AGILE ADOPTABILITY AND SCALABILITY FOR LARGE SCALE PROJECTS

By

HINA SAEEDA

A Thesis submitted to the Faculty of Computer Science, Military College of Signals, National University of Sciences and Technology, Rawalpindi in partial fulfillment of the requirements of a MS Degree in Software Engineering

SUPERVISOR CERTIFICATE

It is to certify that the final copy of the thesis has been evaluated by me, found as per the specified format and error free

Date_____ _____

Brig .Dr. Fahim Arif

ABSTRACT

In different innovative shapes agile is applicable for software projects. While talking about agility it is very successful in the small and medium size projects. However have limitations when applied on large size projects like documentation, coordination and communication, time period and distributed teams. The aim of the study is to conduct a detailed SLR (Systematic Literature Review) on agile scalability and adoptability identifying limitations faced by agile techniques in large size projects ,analyzing existing models / frameworks /guidelines for agile scalability and on the basis of this detailed analysis to propose a remedial Framework for agile scalability for Large Scale projects, so that agile approach can be applied on large size projects with covering existing challenges and can achieve same results as we can achieve by applying agility on small and medium size projects. After detailed analysis of existing limitations and techniques of agile scalability a framework is suggested. The framework is validated from industry against five general and nine technical identified issues of agile scalability these issues were identified from SLR. Industry validation results that were gathered in form of expert's reviews show that framework shown its worth till now.

ACKNOWLWDGEMENTS

First and foremost I must pay gratitude to Allah Almighty for giving me access to new horizons in the literary research. All the best blessing be upon our Holy Prophet (P.B.U.H) who directed women as much as men to pursue knowledge without any discrimination.

I would like to thank Military College of Signals, NUST for providing admirable research environment at the institute. This painstaking task was not possible without the support of my Supervisor Brig.Dr.Fahim Arif and Co- Supervisor Assistant professor Nasir Mehmood Minhas who ignited critical abilities in me. I am indebted to them for their timely guidance, profound encouragement and formative criticism. There affectionate and kind consideration towards my research helped me to carry on with my project in odd circumstances. I am also very grateful to my committee members Dr.Mammona Humayun, Dr. Sarmad Sadik and Asst. Prof. Athar Mohsin for their involvement in this research work.

This work could not have been completed without untiring support and prayers of my parents, my family especially Assistant professor Abdul Haq Qazi and Col Sadat Hussain Shah. They supported me whenever I needed them.

Above all, I am thankful to Allah, the exalted, for blessing me the ability and strength to complete this work.

DEDICATION

This report is dedicated to my dear parents whose love, affection and support enabled me to achieve yet another milestone in my professional life.

LIST OF ACRONYMS

1) Dynamic Systems Development Method	**DSDM**
2) Extreme programming	**XP**
3) Systematic Literature Review	**SLR**
4) Literature Review	**LR**
5) Capability Maturity Model	**CMM**
6) Institute of Electrical and Electronics Engineers	**IEEE**
7) Research Question	**RQ**
8) Population, Intervention, Context, Outcome, and Comparison	**PICOC**
9) Disciplined Agile Delivery	**DAD**
10) Agile Modeling	**AM**
11) Feature driven Development	**FDD**
12) Business intelligence	**BI**
13) Agile Software Development	**ASD**
14) Lean Based Scrum Of Scrum	**LBSOS**

Table of Contents

Figure No **LIST OF FIGURES** **Page No**

Table No	LIST OF TABLES	Page No

Chapter 1 Introduction

1 Background

Different flavors of agility are available such as SCRUM, DSDM CRYSTAL, XP and XP2, each having some unique and specific properties .While talking about agility it is very successful in small and medium size project [1]. Agility is combination of the factors that leads the project towards success these factors make the project to have qualities like Modularity, Iterative nature, Time-Bound, Parsimony adoptions, Incremental nature, Convergent, People-Oriented and Collaboration among team members [2]. The agile qualities such as less documentation, pair programming, and high teamwork produce excellent results for the small and medium level projects large size projects are also using the approaches but with challenges [3].

In contrast, when applied agility on la large size projects it does not provide the desired results. When talking about scalability of agile two terms are in use "Scaling out" and "scaling up" Scaling up' is concerned with using agile methods for developing large software systems that cannot be developed by a small team. Scaling out' is concerned with how agile methods can be introduced across a large size projects with many years of software development experience [4]. It is not the truth that 100 percent its application is going to fail but like small level and medium level projects results, it does not show the same for large size projects. The agile approaches such as crystal blue are in use for large size projects but show less agile properties [5]. Techniques of agility like SCRUM are applied for the large size project but it has also some limitations [6]. The current research is focused on how to extend agility towards regulated environment or large size projects and according to researchers; it seems as the final frontier of agility approaches [7].

The discussion on ability of agile practices to scale to "large" software development efforts (more than 3 teams or 30 team members) has been widely debated in recent years [8].In this research work purpose is to suggest Framework for agile scalability for large size projects

1.1 Problem statement

In software engineering, agile methodologies are used to handle current challenges in software industry such as low cost, time deadlines continuously changing requirements of market and uncertain situations. Agile is beautifully answering these challenges while apply on the small and medium level projects, but it has limitations when it is applied on large size projects. The problem identified is "Scalability of the agile approaches for large size projects".

1.2 Reason/ Justification for the Selection of the Topic

The aim of the research is to investigate the existing problems and challenges in agile processes scalability for large-scale projects. The objective is to provoke understanding, proper analysis of agile limitations for large size projects and designing a Framework for the agile processes scalability for large size projects.

1.3 Material and Method

This research work aims to address scalability challenges of agile approaches for large-scale projects by identifying various challenges after conduction of detailed SLR and then designing a remedial Framework, which will help minimizing, and reducing the observed challenges faced by agile approaches when applied to large scale projects.

In this research work, after conduction of detailed SLR and analyzing of the existing approaches of agile for large size projects we will present a remedial Framework. Remedial framework purpose is improvement in the efficiency of agile processes and techniques for large-scale projects.

Once the challenges identified will move to the next step of analyzing the identified challenges followed by identification of the treatment technique. The evaluation of this Framework will conduct through feedback from expert reviews.

The Framework will provide a guideline for “how to make scalable the agile techniques for large –size projects “successfully as it is producing results for small and medium size projects.

1.4 Objectives

In this, study systematic literature review will be done of previous existing Frameworks /Models /Guidelines for Scalability of Agile approaches for Large-scale projects. From systematic literature review the deficiencies of existing frame works/Models/Guidelines will be studied and reviewed. Based on these findings Framework will be introduced after the framework introduction it will be validated for agile scalability and adoptability in large size projects.

1.5 Relevance to National Needs

Pakistan Software industry is a vast growing industry in the world .Pakistani Industry is producing software at large scale in global market. The emergence of software offshore outsourcing phenomena hits every country positively and one of the offshore client/vendor attracting factors is CMM certification. CMM is a well-recognized and globally accepted standard in software development to process control, quality improvement, and measurement of software firm's capability.

Therefore, Pakistani software firms are striving to achieve the CMM certifications to earn profit from offshore clients. In fact, CMM certification is not easy to adopt for all level of firms. Pakistan software industry is at its initial stage, so it is not recommendable to jump for CMM by software firms directly.

The present research suggests that these software firms should go for adaptation of agile development methodology for excellent performance on the footsteps of CMM. In this way, software firms can leads toward CMM after agile maturity. The use of the extreme Programming (XP) as an example that proves Agile Development Methodology can well fit the CMM Framework and software development firms can achieve their objectives to participate in offshore outsourcing phenomena [19].

From literature review it is found that researchers are putting emphasis on suggesting remedial frameworks for agile scalability.

1.6 Advantages

Making Scalable Agile Approaches for large size projects. It will cover existing limitations of the agile techniques for large size projects. It will give quality result by agile applications in software industry by producing following traits in agility such as large size teams ,distributed environments ,needed proper documentation ,high coordination among teams ,amplified learning ,reduce waste during process flow, team empowerment ,building integrity and reducing risk .

1.7 Area of Application

The Large size software projects in software industry applying agile software development methods and techniques .The projects faced problems in agile scalability having large size teams ,distributed environments ,needed proper documentation ,high coordination among teams ,amplified learning ,reduce waste during process flow, team empowerment ,building integrity and reducing risk will apply the proposed technique for overcoming the limitations faced by the agile approach while applied in large size projects .Organization striving for achievement of CMM standards can adopt Agile practices and by Agile Practices Maturation they can lead towards maturity levels .

Chapter 2 Literature Review

2.1 Introduction

Implementation of the agile practices in different organization worldwide is at increasing speed [28]. More and more organizations are moving toward adopting agile software development. This is ambitious by the steady need of producing better, faster and cost-effective software solutions and at the same time keeping a high speed of employee job satisfaction [34].

Agile software processes is distinguished by its aptitude to develop in module form, few weeks' time-frame for delivery with short verifications and corrections cycle, and incremental methodology. As such, this process requires vigilant teamwork and contact in different teams working in a project.

This method can be explained by co-located development team, as this needs smaller steps, easy solutions that look forward for regular testing, resources need to increase and project will need to be divided into sub- projects for larger size projects. Since minimum documentation is more suitable in the agile process, resources will entirely concentrate on development and testing only [35].

They are being applied by a series of organizations agile software development methods are contributing in industry very strongly, with 76 percent of organizations reporting that they had one or more agile projects. Agile is coming out as widespread used methodology because it works ,organizations are analyzing that agile project teams when analyzed against traditional project teams gain higher success rates, deliver higher quality, have greater levels of users satisfaction, provide better return on investment (ROI) and deliver systems to market earlier as compared to other methodologies [36].

The agile development approaches of information technology projects have many distinctive properties, including early customer participation, iterative release, self-organizing team, flexibility of the project etc. It can give customer acceptable product

knowledge and completely adapts to specific situations, such as high-risk, irregular and small-scale exploration software projects [52].

Agile principles give us a good combination of characteristics that make a project winning the principles along with the key words are given in the below table 2.1 [58].

Agile methods application according to the earlier surveys performed is a great deal as compared to the other software process in use ,the surveys calculation is representing that agile applications are more famous between users as compared to the other traditional software development process. In the below figure 2.2 term RO and VO are representing two teams studies conducted on two different projects [62].

