levels of responsibility, their comprehension of the material and the actual course will move over the span of the quarter or semester.

As ideas and activities become more muddled, the understudies will progressively depend on their teacher and on one another to acquire extra arrangement to finish the necessary work as their convictions shift in light of new implied and classified information. The jobs in the lithe programming improvement group additionally relate straightforwardly to the jobs in the study hall.

The educator capacities as the client, client delegate or undertaking champion. The singular understudy or

understudy group assumes a similar part as the engineer or improvement group. Accomplishing a "Successful learning outcome" in a homeroom is like recognizing a mechanical arrangement or conveying a task that meets the client's assumptions. Contingent on how the course is organized, the educator can completely expect or share the job of the undertaking administrator or programming advancement director with their understudies.

AGILE SUPPORTS IMPROVED CULTURE AND QUALITY FOR HEALTH WISE

Agile technology is developing health wise. New items were being planned, new workers were joining the Organisation to assemble these items, and the rundown of customers was expanding. The number and sort of electronic items produced to convey proof based clinical data, conveyed quarterly, had expanded from two to more than 10. The Agile rollout came because of the head working official's test to his group to resolve issues brought about by this extension, among them

- Increasing specialized deformities.
- Unpredictable delivery plans.
- Declining camaraderie and joint effort.

An action team was formed to find a new approach to product engineering and delivery. This report describes the transition Healthwise underwent in adopting Agile product development practices across software and non-software development teams.



AGILE DEVELOPMENT SOFTWARE METHOD

The agile philosophy provides an organization or team with the flexibility to adopt a selected subset of principles and practices based on their needs. . It is often the case, however, that the necessary agile principles and related practices are not reflected in the modified technique. Additionally, Organisations may likewise come up short on the supporting climate to viably execute the adopted practices. Thus, the advantages managed by agile practices are not completely acknowledged [34].

As a result, we are scrutinizing the 'integrity' of a tool on agile technique. Existing strategies for evaluating the 'integrity' of agile techniques are either restricted in degree or application, or spotlight basically on near examinations. Besides, on the grounds that the agile way of thinking stresses "working software" just like the essential proportion of progress [35], numerous appraisal approaches place an accentuation on estimating item attributes, and will in general overlook potential estimates reflecting cycle, task, and individuals qualities. Those couple of approaches (or systems) that do address the interaction, e.g., the Sidky Agile Measurement Index (SAMI) [36] and the Agile Adoption and Improvement Model (AAIM) [37], expect the reception of an underlying arrangement of practices.

Tragically, this has a potentially negative result of decreasing an

Organisation's adaptability to tailor its agile strategy to accommodate its objectives. At long last, outsider agility estimation tools, for example, Comparative Agility [38] and the Thought works Agile Assessment overview [39,40] spotlight on surveying the degree to which an Organisation or a group is effective in embracing and utilizing agile techniques. As individual devices, be that as it may, they give little direction regarding whether the utilized practices are fitting for an Organisation. Consequently, we imagine the requirement for a more far reaching way to deal with evaluating the 'goodness' of agile strategies, that is, surveying (a) the adequacy of a agile strategy to help its expressed targets, (b) the capacity of an Organisation to carry out its embraced technique, and (c) the viability of that technique (Effectiveness of the Method).

Our exploration is spurred by the absence of an extensive way to deal with evaluating coordinated strategies. We survey the aggregate 'goodness' of a agile technique tool on by an Organisation dependent on (1) its sufficiency, (2) the capacity of the Organisation to provide the supporting climate to carry out the strategy, and (3) the methods effectiveness. We characterize adequacy, capability and effectiveness as beneath [41,42]

Adequacy - Sufficiency of the technique as for meeting its expressed destinations.

Capacity – Ability of an Organisation to give a climate supporting the execution of



its took on technique. Such capacity is reflected in the qualities of an

Organisation's kin, interaction and task.

Effectiveness – Producing the planned or anticipated outcomes. The presence of important interaction relics and item qualities show levels of adequacy.

With the help of above three points a survey was planned to get a good and decent agile technique model for the freshers and for the senior employees in a working environment. We have planned the Objectives Principles and Practices (OPP) Framework to survey.

Scrum of Scrums

A Scrum of Scrums upholds all advancement groups. During these day by day gatherings, Scrum masters and project managers can introduce hindrances that prevent team success. The gatherings are gone to by executive team members who wish to escalate the evacuation of obstacles. Groups raising hindrances to this gathering comprehend that the executive team will settle them or discover assets to eliminate boundaries to advance.

ACTUAL AGILITY FRAMEWORK

The Action Team expected to stabilize, reenergize, and improve communication. To start the interaction, Jeff Sutherland, an founder of Scrum, introduced Scrum preparing to the greater part of the engineering group, project managers, product managers, and some executives. Groups consented to follow the Scrum execution procedures introduced and to carry out the system

without alteration for somewhere around 90 days. This key choice gave groups the devices to start working solidly and the jargon to see one another. Preparing facilitated the change to Agile and moved forward hierarchical responsibility. The Action Team immediately understood that this was the correct way of starting: Bring in an external master to sell and characterize the cycle to the organization on the loose. After the primary year of tough times and bogus beginnings, Agile and Scrum started delivering profits.

Agile methodology requires Agile Quality Assurance

A significant initial step for Quality Assurance incorporation is for both the business and the group to perceive that a quality affirmation group is useful to the venture. Advantages incorporate yet are not restricted to: finding errors before they go to production, adding an alternate point of view during improvement, and being advocates for basic bug fixes. We would say we have heard two normal justifications for why a business may guarantee the quality confirmation practice isn't required:

A) QA slows down development.

B) A good set of unit tests should suffice.

In respect to QA slowing down development, we respond by pointing out that speed and quality are often competing forces on a project. The key is to find some kind of harmony that is appropriate for the business needs.



Absolutely quality would be basic in case you are dealing with a clinical gadget that, should it fizzle, could put a day to day existence in danger. For this situation, it is sensible to expect that item advancement would take longer. Then again, in case you were chipping away at a demo for a career expo that won't ever be delivered to creation yet will be utilized to draw in financial backers, it is fitting to forego quality thusly for speed of improvement.

Have an open discussion with key business partners to examine if an interest in QA is essential to the business needs and the level of that speculation. Void of this discussion, it would be simple for crafted by the QA group to be at chances with the current business objectives, hence seen as only an impediment to the interaction. We would say, QA is best coordinated when the requirement for a QA group is perceived however the measure of assets dedicated to quality is offset with the necessities of the business. In respect to unit testing, a common debate among those who practice agile methodologies is whether a robust set of developer written unit tests replaces the need for quality assurance.

When constructing a bridge, each component must pass a rigorous set of tests before it can be used. For example, concrete must pass a strength test, each piece of steel must be of a certain grade and standard, and so on, such that every piece of the bridge has passed its respective test. Indeed, after consummation the scaffold should breeze

through one more arrangement of assessments to demonstrate it will work under tension from vehicles, trucks, wind, and so on before it is opened to people in general. Since every part to the scaffold has breezed through its assessment, this doesn't mean every one of the parts will cooperate true to form.

The contemplations are comparable when composing software and talking about the contrast between unit tests and practical tests. Unit tests check that every part of an application functions as planned. The useful tests, as planned by the quality affirmation practice, center around the client viewpoint to confirm that all parts cooperate. Coordinated software improvement strategies were initially intended for single groups of under 10 individuals.

LARGE SCALE AGILE DEVELOPMENT

Large Organisations have progressively taken on coordinated software advancement, prompting a requirement for scaling the techniques. Scaling includes difficulties like coordination between a few agile groups, absence of compositional arranging, absence of necessity investigation, just as every one of the difficulties of disseminated projects, as many large organizations are distributed [43].

Despite these challenges, a number of large companies have already chosen to adopt agile methods, even



though there does not yet exist much written knowledge either on how to scale the agile methods to large-scale projects [44], nor on the most proficient method to effectively direct a agile change in a Large Scale Organisation [45].

The significance of better understanding this subject is exemplified, e.g., by the way that professionals at the XP 2010 gathering recorded the theme Agile and enormous undertakings as the main top consuming examination question [46]. One of the practices that the agile practitioner literature, mainly written by consultants, suggests to help with knowledge sharing, organizational and process development, and coordination, is the use of Communities of Practice (COP) [47], [48].

Despite the fact that COPs have been portrayed and broadly utilized in other modern settings, see for example [49] their utilization in proficient software advancement, and specifically in scaling agile improvement to huge Organisations, has gotten little consideration in the examination writing.

AGILE CHALLENGES

Organizations face plenty of difficulties as they introduce agile techniques, practices, and tools. Because of the new technology, the methods becomes dexterous. In fact, 30% of respondents identified no fewer than ten different challenges faced while adopting Agile

The most significant agile adoption barriers include:

- Inconsistencies in processes and practices 46%
- Cultural clashes 43%
- General organizational resistance to change 42%
- Lack of skills and experience 42%
- Absence of leadership participation 41%
- Inadequate management support and sponsorship 40%

The key difficulties associations face while taking on Agile have remained largely unchanged for the past several years. Difficulties with organizational culture, protection from change, and absence of help and abilities keep on being issues.

While a wide range of scaling frameworks are in use, the Scaled Agile Framework continues to be the most popular with 37% of respondents identifying it as the framework they most closely follow. SAFe significantly outdistances the next nearest scaling method, Scrum@ Scale/Scrum of Scrums (9%). Over the last several years, there has been increasing awareness of both the opportunities and challenges offered by scaling agile practices across the organization. Initially, scaling agile was addressed through a “Scrum of Scrums” approach. Over the past five surveys, we have seen the use of SAFe grow significantly to become the dominant approach, in use by more than a third of respondents

AGILE TESTING THROUGH CONTINUOUS INTEGRATION



“Continuous Integration” describes a set of software engineering practices that speed up the delivery of software by decreasing integration times. It emerged in the Extreme Programming (XP) community, and XP advocates Martin Fowler and Kent Beck first wrote about continuous integration eight years ago. **Martin Fowler** characterizes constant mix as: "a product improvement practice where individuals from a group coordinate their work often, typically every individual incorporates to some extent day by day - prompting various combinations each day. Every integration is checked by a robotized construct (counting test) to distinguish combination blunders as fast as could really be expected." [4].

Typically a persistent coordination structure accommodates computerized source vault change recognition. At the point when changes to the vault are recognized (for example at the point when engineers check in new code) an expected chain of occasions is kicked off. A typical first step in this chain of events is to get the latest source code and compile it. If compilation does not fail, then unit tests are executed. If unit testing does not fail, the application is deployed to a test environment where automated acceptance tests can be executed. Whenever computerized acknowledgment tests don't fizzle, the form is distributed to a public area for the group. The group is then advised (for example through email or RSS) and a

report produced for the exercises that included what and the number of tests were run, the form number, connections to results and the form, and so on

STATE OF PRACTICE & STATE OF ART PRACTICES

Software development is considered to be successful if it is completed within time and budget while satisfying customers' needs. Software development is always associated with ambiguity due to changing requirements, change in technology or early delivery, hence plan-driven, engineering based methodologies are not suitable for such dynamic environment. Software improvement is viewed as effective in case it is finished inside time and financial plan while fulfilling clients' requirements. Software improvement is constantly connected with vagueness because of evolving prerequisites, change in innovation or early conveyance, henceforth plan-driven, designing based philosophies are not appropriate for such unique climate. Agile development methodologies provides flexibility to deal with evolving prerequisites, further developed correspondence and coordination instruments, and worked on quality while upgrading an opportunity to showcase speed. Numerous exact investigations report the effective reception and execution of agile strategies. Software companies are not following any agile method as is; rather they are using the most suitable agile practices.