Table: 2 .1 Agile principle and the corresponding keywords [58]

S. No	**Agile Principles**	**Keyword**
1	Our highest priority is to satisfy the customer	Customer Satisfaction
2	Welcome changing requirement even late in development	Embrace change
3	Deliver working software frequently	Frequent software delivery
4	All team must work together on project	Customer involvement
5	Build projects and motivated individuals	Team motivation ,training and support
6	Face to face conversion for conveying information	Face to face communication
7	Working software is the primary measure of progress	Working software
8	Agile process promote sustainable development	Sustainable pace
9	Continuous attention to technical excellence	Technical excellence
10	Simplicity -the art of maximizing the amount of work not done is essential	Simplicity
11	The best architecture, requirements, and design emerge from self-organizing teams	Self-organizing teams
12	At regular intervals the team reflects on how to become more effective than adjust behavior accordingly	Inspect and adopt

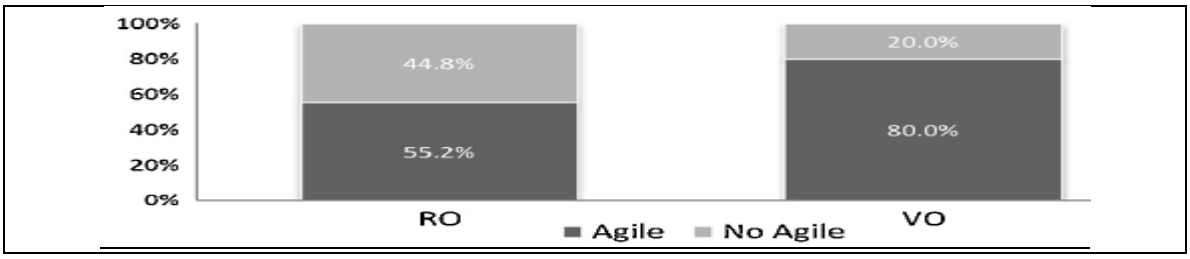

Figure 2.1: Current rate of agile method usage as compared to no agile (or alternative) methods [62].

2.2 Agile software development success in small and medium size projects

According to literature review agile methods used for the current challenges of software development industry are giving good results on small and medium level projects ,but showing limitations when applied on large scale projects the methods offered by agile software engineering are XP,XP2,SCRUM,DSDM ,Test Driven Pair programming etc. each showing some strong and some weak areas when applied[1]. Agility is showing best output for Small and medium size projects, but has limitations for large size projects [3].Small companies are enjoying the agility in true senses [24].

It has been suggested that early adopters of agile methods have been small high-tech companies working for in-house teams developing software for quick to market applications [28].

Agile techniques have beauty to handle volatile Requirements [23].Agile has success factors that are giving benefits when applied in the small or medium size projects according to the current cited research paper there are different benefits that agile can provide to the small size projects table 2.2is giving more detail explanation [48] .

As mentioned earlier, many agile success stories come from projects with similar properties, such projects are small (5-50 members) and have teams working on a single place collectively working on same projects that need a comparatively less set of skills (web, database). Project members are part of the same management organization and are easily redirected based on project requirements.

These projects are comparatively free of governance and reporting such as long-range cost estimates, charge numbers, earned value, and the like. In addition, developers

normally control the change management and delivery method instead of a separate organization that limits the capability of the team to accept these practices [52].

XP is dynamic and effective provided that it is modified for the individual projects. Then, one can obtain highest benefits from XP [51]. The basic benefits when using agile approach in combination with cloud computing comprise are: 1) Increased quality of application 2) efficient use of resources 3) less time-to-market 4) budget management [57]. Agile when applied to small team, co-located team, generate improved results [58]. Agile approaches are frequently welcomed by both executive level members and programmers as providing a much wanted delivery from the all extra efforts in general seeming as being forced by traditional software development methods.

2.2: Table Benefits provided by the agile methodologies [48]

1	Improved communication and coordination
2	Quick Releases
3	Flexibility of Design –Quicker Response to changes
4	More Reasonable process
5	Increased Quality
6	Better customer focus
7	Improved focus –Better Prioritization
8	Increased productivity
9	Better Morale
10	Testing First

Created in the context of small, projects, agile methods are mostly seen as not able to size to larger projects. Their implementation seems to need an all or nothing approach, suggesting that ‘‘being agile” is binary [60].

2.3 Limitations of Agile scalability and adoptability in large size projects

In literature there is complete proof that agility is applied beautifully in small size and medium size projects and on the same time all authors are raising the issue that

agile benefits are just limited to the small scale projects and have limitations on large size projects due to the nature and properties of the large size projects agility faces certain limitations in large scale projects.

But somewhere in research we have also proofs that agile techniques are not 100 percent failed when applied on large projects, in fact some agile techniques are generating good results such as SCRUM ,and CRYSTAL but facing some limitations also and lose their agility as they are applied in hybrid form [5][6].

According to the researchers if agility gets success in scalability towards the large size projects it will be its success last frontier [7].The argument on ability of agile methods to scale to "large" software development projects (having more than 3 teams or 30 team members) has been broadly debated in latest years [8].

Researchers reviewed agile processes to determine where and how to insert inventiveness techniques into them. They adopted Robert Sternberg's definition of creativity as the ability to produce work that is both novel and appropriate. Therefore, they wanted to inject techniques that would generate novel and useful requirements. By useful, they are talking about requirements that, if satisfied by the software solution, would contribute positively to achieving one or more Project-specified goals [9].

In spite of agile software development being thought as Mainstream model from about 2010, the idea, and hence its strengths and restrictions are still comparatively blurred. One approach to improve clarity is through analyzing, real practice across a broad range of projects. Two such studies performed using detailed surveys of software development practice in companies in Northern Ireland in 2010 and 2012.

The main aim of the first survey was to recognize the level of agile espousal in the area, while the second focused on understanding how agile principles and practices were growing and strengthening [10].

The implementation of agile development techniques has increased considerably in different software areas. This case study explains some of the benefits of the agile methods to the scientific-software development. Agile software practices are well suited to the investigative, iterative, and collaborative nature of scientific inquiry, especially when considering theory development through scientific experiments [11].

In current research researcher purpose was to contribute with empirical proof on the impact of using agile principles and practices in large-scale, industrial software development. Research was paying attention on impacts within seven areas: Internal software documentation, Knowledge sharing, Project visibility, Pressure, stress, Coordination effectiveness and Productivity [12].

The agile software development approach does not stop ensuring the security of software increments formed at the end of iterations. It proposes a technique for security guarantee of software increments and integrates security-engineering behavior into the agile software development [13].

Up to date software products solve more and more complex problems leading to growing complication of the software products themselves. An uncontrolled growth of complication over a long time introduces a mixture of technical risks that have potential of jeopardizing the business of companies.

There is an increasing need for frequently managing these risks, since in Lean/Agile software development the self-organized teams deliver small increments of software almost endlessly.

The study developed a method and supporting measurement system for identifying the risky source code files and assessing the scale of the risk [14].The extensive use of lean and agile development Methods show a fundamental move in how projects try to manage with complication and instability issue.

In large-scale systems, the coordination of different people generally results in a team of teams' system. The multi team systems point of view describes diverse conceptual policy types for interterm coordination.

These types are explained with examples from a large enterprise software development organization [15]. In this paper researcher answered the need to engage customers in the software development phase. Specifically saying this directs attention to large-scale software development where companies work hard to meet the specific requirements and requests of a large customer base.

In this research the researcher is contributing in two-fold. First, suggest customer-specific teams (CST) as a way to build and maintain agile practices in the evolution phase of large-scale software development. Second, study insure about customer-specific teams

successful for improving (1) customer responsiveness, (2) customer satisfaction and (3) feature quality during software evolution [16].

A lightweight Release Framework can be very useful in scaling at the large set up level [17]. Regardless of the fact that lean and agile software development is popularly used recently, especially for larger-scale projects building complex products, the methodology leaves many architectural questions unsolved. For instance, agile techniques such as Extreme Programming suggests late architectural decisions and frequent remaking ,while others suggest an "architectural runway" as infrastructure for a certain set of upcoming customer features. Software "product lines" consist of a set of software products that have a common set of characteristics.

These product lines are developed from reusable main features incorporating changes in order to make customer-specific product variants. Therefore this research explorer's interoperability and complementarities of lean and agile methods in mixture with a software product line engineering methodology [18].

Issues like proper documentation and training in large size projects is also a major problem while applying the agile techniques .Factors like customer satisfaction, customer collaboration, customer commitment, decision time, corporate culture, personal characteristics, societal culture, training and learning has significant relationship with success of agile techniques in large size projects [21].

The current research issue is team and project size effect on agile processes success. Researchers are focused on finding the relation between size of the project/company and agile techniques [22].

Agile application in large size project not produced complete agility results, even agile technique name Scrum cannot be apply without making changes in it [27].

Scaling activities like scaling up agile is the problem according to the researchers the agile techniques such as Scrum and others create problems when used for large size projects [31]. Agile practices such as code refracting, less documentation etc, are not feasible in large scale systems and here agile approaches have boundaries [32]. Agile software development is about feedback and change in the large size projects if applied agile techniques regular feedback and change will be needed [33].

Case studies are important to build proofs and making conclusions on software engineering tools and approaches. Here the researcher presents a case study on an agile adoption for an organization in the United Arab Emirates (U.A.E). The institute follows the traditional Waterfall model and wants to use agile software engineering approach.

Many challenges were faced by the agile teams which caused the use to fail. The main reasons for this ineffective adoption are lack of sufficient support and restricted financial and human resources [34].

Current research is identifying problems regarding ability of the agile methods to balance the execution cycle and the proposed Framework to be used successfully across large scale projects [35].

Researchers found that using agile and incremental techniques in large-scale software development leads to benefits in one part of the process, while creating issues in another part of the process. For example, using small and interacted team's increases control over the project, but leads to new problems at the management level where coordination of the projects has to take place [41].

They also stated the main challenges while using agile and incremental practices in large- scale software development as (a) challenges in regard to realize continuous testing,(b) increased maintenance effort with an increase in the number of releases,(c) management overhead due to the need for coordination between teams,(d) detailed dependencies are not discovered on a detailed level due to lack of focus on design,(e) long requirements engineering duration, due to complex decision processes in requirements engineering,(f) requirements priority lists are hard to create and maintain,(g) waiting times in the process, specifically in design waiting for requirements,(h) reduction of test coverage due to shortage of projects and lack of independent testing,(i) increased configuration management effort. According to some researchers existing agile principles do not support distributed development environment architecture. This current researcher work is focused on the problems that arise when client is offshore and proposes an agile model for distributed development environment. These are some highlighted limitations of distributed projects a) Poor documentation b) Communication gap between Client and Developers while applied agile techniques [46].

The Scrum' an alternate of agile techniques is by far the most popular at Microsoft. Researchers results also point out that developers are most concerned about scaling Agile to larger projects (greater than twenty members), attending too many meetings and the coordinating Agile and non-Agile teams. Many developers seemed problems with agile development techniques. [48].

The first challenge for large scale systems is the scope and complication. Secondly large size projects generally involve subcontracts with other businesses (sometimes competitors) for building or integrating substantial parts of the system.