Consequently they ought to have the information on the advantages and issues related with the utilization of each training. Empirical studies on agile methods are mostly expected towards the discoveries of taking on a particular agile technique with the advantages and difficulties in reception. Also the literature addressing the benefits and challenges of each practice is scattered, hence it is difficult to investigate each practice with respect to the experiences associated with their usage. In order to adjust the gap, a modern overview is directed to catch the experience of specialists to explore the agile works on in regards to their advantages and difficulties. The motivation behind this review is to aggregate the experience of industry specialists to more readily comprehend the advantages and difficulties of applying agile practices in projects.

The Agile techniques: XP and Scrum are observed to be the most generally utilized strategies, henceforth the acts of XP and Scrum are considered for this review. By benefits we imply that how the utilization of coordinated practices in activities can be significant. Essentially by challenges we mean the elements which make obstacles for carrying out agile practices in projects. Similarly, by challenges has been finished with the emphasis on collecting advantages and difficulties related with the utilization of agile practices. The modern review researched the strength of advantages of coordinated practices by

offering close-finished conversation starters and investigated the difficulties of agile practices by suggesting the open-finished conversation starters.

IV.STATE OF ART AGILE

Perhaps fuelled in part by an increasingly distributed global workforce, this year's report sees an explosive increase in Agile adoption across the functions of the enterprise. A consistent expansion in the number of organisations adopting agile practices and processes both inside Development and within non-IT groups including Finance, Human Resources, and Marketing. As per the state of agile report 15th Edition, the current year's discoveries show critical development in Agile reception inside software development teams, increasing from 37% in 2020 to 86%. Growth in non-IT lines of business also rose significantly, doubling in adoption since last year's report. Irregularities in cycles and practices 46%. Cultural conflicts 43%. General authoritative protection from change 46%.

Despite these challenges, it's clear that respondents continue to see the value of broadening agile adoption as a way to achieve critical business outcomes. In fact, 2/3 of respondents identified managing priorities, visibility, or alignment between business and IT as seeing the strongest positive impact from agile adoption. Conversely, organizational culture once regularly cited as a barrier to adoption is no longer a top challenge,



with only slightly more than four in 10 respondents identifying it as a barrier.

The two most urgent reasons for adopting Agile are the speed and flexibility needed by workspaces that keep on being both unpredictable and volatile. These are firmly trailed by a proceeded with continued need to focus on arrangement across groups to smooth out the software delivery process. Business and IT alignment and visibility across agile teams were highlighted as key to leading agile adoption. 2/3 of respondents identified management, visibility, and alignment as receiving the strongest positive impact from adopting agile approaches.

I would say that the number one challenge facing capital 'A' Agile today is a lack of awareness of systems thinking. And the fact that we have all these teams and organizations who have adopted agile and agile frameworks, be it Scrum or SAFe or XP, are doing so in one part of the system

KEY INSIGHTS FROM STATE OF AGILE REPORT – A SURVEY ANALYSIS

State of Agile surveys indicated a workforce becoming increasingly distributed and the frequent outsourcing of work to external vendors located across the globe. Unsurprisingly, this year's survey saw a dramatic increase in this move toward remote work and a clear indication that this direction isn't temporary.

There are few reasons for adopting Agile. State of Agile report clearly defines the various perspectives of the agile work nature. It paves way to the researchers to incorporate the current statistical scenario(Fig 1- Fig 7) in a well-defined manner. The State of agile report stated the agile adoption percentage, agile techniques employed, benefits of agile employed, agile management tools, challenges in agile work environment, agile success transformation and agile maturity. The statistical survey and the analysis helps to lead the research in various ways. The state of agile report helps to channelize the researchers work in a proper channel.