Third point is about Reporting governance, and fulfillment place extra load on large projects. Fourth point is that in many large organizations, a separate group "Quality Assurance" certifies software is suitable for release, while in agility this concept is missing [52].

Some authors have argued that ASD methods have limited applicability to large-scale projects because agile stories are not enough to get the complexities of up-front design.

This research reports a 2.5-year field study of how an ASD team for a complicated software system modified the user story concept and the Scrum technique. The team wanted to create a convention for showing agile stories which could get the complexities of the system requirements without burdening the team with unwanted documentation [56]. Agile methodologies when used on the Distributed systems overcome some restrictions but also create some more challenges and risks to the result [58].

The rank of difficulty of an organization's IT landscape has effect on agile development teams, and creates restrictions in the agile methodology adaptation figure 2.3 is explaining the increased IT landscape complexity in detail [63].

Scaling is a problem when applying Scrum in large structures .The large structures with mixtures of Scrum teams is far more complicated than a single Scrum team and faces many scaling problems. Some researchers studied these agile scaling problems when adopting Scrum in the large size projects.

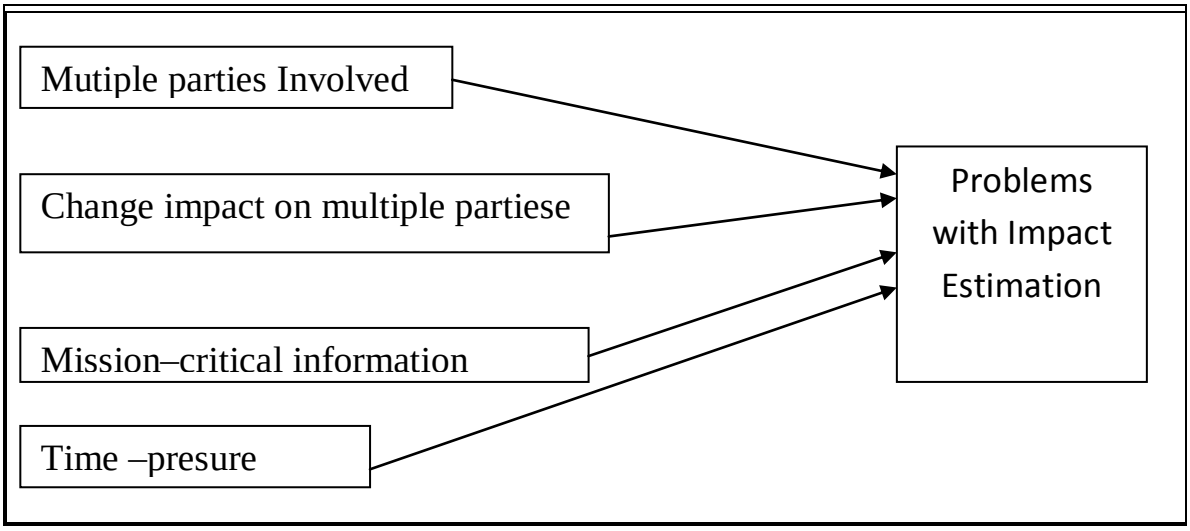

Figure 2.2: Increased IT landscape complexity [63]

These scaled agile studies found a varied set of problems, except for priority issues which are mentioned by different studies. Researchers studied a case of Agile projects with Scrum properties, and identified issues with (a) creating and maintaining the priority list, (b) effort to setup and maintain a test basis that covers a sufficient part of the applications, (c) increased management overhead due to a high number of teams that requires coordination and communication and (d) increased configuration management effort due to an greater number of releases.

The research does not indicate whether the issues are identified in a Scrum chain setting. The issue with prioritization has also been identified by researchers, who analyzed Scrum problems in medium size software enterprise, while the setting of the Scrum teams is not cleared.

In their study they summarized that prioritization of the important goals was unclear which affect organizing and tracking development work. They also identified traceability issues from high level goals to detailed plans and a lack of information about progress which made it not viable to take corrective actions in time.

Research was conducted about two companies with Scrum teams. In this research researcher does not describe whether the teams are part of a chain, they also identified issues with prioritization, as a consequence of the lack of business involvement. Other

identified issues are (a) requirements gathering problems(b) limited business feedback,(c) communication problems and (d) dependencies between Definitions of work is Done.

In other related work report experience with scaling activities and effects researchers described a success case about scaled product ownership in multiple mutually dependent Scrum teams. They state that visibility and alignment of vision, goals, teams and workload were essential elements to the success of the project. Other research describes a company that implements an agile portfolio management structure to solve prioritization issues.

Listing the projects in priority order on a single backlog creates visibility about ongoing projects, which benefits the coordination between Scrum teams. Even though agile principles aim to introduce flexibility the need for plans and structure remains. Researchers conducted a case study on a large project and conclude that combining agility with traditional plan-driven methods is essential for having both flexibility and control. Researchers argue that agile practices need to be structured to develop large software systems.

Their proposed soft structured Framework consists of a requirements gathering approach and a tailored development process. A comprehensive structure for a scaled agile application in the enterprise is the Scaled Agile Framework (SAFE) of Leffingwell. The Framework targets seven areas to achieve parallel Scrum development: (a) cross-functional teams, (b) standardized planning and tracking, (c) standardized iterations, (d) smaller, frequent releases, (e) concurrent testing, (f) continuous integration and (g) regular reflection and adaptation. Figure 2.4 is explaining the issues in detail [64].

Certain issues were identified while applied Scrum of chain concept for complex projects the issues were lack of coordination among the teams, lack of visibility of tasks , mismatch in align activity of teams ,a lack of IT chain process automation chain and unpredictability of delivery to commitment.

2.4 Scaling Concept in Agility

About scalability of the agile process in literature, the two terms are used scaling out and scaling up. Scaling up is about agile methods for developing large software systems that cannot be developed by small teams. Scaling out is about how agile

methods can be used across a large scale projects with several years of software development experience [4].

Agility for Scale means scaling of agile methods for large size projects, this grouping focuses on disciplined agile delivery where one or more scaling factors are appropriate.

The eight scaling factors are team size, geographical distribution, regulatory compliance, organizational complexity, technical complexity, organizational distribution, domain complexity, and enterprise discipline.

All of these scaling factors are ranges, and not all of them will likely to be applicable to any given project, so you need to be flexible when scaling agile techniques to meet the needs of your specific condition.

To deal with these scaling factors you will need to follow your disciplined agile delivery practices and in some cases accept a handful of new practices to address the additional risks that you face while scaling [36]. Factors of scaling in a chart form are given below in figure 2.5.

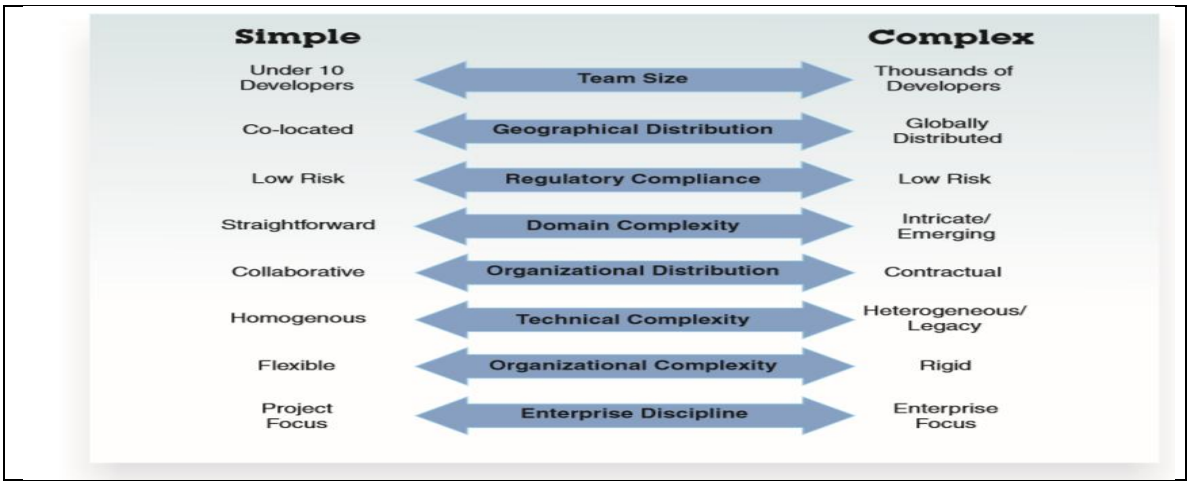

Figure 2.3: Potential scaling factors for software development [36]

2.5 Reasons motivating to adopt agile methodologies in large size projects

Agility provides ability to accept swiftly to change, market pressure demand short time period and releases.

It has ability to get immediate feedback from customer, organizational process demand high quality problem free software [22].

Agile is applied in small team successfully but can produce same results in large size projects as in large size projects the probability of change will also be larger and agility handles it beautifully [31].

Agile methodologies such as evolutionary requirements, pair programming, and direct stakeholder participation have been verified to be successfully for larger scale systems development [32].

Researchers analyzed that most experts agree that agile techniques and traditional techniques are philosophically well-matched [38].

Researchers mapped XP practices to SW-CMM model that is typically considered suitable for large-scale projects.

Additional hard work has been made to tailor the XP technique for large complex projects to fulfill speedy growth cycle times [39, 38, 40]. By an industrial case study, researchers analyzed that the basic advantages of implementing agile and incremental approaches in large-scale software development can be listed as follows: (a) requirements are more precise due to less scope and thus easier to estimate,(b) direct coordination in teams minimizes the need for documentation,(c) early feedback due to frequent deliveries,(d) rework reduction,(e) testing resources are used more efficiently,(f) higher transparency of who is responsible for what creates incentives to deliver higher quality,(g) low requirements volatility in projects and (h) reduction of waste (discarded requirements)in the requirements engineering process.

Agile practices such as evolutionary requirements, pair-programming and direct stakeholder participation have been confirmed to be successful for the development of large size systems [41, 42, and 43].

A small number of studies reported that some techniques from agile approaches such as iterative development, repeated testing and feedback, small release and refactoring are suitable for large size projects, while others like stand meeting Scholar and the use of metaphors are not possible to apply [38,40]. The Scrum variant of agile techniques is by far the most popular at Microsoft. Researchers' finding Scholar in

research also indicates that developers are most worried about scaling Agile to larger projects (greater than twenty members) [48].

In research it is stated that re- architecture project certainly encountered its limitations and has a precious lesson in learning that solid project management was just as important as a wise architectural design.

More importantly, it proved to a doubtful audience that the old ways of "rebuild from the ground up" have become out-of-date and insufficient in the modern Internet business climate. Refactoring large scale architecture is not only possible using agile methods, in research it proved instrumental to its success! Don't let the architects sell you on a full redesign; Agile Architecture is possible you have to believe [49].

2.6 Work Already done for agile scalability

By researchers a concrete Framework for analyzing of the acceptance criteria for agile SDM to apply in different projects to apply has been recommended framework presentation is given in figure 2.6 [23].

Agile scalability according to researchers can achieve by breaking the large projects into small projects [24].

In embedded systems it is recommended to keep in mind the entire thing relating to the adaptation of the agile scalability issues. According to researchers when applied the agile scalability to large size projects the main problem they faced is lack of domain knowledge about the projects [25].

In large setups the agile technique named totally integrated Scrum methodology is applied where the teams are cross functional and the teams are distributed [26]. According to the recent techniques studied for the agile approaches scalability if applied the lean in combination with the Scrum can help in the scalability of the agile approaches [27].

In research it is clear that agile methods lack detail documentation but when talking about the Scrum and XP it is proven that lack of documentation is not present but the less of documentation concept is present [28].

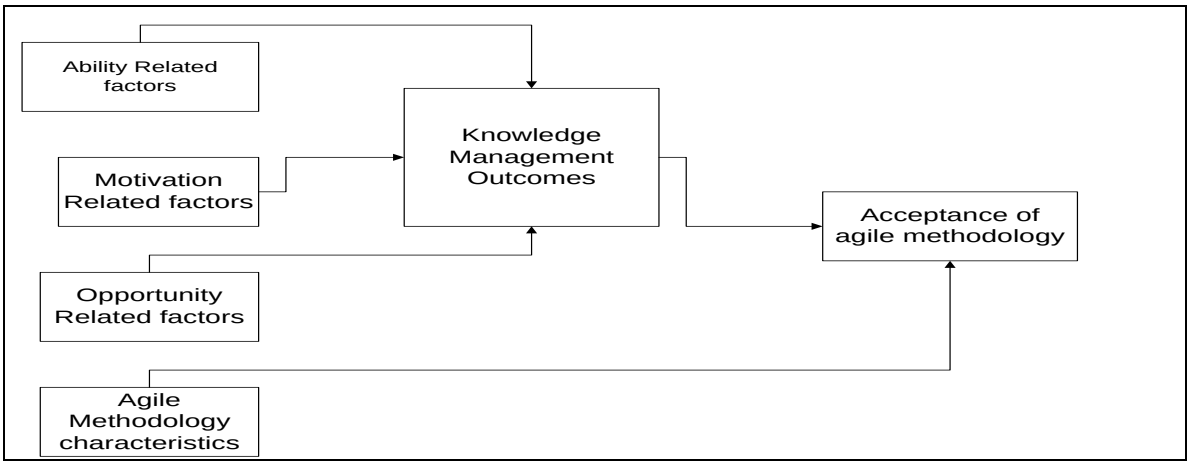

Figure 2.4: Conceptual Framework [23].

yahoo a large enterprise with a $32 billion market cap and is the largest enterprise adopted agile methods when applied agile techniques in yahoo setup it showed the good result but it is also clear that patience is necessary thing to focus is on as it will be not an easy task to migrate from traditional water flow process to the agile at once [29].

A lot of the questions are still unanswered relating to the agile scalability and adoptability, researchers are advising to adopt some Frameworks for scalability, but also identifying these proposed Frameworks missed the agility properties [30].

By adding different things to agile can make it flexible for large size projects such as dividing the large size teams into smaller size teams. Customers' coordination, communication and work in small projects and teams are hand able and easy to manage Scrum provides central sharing for global teams in shape of product backlog and pair. Programming is also applicable.

Developers are also working for better testing and analyzing of the agile activities. All the done code is stored into the central code saving place, such as SourceSafe or Open Source the code done and stored is validated and analyzed.

For each individual project there should be saving depository and a common place also to share all the projects done each project deliverable will be in small portions so that can be coordinated with other parts of the projects . At the end of every month the Scrum will be entered into the final integration testing phase and all the done parts of project will be integrated with each other [35].

All the large scale projects faced a lot of problems in shape of the dynamic requirements change and the time to market is also issue. In current research, instead of using Scrum's 30 days iteration, XP's 1 or 2 week iterations were used to get customers' feedback frequently. In research conducted by Deepti Mishra and Alok Mishra they discussed a presented conceptual frame work for the development process of the complex projects using agile methods.

The main entities of the frame work includes a) face to face meeting with the customer b) Analysis of information c) workshop d) Requirements repository e) preliminary architectural design f) Iterative incremental development g) testing by the customer all these phases perform some specific task such as face to face customer meeting with the customer will gather the users stories for requirements gathering analysis of information is for finalizing the requirements from that users stories workshop here is basically referring to the formal training and meeting of the staff and that also helps in finalizing the requirements from the captured stories from the customers .

Here the requirement repository concept is given which shows some central access point for all of the team members to access their data and share among other teams' members, specifically here in the frame work the central repository is storing the requirements finalized from the user's stories.

The next phase identifies the preliminary architectural design which shows how to convert the finalized requirements in to the design shape.

After the design phase is the incremental and iterative development phase here the design is shaped into some develop product or some completed work shape this phase is followed by the testing by the customer here the customer is the authority to finalize the correctness of the developed module or work product.

During the testing phase the customer will also give the feedback about the product deliver and his/her feedback will be added again in the requirements phase to add the missing quality and components to the product figure 2.7 is giving the detail description of development process for complex projects [37].

Combination of the agile techniques is applied at different steps of the projects to achieve flexibility, adoptability. XP, FDD and Scrum are together applied to minimize restrictions of the large scale projects. In this research researchers are discussing about

the agile flexibility and adoptability. They are discussing a frame work for agile scalability and adoptability. The basic idea given by them is the combination of the different agile techniques. In the software process flow step they have showed the combination of the agile techniques combination. In first phase named planning they are applying combination of agile techniques XP and FDD and in planning phase they gate familiarity with the existing systems ,gathered the users stories ,develop the planned about the user stories development ,all over planning is done here with the finalized step of iteration planning .In second phase it is about development and here the SCRUM technique is applied with daily SCRUM process implementation ,along with mix of XP and FDD.

In third phase the executable release is produce with proper evaluation and here XP and SCRUM are used .The last phase is about post evaluation and review ,this is the phase which can provides requirements shocks and additional stories development and in fact these factors leads towards the updating of overall model. In this last phase the agile technique named XP is used, mix of techniques is represented in figure 2.8 in detail [44].

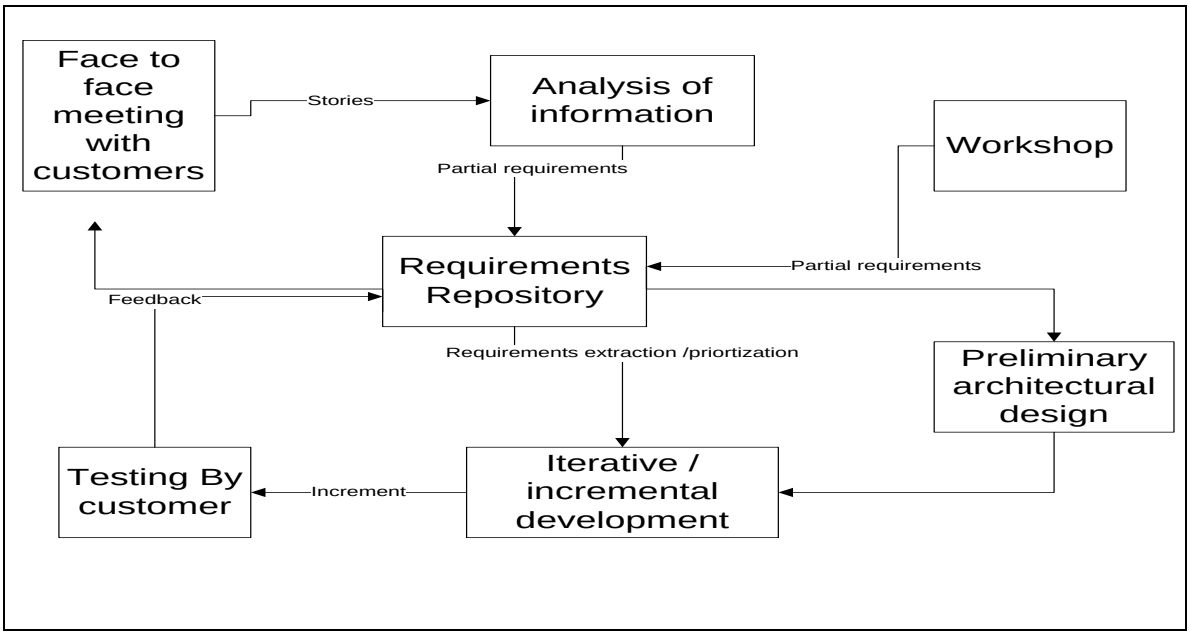

Figure 2.5: Development process for complex software project [37]

In the distributed environments and globally distributed teams the communication gap is a big limitation for agile methods adaptations and for covering the problem a

Framework was suggested that focus on the treatment of the communication problems. In paper "Agile Framework for Globally Distributed Development Environment

Researchers are talking about the distributed environments and globally distributed teams and are pointing that the communication gap is a big limitation for agile methods adaptations and for covering the problem they are suggesting a framework that focus on the treatment of the communication problems. In the model every phase is dependent on is previous phase requirements are gathered turned into the featured list, from the featured list the tasks are prioritized and after that the prioritized tasks are assigned to specific team members and the assigned tasks are transformed to the development phase by coding and testing phases .the gathered requirements and features list are properly maintained in shape of the document .The DAD model work incrementally iteration

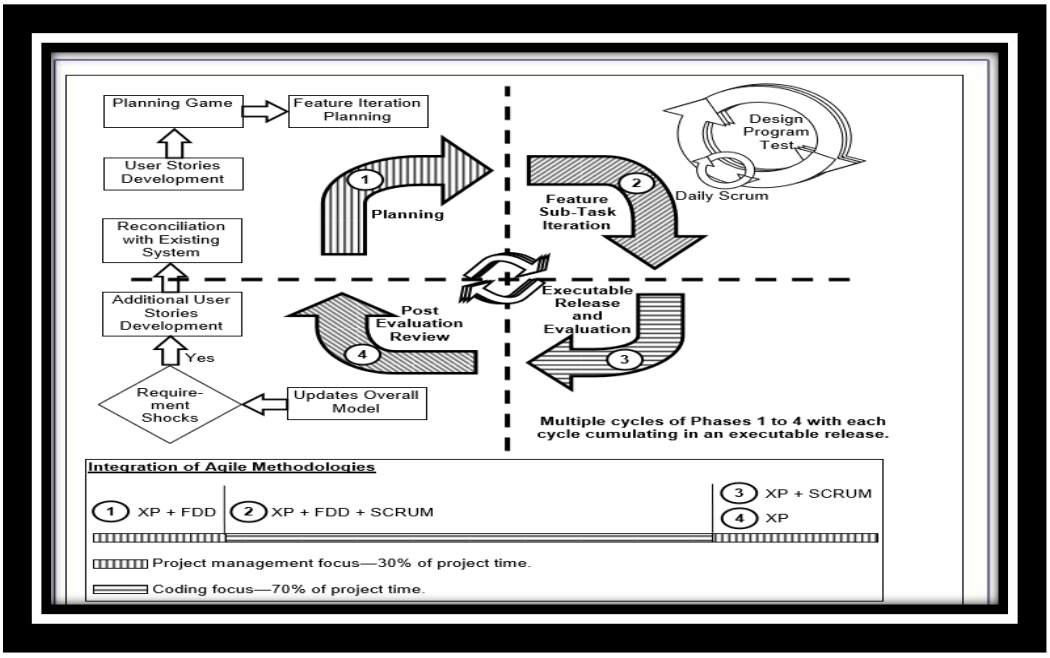

Figure 2.6: Frame work for Agility, flexibility and adoptability [44**].**

Wise. The DAD model representation is given in figure 2.9 in detail [46].