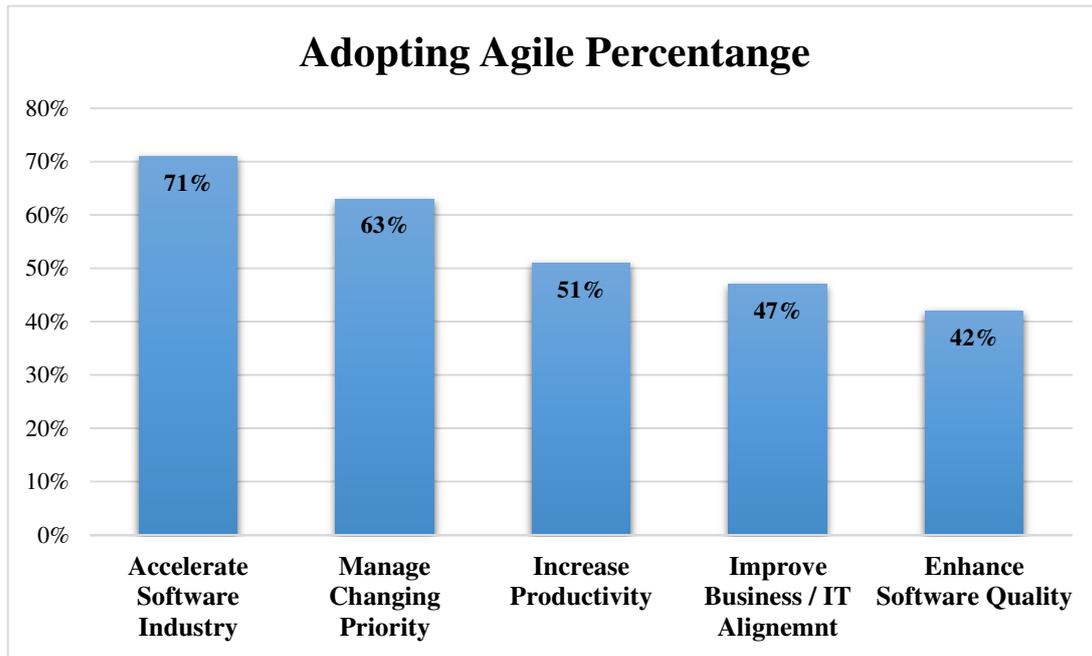


Fig 1:Agile adoption percentage

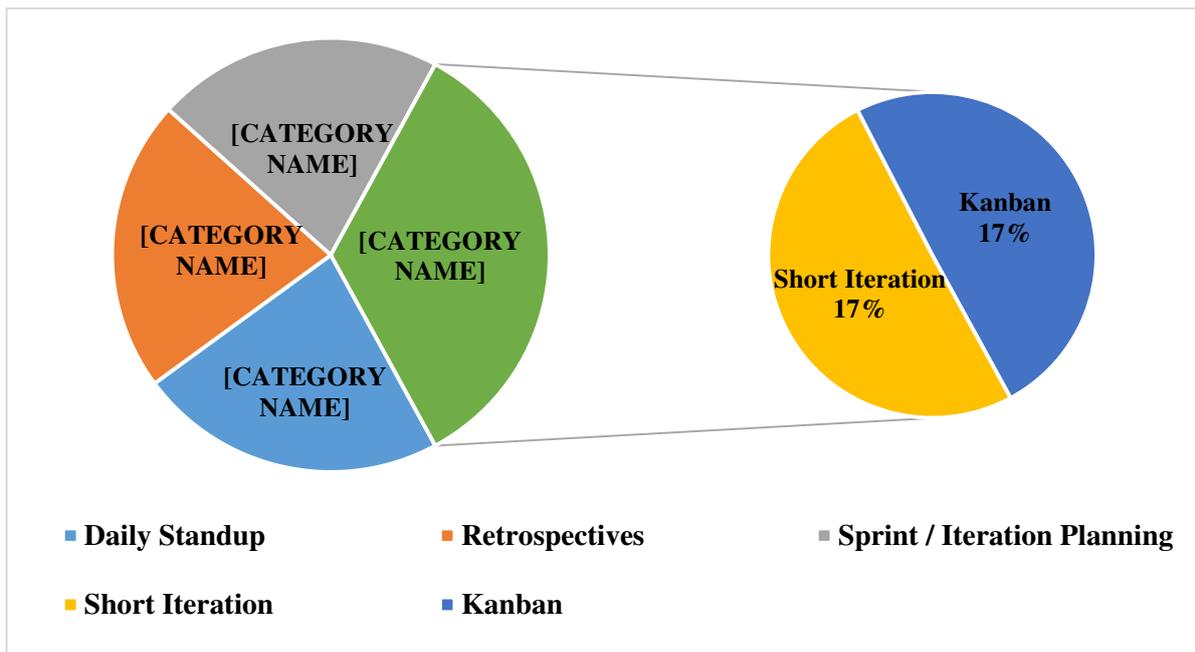


Fig 2:Agile techniques employed



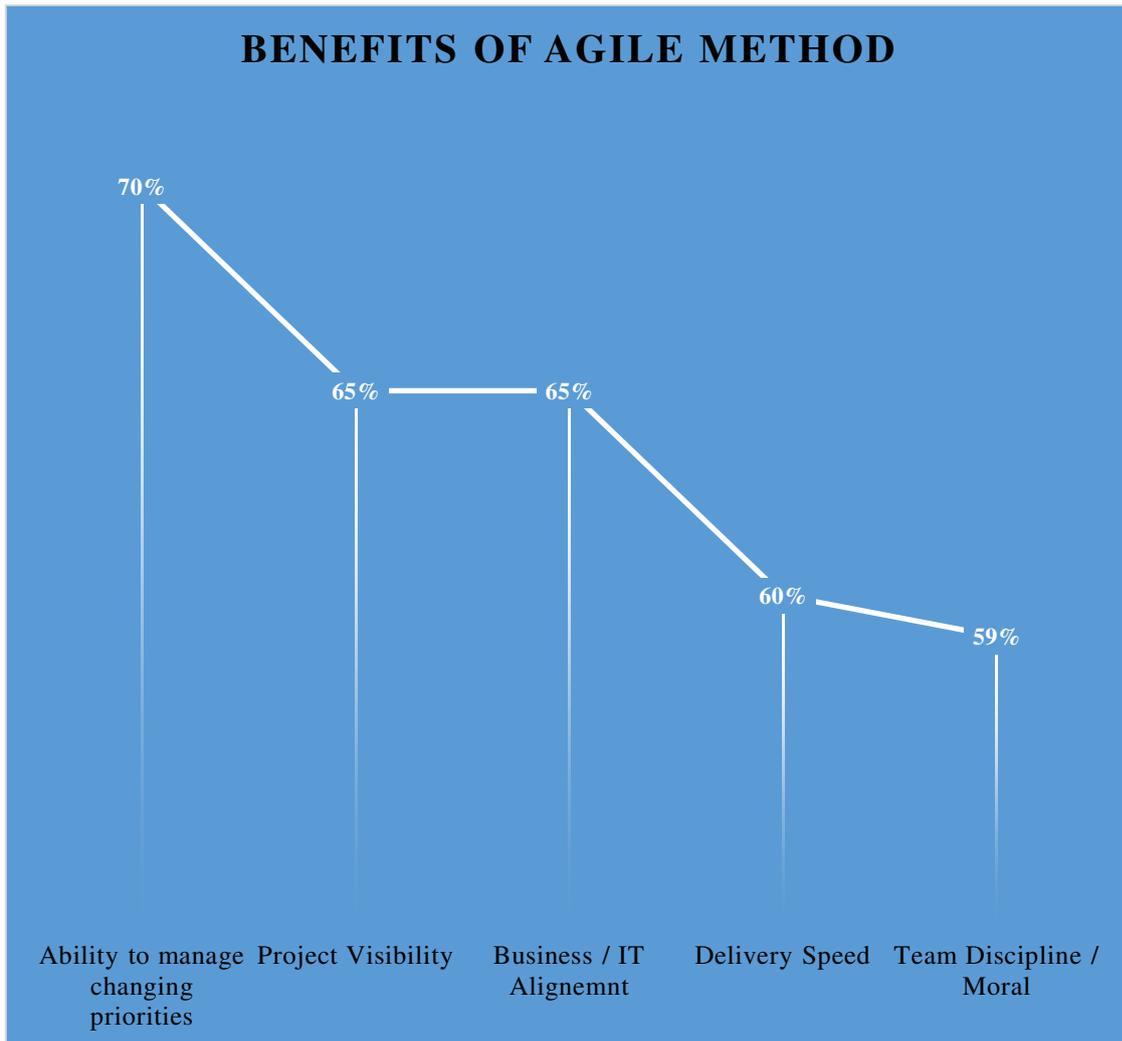


Fig 3: Benefits of Adopting Agile



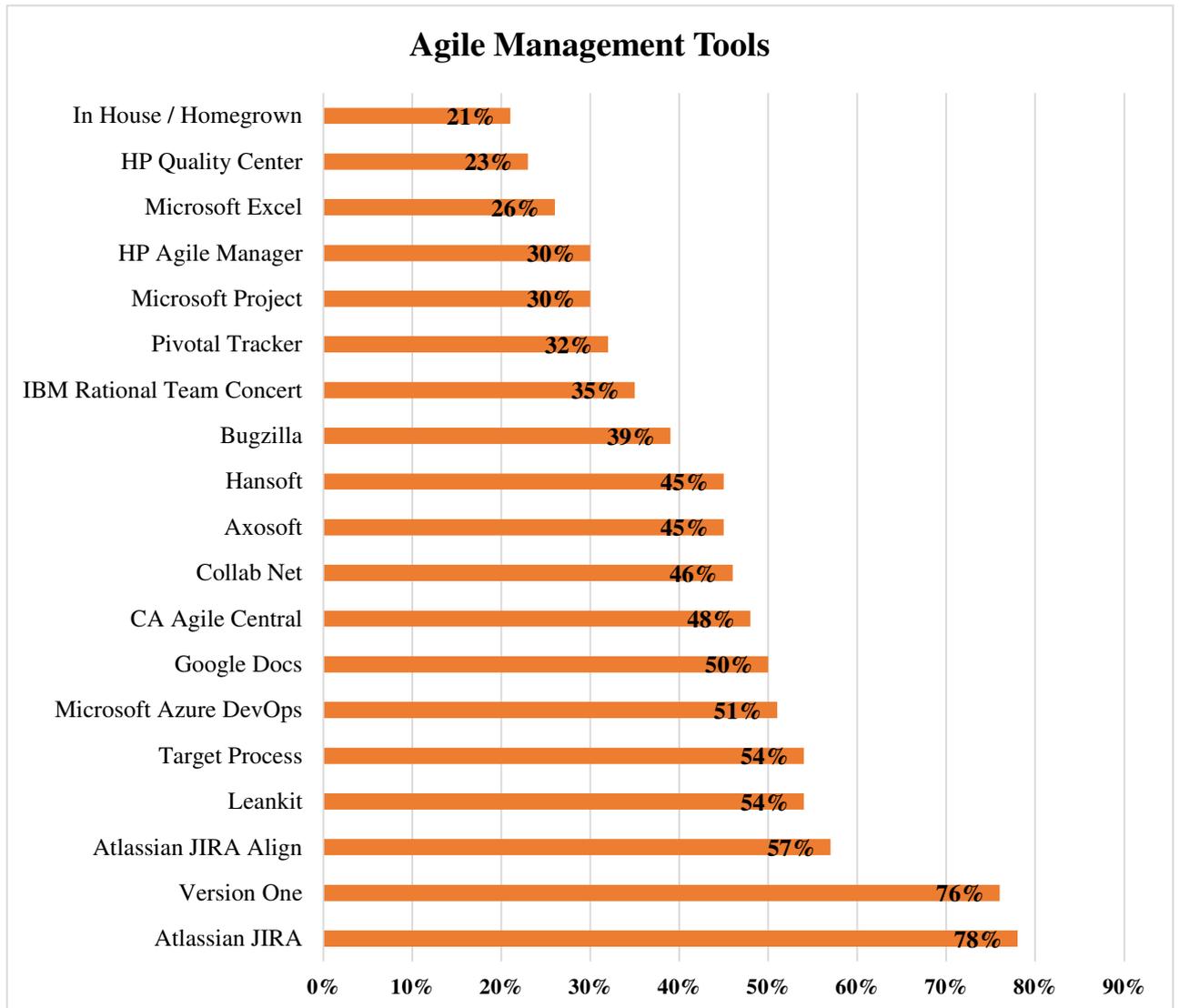


Fig 4: Agile Management Tools



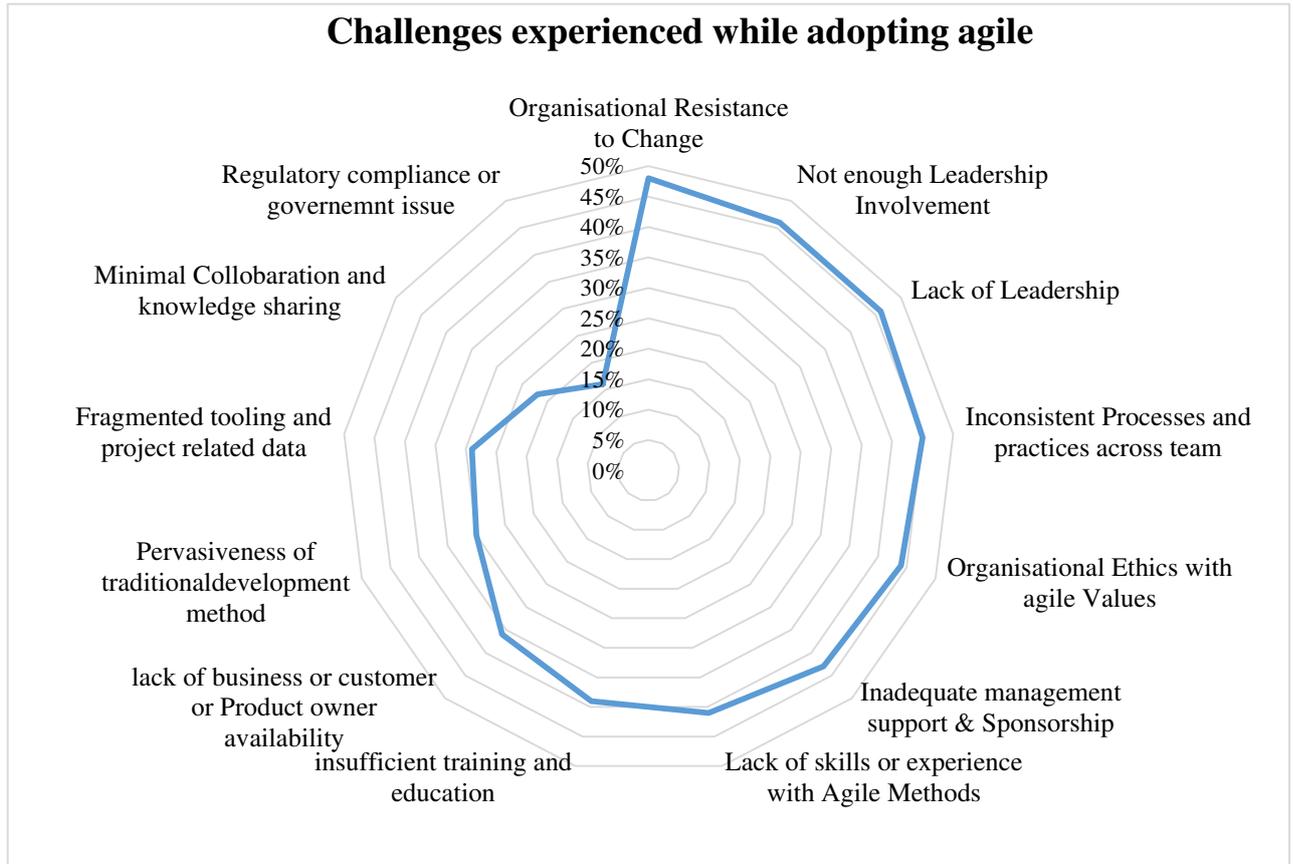


Fig 5: Challenges adopting agile

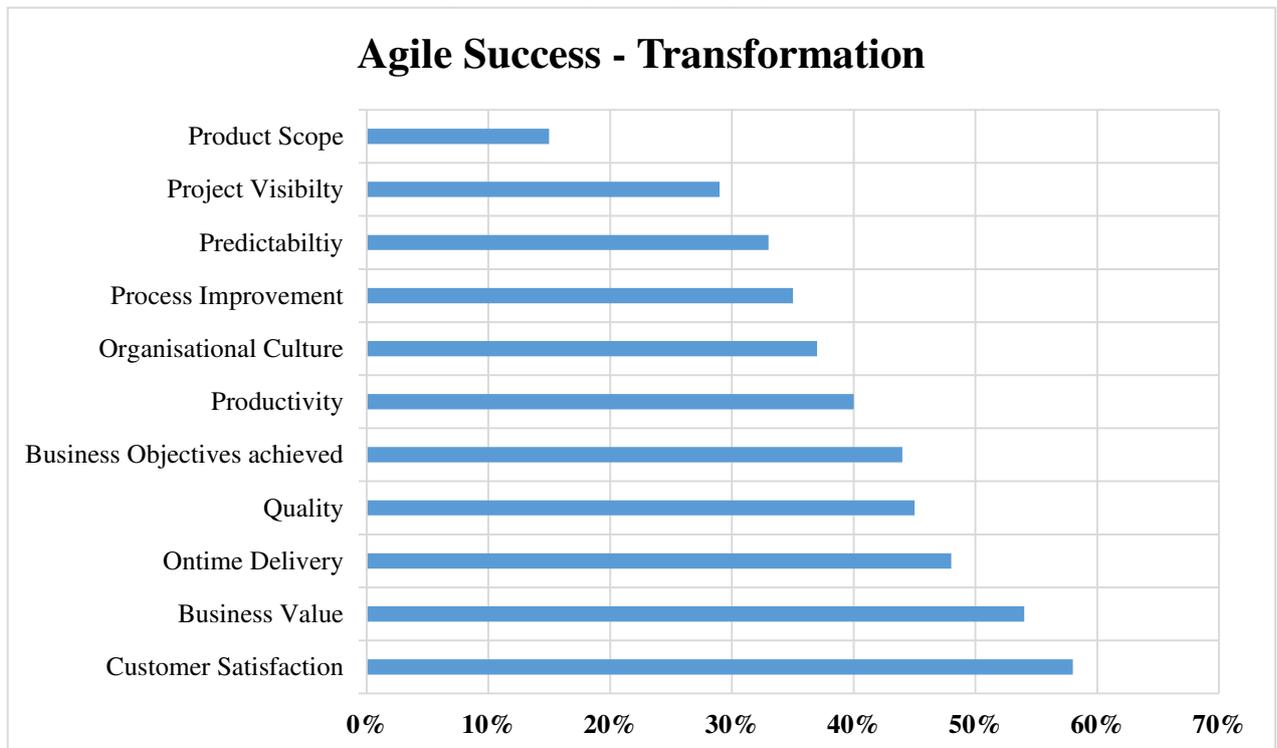


Fig 6: Agile Success transformation



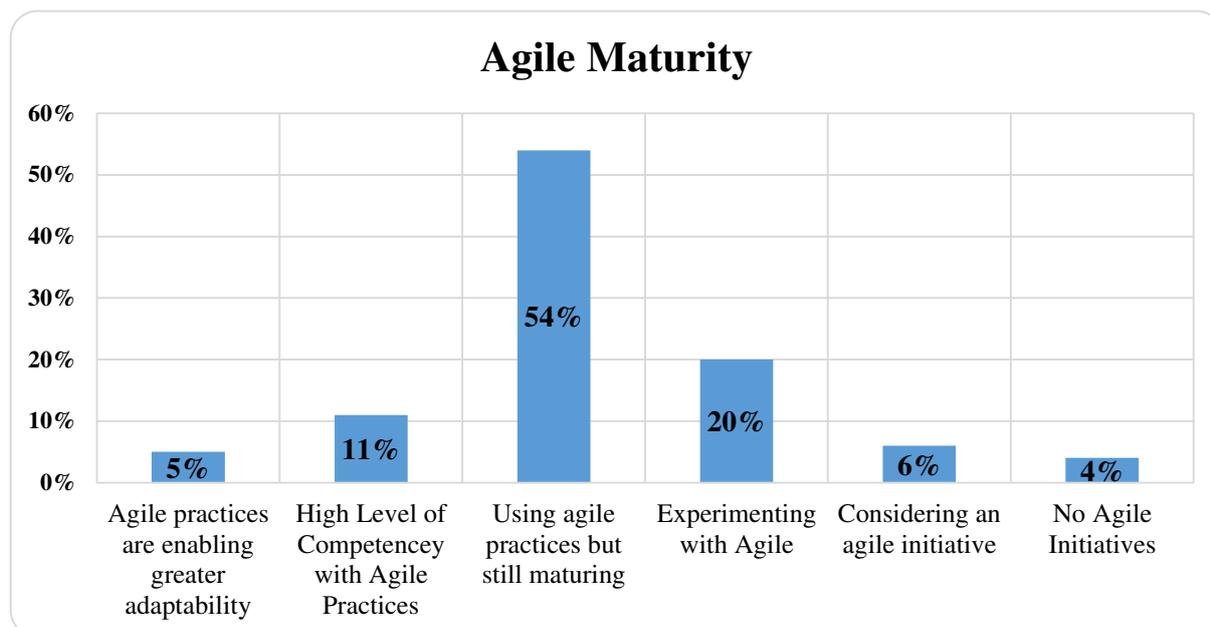


Fig 7: Agile maturity

Lessons learned from the above survey:

- Take the time to educate everyone involved about process details before implementing.
- Get a specialist or experts to mentor your Organisation
- Follow a methodology atleast for 3-4 months.
- After applying the methodology, start up with a small improvements over time.
- Get executive sponsorship and backing from as high up in your Organisation as possible.
- At the point when an error is discovered, fix it right away.
- At the point when groups team up with one another, better products emerge.
- Often scrum meeting has to be followed.
- Update the stream of work according to the Industry environment

- Agile and Scrum procedures are valuable for item advancement in numerous conditions, not simply software improvement.