Globalization is a fact today and software industry is moving towards it and in software industry success we cannot ignore this fact .Across the boarders and limits in globularly distributed teams and in the offshore phenomenon multiple things we have to keep in mind for achieving better results from the projects. All agile methods especially

Scrum has to be purely agile when applied on the projects so that can be applied on the new domains for their survival [47].

Industry experience reflects the fact that SCRUM is used to handle the issues like communication, coordination and control in the globally distributed teams' in return there is evidence that Scrum is widely applied in small scale environments. However there are less empirical results found on the usage of the Scrum technique in large size projects or in global distributed environments. That's the problem that knowledge is missing between the research and practical implementation of the Scrum in large scale environment [47].

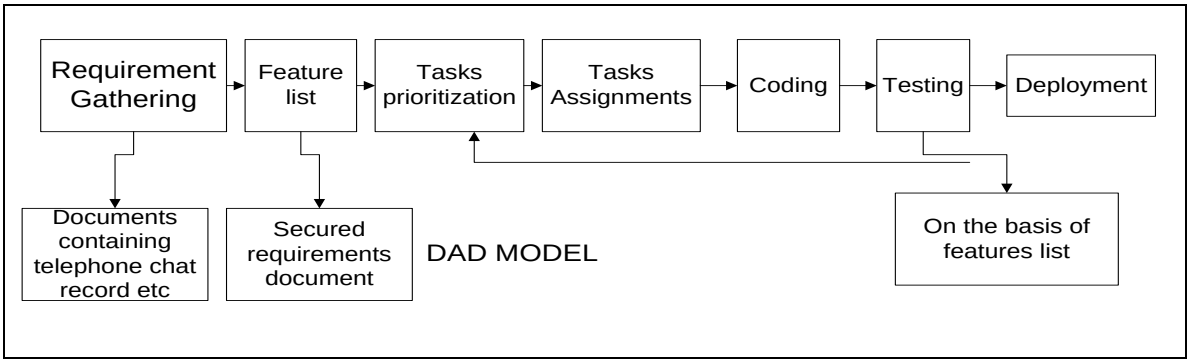

Figure 2.7: The distributed Agile Development Method [46]

Large size /scale projects in software industry are popular in IT industry. But it is truth that the heavy weight and larger size /scale projects have more complexities and problems, such as higher risk and unpredictability.

Traditional IT project development techniques and management approaches may bring complex IT projects of long time periods, complex inputs and unclear and unwanted results. Thus, it is hard to make customers' satisfaction and recognition easy in the complex IT projects. The agile methods should be applied to complex IT products. Framework for the complex IT projects can be successful for the team coordination and communication [53].

On the other hand, large and complex IT projects due to their limitations causes changes in the agility properties of the agile techniques. While talking about the complex

IT projects they required continuous changing requirements, business missions and regular improvement so agile ideology should be applied in a well-mannered way.

Currently researchers are working on modifications of the agile techniques for large size projects but facts are showing that agility has its properties for dynamic requirements, small scale /size teams /projects [53].

Issues regarding waterfall techniques are greatly elaborated in the research. But still its application in the larger projects such as aerospace is greatly in use but need some new approaches adaptations. So here is need to modify the existing techniques for better results researcher is giving a frame work from the waterfall model to the agile practices use in the Aerospace projects[54].

Researchers suggested a hybrid development method for managing larger scale projects that are expected to live for many releases. The hybrid technique employs agile for all new development work. That is the best way to create new purpose in a timely and creative way. The model is basically handling the design issues in beginning the high level requirements and project goals combine produce the development demo iterations .the demo iterations includes project detail system designs users interface design ,system design ,feature set ,data model and testing details and at the end the testing is prior to the production. Hybrid technique for applying agile at large size projects is given in detail in figure 2.11 [55].

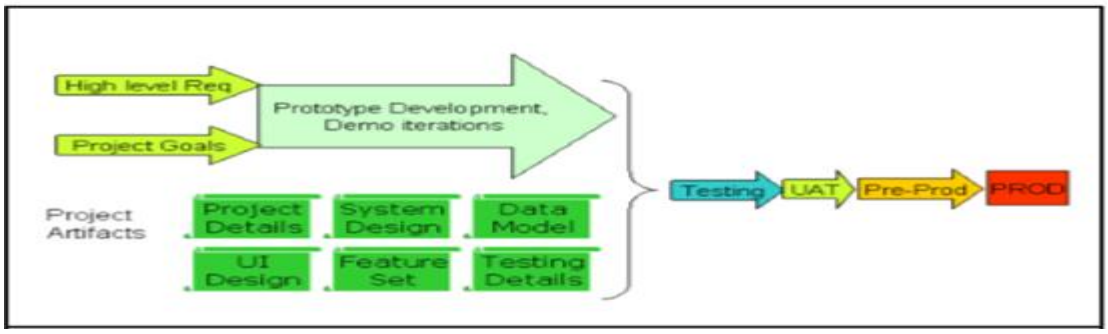

Figure 2.8: Hybrid technique for applying agile at large size projects. [55]

Researchers work for agile adoptability in large distributed projects found that on one side agile is effective for the distributed large size projects while on other hand facing risk and challenges when applied on the distributed projects, so work is done for the

identification of these risks and handling these risks to some extent dimensions of risks and their description is given in detail in table 2.4 [58].

Researcher presents a case study of a software product company that has successfully

Integrated practices from software product line engineering and agile software development.

It shows how practices from the two fields support the company's strategic and tactical ambitions, respectively. It also discusses how the company integrates strategic, tactical and operational processes to optimize collaboration and consequently improve its ability to meet market needs, opportunities and challenges. The finding Scholar from this study is relevant to software product companies seeking ways to balance agility and product management.

The findings also contribute to research on industrializing software engineering [59]. Agile adaptation is addressed and for this purpose agile adoption and improvement model is designed the model is presented in figure 2.12 [60].

Table 2.4: Risk categorization matrix (Dimensions for segregating the incidents/cases mentioned in responses). [58]

S. No	**Dimension of Risk Categorization**	**Description**
1	Risk factor title	Provisional name given to the risk
2	Risk explanation	A brief description of the risk.
3	Causes of the risk	Indicate the reason by which the risk is generated.
4	Source of the risk	Represents the origin of the risk.
5	Risk management technique	Refers to the method practitioners use to control the risks stated by them.
6	Literature support	Refers to the set of research papers, which relate to the risk factor identified from interviews.

Researchers argue that lean is a necessary progression for organizations planning to scale up agility from the project or team level to the organizational level, which agile

methods fail to address satisfyingly [61].So from it is concluded that agile and lean in future when applied together on large scale projects can produce beneficial results .

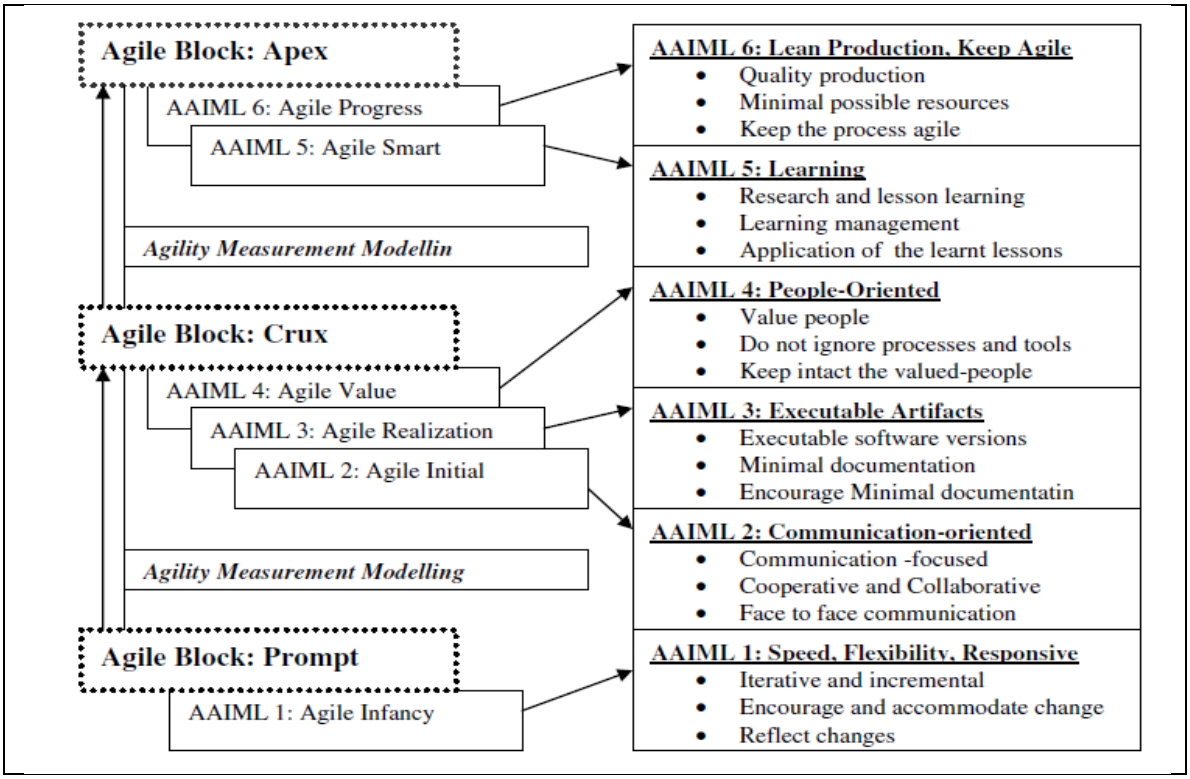

Figure 2.9: Agile adoption and improvement model [60].