Research Questions:

Sample research questions that could be explored into a context for the agile learning environment include:

1. Can an agile learning environment enhance the achievement of particular learning objectives?
2. Will an agile learning environment develop more capable entry-level software developers and engineers?
3. Does an agile learning environment improve critical thinking or problem-solving skills?
4. What is the effectiveness of each agile practice in the classroom, as described by the agile enablers?
5. Are certain agile practices more effective than others in helping students learn?



6. Are ambidextrous classrooms (that balance exploitation and exploration effectively) more productive learning environments?

7. What are specific course management practices that support the achievement of agility ?

V.CONCLUSION

There are multiple flavors of Agile like Feature Driven Development, Kanban, XP, etc. Kanban, due to its inherent nature being task oriented is better suitable for maintenance projects as compared to Scrum. On similar lines Feature Driven Development is more often used for product development where multiple versions of the same product exist. This paper reveals all the nature of agile framework and the difficulties to adopt the agile methodology. With the help of the survey the agile working new model for each companies can be easily framed.

VI.REFERENCES

- [1] P. Abrahamsson, A. Hanhineva, H. Hulkko, T. Ihme, J. Järälinoja, M. Korkala, J. Koskela, P. Kyllönen, and O. Salo, "Mobile-d: an agile approach for mobile application development," in Companion to the 19th annual ACM SIGPLAN conference on Object-oriented programming systems, languages, and applications. ACM, 2004, pp. 174–175.
- [2] R. C. Martin, Agile software development: principles, patterns, and practices. Prentice Hall, 2002.
- [3] L. Williams and A. Cockburn, "Agile software development: its about feedback and change," IEEE Computer, vol. 36, no. 6, pp. 39–43, 2003.
- [4] B. Boehm, "A survey of agile development methodologies," Laurie Williams, vol. 45, p. 119, 2007.

[5] K. Schwaber and M. Beedle, Agile software development with Scrum. Prentice Hall Upper Saddle River, 2002, vol. 1.

[6] S. Ambler, "Effective practices for extreme programming and the unified process," 1st. Ed. John Wiley & Sons, Inc, 2002.

[7] J. A. Highsmith and J. Highsmith, Agile software development ecosystems. Addison-Wesley Professional, 2002, vol. 13.

[8] J. Stapleton, DSDM, dynamic systems development method: the method in practice. Cambridge University Press, 1997.