2.7 Identified problem

Agile is successfully applied in small and medium size projects but showing limitations and problems when extended towards the large size projects From literature review it is also extracted that very little work is done on the current problem of agile scaling and adoption for large size projects. All these problems are motivating us for conducting research on the issue "Agile adoptability and scalability for large size projects "and suggestion of remedial Framework for the Agile adoptability and scalability.

Chapter 3 Materials and Methods

3.1 Introduction

A systematic literature review is a means of identifying, evaluating and interpreting all available research relevant to a particular research question, or topic area, or phenomenon of interest [20].

3.2 Why Systematic Literature Review

a) To sum up the existing knowledge evidence concerning to an area or topic of interest.

b) To identify gaps in current research in order to suggest areas for further investigation.

c) To provide a Framework/background in order to appropriately position new research activities.

3.3 Primary search phase

This phase involves identifying and searching through primary sources of relevant literature. In primary search automated search is used by applying the search strings on different sources. These sources include IEEE, ACM, Science-Direct and Google scholar. The primary search phase will only consider literature published from 2007 to 2014.

3.4 Secondary search phase:

The secondary search phase completes the search procedure by ensuring that the primary search phase has not missed any relevant literature.

In secondary search research questions are searched without making the search strings.

3.5 Research Questions

Research Question 1: *how to make agile approaches scalable and adoptable for large projects?*

Research Question 2: *What are the existing methods, approaches, Frameworks and practices support agile process in large scale projects?*

Research Question 3: *what are limitations of existing agile approaches, methods, Frameworks and practices with reference to large scale projects***?**

3.6 Developing a Review Protocol

After specifying research questions a review protocol is developed, this includes the definition of the following:

A. The Search Process
B. Inclusion And Exclusion Criteria
C. The Selection Process
D. The Data Extraction Process
E. Data Synthesis

3.7 The Search Process

The search process was executed through online search using search terms and resources to be searched to extract primary studies. The time period of studies selection range from 2007 to 2014.Each research paper was reviewed carefully and papers that addressed agile adoptability and scalability for large scale Projects were considered potentially relevant. The researcher responsible for searching the specific digital libraries applied the detailed inclusion and exclusion criteria to the relevant papers.

3.8 Identifying Search Terms Process

The following search strategy is used for the construction of search terms.

a) The research questions for the derivation of major terms, by identifying population, intervention and outcome
b) For these major terms, find the alternative spelling and synonyms
c) Verify the key words in any relevant paper
d) Use of Boolean Operators for conjunction if the database allows, in such a way, to use 'OR' operator for the concatenation of alternative spelling and synonyms whereas 'AND' for the concatenation of major terms.

3.9 Search Terms/First Step: PICOC Structure

First: Classify each research question according to the PICOC (Population, Intervention, Context, Outcome, and Comparison) structure.

Table 3.1: PICOC structure

Population	Set of articles describing the studies for Agile Adoptability and scalability.
Intervention	Solutions proposed in the literature for Agile Adoptability and scalability.
Outcome	Quantity and type of evidence related to the Agile Adoptability and scalability.
Context	Within the domain of Agile software engineering and with a focus on Agile Adoptability and scalability for large size projects.

Second: Defined synonyms for the keywords used in the questions.

Third: Build the search terms by joining the synonyms with operator OR and each element of the PICOC structure using the operator AND.

3.10 Research Questions in PICOC structure

Following PICOC structure three main research questions are developed on the basis of which the SLR process is started and conducted.

3.10.1 Research Question 1

How to make agile approaches scalable and adoptable for large scale projects?

a) **Population:** Agile Approaches
b) **Intervention:** Large scale projects
c) **Outcome:** parameters for scalability and adoptability

3.10.1.1 Search string /Second Step: Synonyms

a) **Population:** "Agile Approaches "; "Agile software development approaches "; "Agile software development techniques "; "Agile software engineering"; "Agile software methodologies "; "Agile software engineering methodologies "; "Agile software development approaches "; "Agile software engineering methods "; "Agile software engineering methodologies "; "Agile software development processes "; "Agile software engineering practices".
b) **Intervention:** "Large scale Projects"; "Big scale Projects"; "Vast scale Projects"; "Large size Projects"; Big size Projects"; vast size Projects".
c) **Outcome:** "parameters for scalability and adoptability"; "methods for scalability and adoptability"; ways for scalability and adoptability "; strategies for scalability and adoptability".

3.10.1.1.1 Search String Used for Primary Studies for Research Question 1

Database	Search String
IEEE	(((""Agile Approaches "OR "Agile software development approaches "OR "Agile software development techniques "OR "Agile software engineering "OR "Agile software methodologies "OR "Agile software engineering methodologies "OR "Agile software development approaches "OR "Agile software engineering methods "OR "Agile software engineering methodologies "OR "Agile software development processes OR "Agile software engineering practices" AND("Large scale Projects" OR "Big scale Projects " OR "Vast scale Projects" OR "Large size Projects" OR Big size Projects" OR vast size Projects") AND ("parameters for scalability and adoptability "OR "methods for scalability and adoptability "OR ways for scalability and adoptability "OR strategies for scalability and adoptability"'))

ACM	((““Agile Approaches ”OR “Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering “OR “Agile software methodologies ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale Projects” OR “Big scale Projects ” OR “Vast scale Projects” OR “Large size Projects” OR Big size Projects” OR vast size Projects”) AND (“parameters for scalability and adoptability “OR “methods for scalability and adoptability “OR ways for scalability and adoptability “OR strategies for scalability and adoptability”’))
Science Direct	((““Agile Approaches ”OR “Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering “OR “Agile software methodologies ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale Projects” OR “Big scale Projects ” OR “Vast scale Projects” OR “Large size Projects” OR Big size Projects” OR vast size Projects”) AND (“parameters for scalability and adoptability “OR “methods for scalability and adoptability “OR ways for scalability and adoptability “OR strategies for scalability and adoptability”’))
Google Scholar	((““Agile Approaches ”OR “Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering “OR “Agile software methodologies ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale Projects” OR “Big scale Projects ” OR “Vast scale Projects” OR “Large size Projects” OR Big size Projects” OR vast size Projects”) AND (“parameters for scalability and adoptability “OR “methods for scalability and adoptability “OR ways for scalability and adoptability “OR strategies for scalability and adoptability”’))

3.10.2 Research Question 2

What are the existing methods, approaches, Frameworks and practices support agile process in large scale projects?

3.10.2.1 Search string /Second Step: Synonyms

a) **Population:** “Agile Approaches ”; “Agile software development approaches ”; “Agile software development techniques ”; “Agile software engineering”; “Agile

software methodologies ”; “Agile software engineering methodologies ”; “Agile software development approaches ”; “Agile software engineering methods ”; “Agile software engineering methodologies ”; “Agile software development processes ”; “Agile software engineering practices”.

b) **Intervention:** “Large scale Projects”; “Big scale Projects”; “Vast scale Projects”; “Large size Projects”; Big size Projects”; vast size Projects”.

c) **Outcome:** “methods”; “approaches”; Frameworks “; practices”.

3.10.2.1.1 Search String Used for Primary Studies for Research question 2

Database	Search String
IEEE	((““Agile Approaches ”OR “Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering “OR “Agile software methodologies ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale Projects” OR “Big scale Projects ” OR “Vast scale Projects” OR “Large size Projects” OR Big size Projects” OR vast size Projects”) AND (“methods “OR “approaches “OR Frameworks “OR practices ”’))
ACM	((““Agile Approaches ”OR “Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering “OR “Agile software methodologies ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale Projects” OR “Big scale Projects ” OR “Vast scale Projects” OR “Large size Projects” OR Big size Projects” OR vast size Projects”) AND (“methods “OR “approaches “OR Frameworks “OR practices ”’))
Science Direct	((““Agile Approaches ”OR “Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering “OR “Agile software methodologies ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering

	practices" AND ("Large scale Projects" OR "Big scale Projects " OR "Vast scale Projects" OR "Large size Projects" OR Big size Projects" OR vast size Projects") AND ("methods "OR "approaches "OR Frameworks "OR practices "'))
Google Scholar	(((""Agile Approaches "OR "Agile software development approaches "OR "Agile software development techniques "OR "Agile software engineering "OR "Agile software methodologies "OR "Agile software engineering methodologies "OR "Agile software development approaches "OR "Agile software engineering methods "OR "Agile software engineering methodologies "OR "Agile software development processes OR "Agile software engineering practices" AND ("Large scale Projects" OR "Big scale Projects " OR "Vast scale Projects" OR "Large size Projects" OR Big size Projects" OR vast size Projects") AND ("methods "OR "approaches "OR Frameworks "OR practices "'))

3.10.3 Research Question 3

What are limitations of existing agile approaches, methods, Frameworks and Practices with respect to large scale projects?

3.10.3.1 Search string /Second Step: Synonyms

a) **Population:** "Agile Approaches "; "Agile software development approaches "; "Agile software development techniques "; "Agile software engineering"; "Agile software methodologies "; "Agile software engineering methodologies "; "Agile software development approaches "; "Agile software engineering methods "; "Agile software engineering methodologies "; "Agile software development processes "; "Agile software engineering practices".

b) **Intervention:** "Large scale Projects"; "Big scale Projects"; "Vast scale Projects"; "Large size Projects"; Big size Projects"; vast size Projects".

c) **Outcome:** "limitations of methods"; "limitations of approaches";" limitations of Frameworks ";" limitations of practices".

3.10.3.1.1 Search String Used for Primary Studies for Research question 3

Database	Search String
IEEE	(((""Agile Approaches "OR "Agile software development approaches "OR "Agile software

	development techniques ”OR “Agile software engineering “OR “Agile software methodologies ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale Projects” OR “Big scale Projects ” OR “Vast scale Projects” OR “Large size Projects” OR Big size Projects” OR vast size Projects”) AND (“ limitations of methods “OR “ limitations of approaches “OR limitations of Frameworks “OR limitations of practices ”’))
ACM	((““Agile Approaches ”OR “Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering “OR “Agile software methodologies ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale Projects” OR “Big scale Projects ” OR “Vast scale Projects” OR “Large size Projects” OR Big size Projects” OR vast size Projects”) AND (“ limitations of methods “OR “ limitations of approaches “OR limitations of Frameworks “OR limitations of practices ”’))
Science Direct	((““Agile Approaches ”OR “Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering “OR “Agile software methodologies ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale Projects” OR “Big scale Projects ” OR “Vast scale Projects” OR “Large size Projects” OR Big size Projects” OR vast size Projects”) AND (“ limitations of methods “OR “ limitations of approaches “OR limitations of Frameworks “OR limitations of practices ”’))
Google Scholar	((““Agile Approaches ”OR “Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering “OR “Agile software methodologies ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale Projects” OR “Big scale Projects ” OR “Vast scale Projects” OR “Large size Projects” OR Big size Projects” OR vast size Projects”) AND (“ limitations of methods “OR “ limitations of approaches “OR limitations of Frameworks “OR limitations of practices ”’))

3.11 Sources (Digital libraries)

The following databases are search out for primary studies:

Table 3.2: Data Sources

Sr. No.	Data base	URL
1	IEEE Explorer	http://ieeExplorer.ieee.org/
2	ACM Digital Library	http://dl.acm.org/
3	Science Direct	http://www.sciencedirect.com/
4	Google Scholar	http://scholar.google.com.pk/

3.12 Inclusion Criteria

a) Is the article available in full-text?
b) Is it a peer-reviewed article?
c) Does it contain a case study, experiment, survey, experience report, comparative evaluation and/or action research?
d) Does it report success, issues and/or failures or any type of experience concerning requirements gathering over agile adoptability and scalability for large scale Projects?
e) Is it based on research done in the Agile adoptability and scalability for large scale Projects?

f) Does it contain definitions on Agile adoptability and scalability for large scale Projects?
g) Does it introduce new and important claims about agile adoptability and scalability for large scale Projects and supporting these claims with some sort of evidence?
h) Does it identify problems and/or challenges of Agile adoptability and scalability for large scale Projects?
i) The article will be chosen if it addresses a comparison among different agile adoptability and scalability for large scale Projects techniques or an analysis of some agile adoptability and scalability for large scale Projects techniques.

3.13 Exclusion Criteria

a) The articles which don't include the terms.
b) An article with one keyword matching but has no relevance will be omitted
c) Articles which comprise less than 4 pages will be omitted
d) Masters studies not published in refereed conferences or journals
e) Reported only in PowerPoint slides or abstracts

3.14 The Selection Process

a) We have selected the studies in English language.
b) Ranging from the year 2007 to 2014.
c) Read the title, keywords and abstract from the papers found, excluding those which are not related to the research questions.
d) Studies are part of primary studies which directly narrates the Agile adoptability and scalability for large scale Projects.
e) If it satisfied the inclusion criteria the complete research paper was read.

3.15 Publication Quality assessment

The measurement of quality has been assessed on the basis provided by IMRAD structure:

a) **Introduction:** Does introduction incorporates an overview of Agile adoptability and scalability for large scale Projects?
b) **Methodology:** Is research methodology clearly defined in the research study?
c) **Results:** Were the study results presented in a clear manner?
d) Do the results help to solve requirements gathering problem within context of Agile adoptability and scalability for large scale Projects?
e) **Analysis:** How was data analyzed? What types of analysis techniques were used?
f) If the article contained a Framework, then either it has been validated in an industrial setting and/or evaluated by Academia?
g) **Discussion and/or conclusion:** Were negative findings and/or biasness properly reported? Were there any limits or restrictions imposed on the conclusions statements?

3.16 Data Extraction

The designed data extraction from for data extraction is attached in Appendix A.

3.17 Data Synthesis

Data synthesis is done after collecting and summarizing the primary studies which are usually heterogeneous studies. This heterogeneity leads to qualitative analysis. Apart from qualitative analysis there is also quantitative analysis is performed to present results more visible and after that a grounded theory approach is used to formulate a theory/statement of problem.

3.18 Search Strings Used for Primary Studies along with their sub strings

Different search engines are searched according to the search strings made for research questions for finalizing the data bases for research papers.

3.18.1 IEEE Final String for Question 1, 2, 3

IEEE1.7	((“Agile software approaches ”OR “Agile software processes “AND (“Large scale industrial Projects” OR “Big scale industrial Projects ” OR “Vast scale industrial Projects” OR “Large size industrial Projects” OR Big size industrial Projects” OR vast size industrial Projects”) AND (“methods “OR “approaches “OR Frameworks “OR practices “OR “methodologies “OR” techniques ”))
IEEE2.1	((“Agile software development processes AND (“ for large scale industrial Projects ") AND ("methods for scalability"))
IEEE3.8	((“Agile software development techniques ” AND (“ for Large scale industrial Projects AND (“ limitations of methods “OR “ limitations of approaches “OR limitations of Frameworks “OR limitations of practices ”’))

3.18.2 ACM Final String for Research Question 1, 2, 3

ACM 1	((“Agile software processes) AND (“Large scale industrial Projects” OR “Large size industrial Projects” AND (“methods “OR “approaches “OR Frameworks"))
ACM 2.1	((“Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale industrial Projects” OR “Big scale Projects ” OR “Vast scale industrial Projects” OR “Large size industrial Projects” OR Big size industrial Projects” OR vast size Projects”) AND (“methods “OR “approaches “OR Frameworks “OR practices “))
ACM 3.3	((“Agile software development techniques ” AND (“ for Large scale industrial Projects AND (“ limitations of methods “OR “ limitations of approaches “OR limitations of Frameworks “OR limitations of practices ”’))

3.18.3 Google Scholar Final String of search for Research Question 1, 2, 3

Google Scholar. 1.4	((“Agile software approaches ”OR “Agile software techniques ”OR “Agile software engineering methodologies ”OR “Agile software processes OR “Agile software engineering practices” AND (“Large scale industrial Projects” OR “Big scale industrial Projects ” OR “Vast scale industrial Projects” OR “Large size industrial Projects” OR Big size industrial Projects” OR vast size industrial Projects”) AND (“methods “OR “approaches “OR Frameworks “OR practices “))
Google Scholar 2 .1	((“Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale Projects” OR “Big scale Projects ” OR “Vast scale Projects” OR “Large size Projects” OR Big size Projects” OR vast size Projects”) AND (“methods “OR “approaches “OR Frameworks “OR practices ”’))
Google Scholar.3	((““Agile Approaches ”OR “Agile software development approaches ”OR “Agile software development techniques ”OR “Agile software engineering “OR “Agile software methodologies ”OR “Agile software engineering methodologies ”OR “Agile software development approaches ”OR “Agile software engineering methods ”OR “Agile software engineering methodologies ”OR “Agile software development processes OR “Agile software engineering practices” AND (“Large scale Projects” OR “Big scale Projects ” OR “Vast scale Projects” OR “Large size Projects” OR Big size Projects” OR vast size Projects”) AND (“ limitations of methods “OR “ limitations of approaches “OR limitations of Frameworks “OR limitations of practices ”’))

3.18.4 Science Direct Final String for Research Question 1, 2, 3

SD. 1 .2	((““Agile Approaches) OR (“Agile software development approaches) OR (“Agile software development techniques) OR(“Agile software engineering”) OR(“Agile software methodologies) OR (“Agile software engineering methodologies) OR (“Agile software development approaches) OR (“Agile software engineering methods) AND (“Large scale Projects”) OR (“Big scale Projects ”) OR (“Vast scale Projects”) AND (“methods “) OR (“approaches “) OR (“Frameworks“) OR (“practices ”’))
SD. 2.2	((“Agile software development approaches ”)OR(“Agile software development techniques ”) OR (“Agile software engineering methodologies”) OR (“Agile software development processes”) OR (“Agile software engineering practices”) AND (“Large scale Projects”) OR (“Big scale Projects”) OR (“Large size Projects”) OR (“Big size Projects”) OR (“vast size Projects”) AND (“methods “) OR (“approaches “)OR(’ Frameworks “) OR(” practices ”’))

SD. 3 .3	((“““Agile software development approaches ”) OR (“Agile software development techniques ”) OR (“Agile software engineering ”) OR(“Agile software engineering methodologies ”) OR (“Agile software development approaches ”) OR (“Agile software engineering methods ”) OR (“Agile software development processes”) OR (“Agile software engineering practices”) AND (“Large scale Projects” OR “Big scale Projects ”) OR(“Vast scale Projects”) OR (“Large size Projects”) OR(” Big size Projects”) OR (“vast size Projects”) AND (“ limitations of methods “)OR (“ limitations of approaches “) OR (” limitations of Frameworks “) OR (“limitations of practices ”’))

3.19 Numerical Results of SLR

All of data bases numerical results are given gathered from Google Scholar, IEEE, Science Direct and ACM iteration wise.

3.19.1 ACM digital library results

a) **Totalpapers**:ResearchQ1(string1+string1.1+string1.2+string1.3+string 1.4+string1.5)+ResearchQ2(string2+string2.1+string2.2+string2.3+string2.4)+Research Q3(string3+string3.1+string3.2+string3.3)
b) ResearchQ1(842+74+74+97+73+74)+ResearchQ2(392+245+307+ 280+300)+Research Q3(78+77+366+ 365)
c) Primary studies question wise: RESEARCH QUESTION.10/829 RQ2.10/249 RQ3. 10/366
d) Total ACM Papers selected after Inclusion / exclusion criteria for study : 11/1444
e) Total select title wise: 1
f) Total select abstract wise: 1
g) Total redundant studies: Nil
h) Total primary studies: 2/11

3.19.2 IEEE Digital Library Results

a) **Totalpapers**:ResearchQ1(string1+string1.1+string1.2+string1.3+string1.4+string1.5+ string1.6+string1.7+string1.8+string1.9+string1.10)+ResearchQ2(string2+string2.1+string2.2+string2.3+string2.4+string2.5+string2.6+string2.7+string2.8+string2.9+string2.10+string2.11+string2.12+string2.13+string2.14+string2.15)+ResearchQ3(string3