[9] J. R. Highsmith, Adaptive software development: a collaborative approach to managing complex systems. Addison-Wesley.

[10] S. R. Palmer and M. Felsing, A practical guide to feature-driven development. Pearson Education, 2001.

[11] Manisha, Manoj Manuja, "Moving Agile Based Projects on Cloud" ,2014 IEEE International Advance Computing Conference.

[12] Lindstrom, L.; Jeffries, R. "Extreme Programming and Agile Software Development Methodologies." Information Systems Management, v. 21, Issue 3, pp 41-52. 2004.

[13] DSDM
http://en.wikipedia.org/wiki/Dynamic_systems_development_method

[14] Lean
http://en.wikipedia.org/wiki/Lean_software_development

[15] CrystalClear
[http://en.wikipedia.org/wiki/Crystal_Clear_\(software_development\)](http://en.wikipedia.org/wiki/Crystal_Clear_(software_development))

[16] Schwaber, K.; Beedle, "M. Agile Software Development with SCRUM." Prentice-Hall. 158 p. 2001.

[17] Serena, An introduction to Agile Software Development, [online] June 2007. [Cited: January 18, 2012.]
www.serena.com/docs/repository/solutions/intro-to-agile-devel.pdf

[18] J. Cho, Issues and Challenges of Agile Software Development With Scrum, Issues in Information Systems , VOL IX, No. 2, pp. 188-195, 2008.



- [19] P. Abrahamsson , J. Warsta, M.T. Siponen , J. Ronkainen, New Directions on Agile Methods: A Comparative Analysis . Proceedings of 25th International Conference on Software Engineering, pp. 244 –254, 2003.
- [20] Abrahamsson, P., Salo, O., Ronkainen, J., 2002. Agile Software Development Methods – Review and Analysis. VTT Publications 478. <http://www.inf.vtt.fi/pdf/publications/2002/P478.pdf> (accessed December 2005).
- [21] J. Q. Chen, P. Dien, B. Wang, and D. R. Vogel, "Light-Weight Development Method: a Case Study," presented at Service Systems and Service Management, 2007 International Conference on, 2007.
- [22] Begel, A. and N. Nagappan, "Usage and Perceptions of Agile Software Development in an Industrial Context: An Exploratory Study," presented at Empirical Software Engineering and Measurement, 2007. ESEM 2007. First International Symposium on, 2007.
- [23] A. Fruhling, P. McDonald, and C. Dunbar, "A Case Study: Introducing eXtreme Programming in a US Government System Development Project," presented at Hawaii International Conference on System Sciences, Proceedings of the 41st Annual, 2008.
- [24] O. Salo and P. Abrahamsson, Agile methods in European embedded software development organisations: a industrial survey on the actual use and usefulness of Extreme Programming and Scrum, IET Software, Vol. 2, No. 1, pp. 58-64, 2008.
- [25] K. Petersen and C. Wohlin, "A comparison of issues and advantages in agile and incremental development between state of the art and an industrial case," Journal of Systems and Software, vol. 82, pp. 1479-1490, 2009.
- [26] M. Laanti, O. Salo, and P. Abrahamsson, "Agile methods rapidly replacing traditional methods at Nokia: A survey of opinions on agile transformation," Information and Software Technology, vol. 53, pp. 276-290, 2011.
- [27] Bernhard Katzy and Gordon Sung (2006), "Virtual project productivity - a management issue", IEEE - ICE.
- [28] Deniz Kasap, Istemi Sidre Asyali, Kemal Eli (2007), "Risk Management in R&D Projects", PICMET, pp. 2287-2290.
- [29] Katrina D. Maxwell and Pekka Forselius (2000), "Benchmarking Software Development Productivity", IEEE Software, pp. 80-88.
- [30] Suzette Schulze, Susan Webber (1991), "Improving Customer Satisfaction Through Establishing Strategic Goals for Quality and Productivity Improvement", GLOBECOM journals, pp. 148-152.
- [31] Vidgen, Wang, 2009, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529– 551, April 1955. (references)
- [32] Rico, 2009, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.
- [33] Volberda, Lewin, 2003 I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.
- [34] A. Sidky and J. Arthur, "Value-Driven Agile Adoption: Improving An Organization's Software Development Approach," in New Trends in Software Methodologies, Tools and Techniques - Proceedings of the Seventh SoMeT 2008, Sharjah, United Arab Emirates, 2008, pp. 149- 164.
- [35] K. Beck, M. Beedle, A. van Bennekum, A. Cockburn, W. Cunningham, M. Fowler, J. Grenning, J. Highsmith, A. Hunt, R. Jeffries, J. Kern, B. Marick, R. Martin, S. Mallor, K. Shwaber, and J. Sutherland, "The Agile Manifesto," 2001.



- [36] A. Sidky, J. Arthur, and S. Bohner, "A disciplined approach to adopting agile practices: the agile adoption framework.," *Innovations in Systems and Software Engineering*, vol. 3, pp. 203-216, 2007.
- [37] A. Qumer, B. Henderson-Sellers, and T. McBride, "Agile Adoption and Improvement Model," in *European, Mediterranean & Middle Eastern Conference on Information Systems (EMCIS)*, Polytechnic University of Valencia, Spain, 2007.
- [38] L. Williams, K. Rubin, and M. Cohn, "Driving Process Improvement via Comparative Agility Assessment," in *AGILE Conference 2010*, 2010, pp. 3-10.
- [39] Thoughtworks. Agile Assessments. Available: <http://www.agileassessments.com/>
- [40] Thoughtworks, "Agile Self Evaluation Version 1.0 Prepared for Ciprian Mester, Alcatel-Lucent," 2009.
- [41] D. Ulrich and N. Smallwood, "Capitalizing on Capabilities," *Harvard Business Review*, vol. 82, pp. 119-127, 2004.
- [42] J. D. Arthur and R. E. Nance, "A Framework for Assessing the Adequacy and Effectiveness of software development methodologies," *Virginia Tech*, 1991.
- [43] D. Leffingwell, *Scaling Software Agility: Best Practices for Large Enterprises*. Addison-Wesley Professional, 2007.
- [44] E. Hossain, M. A. Babar, and H.-y. Paik, "Using scrum in global software development: A systematic literature review," in *Proceedings of the 2009 Fourth IEEE International Conference on Global Software Engineering*, ser. ICGSE '09. Washington, DC, USA: IEEE Computer Society, 2009, pp. 175–184.
- [45] M. Paasivaara, C. Lassenius, V. T. Heikkila, K. Dikert, and C. Engblom, "Integrating global sites into the lean and agile transformation at ericsson," in *Global Software Engineering (ICGSE)*, 2013 IEEE 8th International Conference on, 2013, pp. 134–143.
- [46] S. Freudenberg and H. Sharp, "The top 10 burning research questions from practitioners," *IEEE Software*, vol. 27, no. 5, pp. 8–9, 2010.
- [47] C. Larman and B. Vodde, *Practices for Scaling Lean & Agile De- velopment: Large, Multisite, and Offshore Product Development with Large-Scale Scrum*. Boston, MA, USA: Addison-Wesley Professional, 2010.
- [48] E. Wenger, *Communities of Practice: Learning, Meaning, and Identity*. Cambridge University Press, 1998.
- [49] E. Wenger, R. McDermott, and W. M. Snyder, *Cultivating Communities of Practice*. Cambridge, MA: Harvard Business Review Press, 2002.
- [50] "Continuous Integration", Martin Fowler, <http://www.martinfowler.com/articles/continuousIntegration.html>