+string3.1+string3.2+string3.3+string3.4+string3.5+string3.6+string3.7+string3.8+string3.9 +string3.10)

b) ResearchQ1(102+2002+2002+2002+2002+2002+2002+2002+2002+2002+2002)+ResearchQ2(81+51+19+13+14+14+19+20+15+19+15+7+15+19+51+19)+Research Q3(81+59+19+ 14+14+14+21+ 14+8+8+5)

c) Primary studies question wise: RESEARCH QUESTION. 15/102 RQ2. 14/51 RQ3. 9/81

d) Total IEEE Papers selected after Inclusion / exclusion criteria for study :20/234

e) Total select title wise: 10

f) Total select abstract wise: 5

g) Total redundant studies: 4

Total primary studies: 19/20

3.19.3 Google Scholar Results

a) **Totalpapers**:ResearchQ1(string1+string1.1+string1.2+string1.3+string1.4)+ResearchQ2(string2+string2.1+string2.2+string2.3+string2.4+string2.5)+ResearchQ3(string3+string3.1+string3.2+string3.3+string3.4)

b) ResearchQ1(1+11+2+4)+ResearchQ2(125+432+128+0+0+10)+Research Q3(160+229+0+ 0)

c) Primary studies question wise: RESEARCH QUESTION. 5/18 RQ2. 18/169 RQ3. 48/390

d) Total Google Scholar Papers selected after Inclusion / exclusion criteria for study:

e) 48 /577

f) Total select title wise: 10

g) Total select abstract wise: 8

h) Total redundant studies: 3

Total primary studies: 21/49

3.19.4 Science Direct Results

a) **Totalpapers**:ResearchQ1(string1+string1.1+string1.2+string1.3)+ResearchQ2(string 2+string2.1+string2.2+string2.3+string2.4+string2.5)+ResearchQ3(string3+string3.1 +string3.2+string3.3+string3.4+ string3.5)

b) ResearchQ1(3+909+1152+187)+ResearchQ2(227+99+234+99+177)+Research Q3(1098+932+1098+932+ 397+932)

c) Primary studies question wise: RESEARCH QUESTION.15/800 RQ2. 22/850 RQ3. 38/800

d) Total Science Direct Papers selected after Inclusion / exclusion criteria for study : 30/577

e) Total select title wise: 5

f) Total select abstract wise: 2

g) Total redundant studies: 2

h) Total primary studies: 9/30

3.19.5 Overall Finding of the SLR

a) Total papers: ACM+IEEE +GOOGLE SCHOLAR + SCIENCE DIRECT

b) Total papers: 2/11+19/20+21/49 +9/30

Total primary Studies selected during SLR = 51/110

3.20 Graphical Representation of Results (SLR)

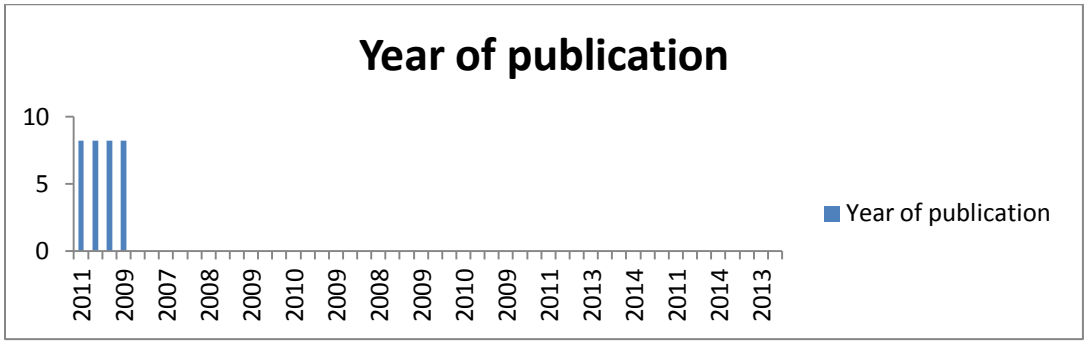

Figure 3.1: Year wise representation of primary studies

Q# 4: Source of publication

Options	Response Percentage	Responses
Journal	35%	18
Conference	61%	31
Thesis	4%	2
Unpublished	0%	0

Total Responses: 51 Answered Responses: 51 Skipped Responses: 0

Mean: 1.686 Std. Deviation: 0.542 Satisfaction Rate:22.876 Variance: 0.294 Std. Error: 0.076

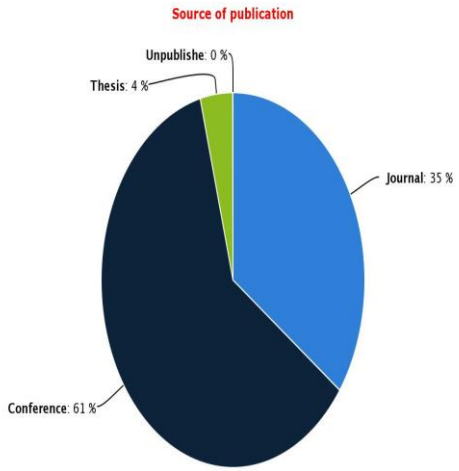

Figure 3.2: Source of publication of the paper is?

Q# 5: Publisher:

Options	Response Percentage	Responses
IEEE	37%	19
ACM	4%	2
Science Direct	18%	9
Google Schoolor	43%	22

Total Responses: 51 Answered Responses: 51 S

Mean: 2.706 Std. Deviation: 1.344 Satisfaction R

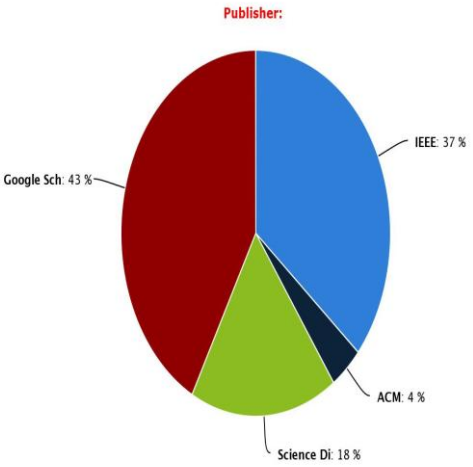

Figure 3.3: The study provides information about the publisher of the research papers

Q# 6: Study provides detailed description of agile scalability?

Options	Response Percentage	Responses
1	67%	34
0	33%	17
-1	0%	0

Total Responses: 51 Answered Responses: 51 Skipped Responses: 0

Mean: 1.333 Std. Deviation: 0.471 Satisfaction Rate:16.667 Variance: 0.222 Std. Error: 0.066

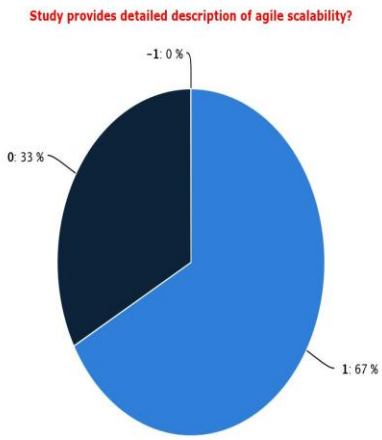

Figure 3.4: Study provides detail description of agile scalability

Q# 7: The study provides guideline as how Agile techniques are used in large size projects?

Options	Response Percentage	Responses
1	71%	36
0	27%	14
-1	2%	1

Total Responses: 51 Answered Responses: 51 Skipped Responses: 0

Mean: 1.314 Std. Deviation: 0.505 Satisfaction Rate: 15.686 Variance: 0.255 Std. Error: 0.071

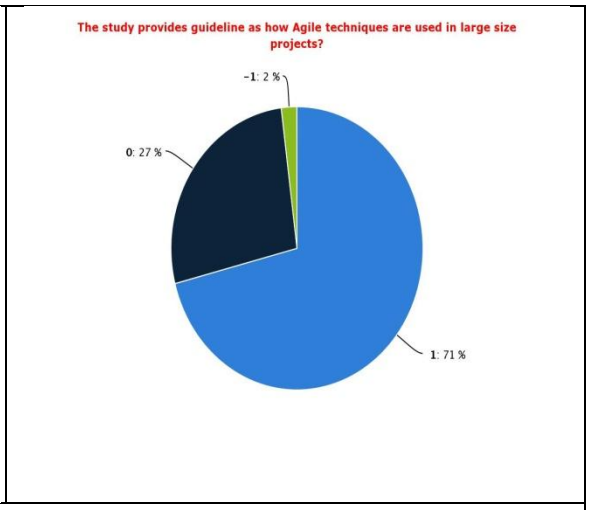

Figure 3.5: The study provides guideline as how agile techniques are used in large size projects

Q# 8: The study provides clear results after application of agile techniques in large size projects?

Options	Response Percentage	Responses
1	71%	36
0	29%	15
-1	0%	0

Total Responses: 51 Answered Responses: 51 Skipped Responses: 0

Mean: 1.294 Std. Deviation: 0.456 Satisfaction Rate: 14.706 Variance: 0.208 Std. Error: 0.064

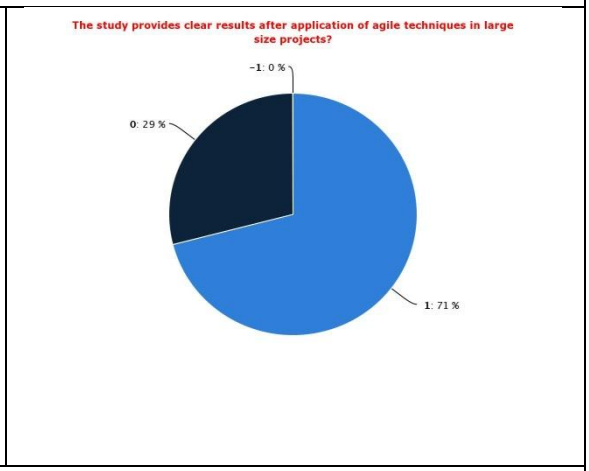

Figure 3.6: The study provides clear results after application of agile techniques in large size projects

Q# 9: The study has been published in a relevant journal or conference?

Options	Response Percentage	Responses
1	98%	50
0	2%	1
-1	0%	0

Total Responses: 51 Answered Responses: 51 Skipped Responses: 0

Mean: 1.02 Std. Deviation: 0.138 Satisfaction Rate: 0.98 Variance: 0.019 Std. Error: 0.019

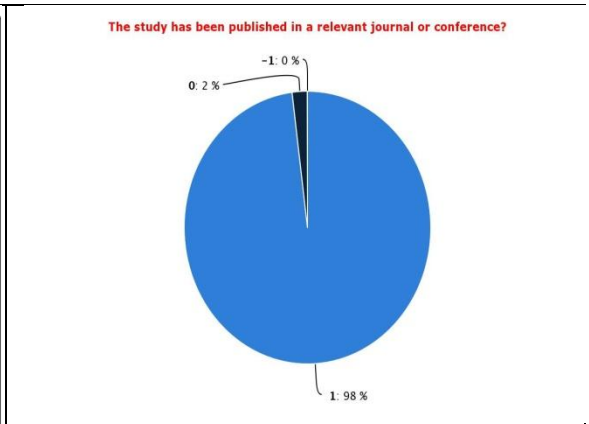

Figure 3.7: The study has been published in a relevant journal or conference

Q# 10: The study has been cited by other authors?

Options	Response Percentage	Responses
1	96%	49
0	4%	2
-1	0%	0

Total Responses: 51 Answered Responses: 51 Skipped Responses: 0

Mean: 1.039 Std. Deviation: 0.195 Satisfaction Rate:1.961 Variance: 0.038 Std. Error: 0.027

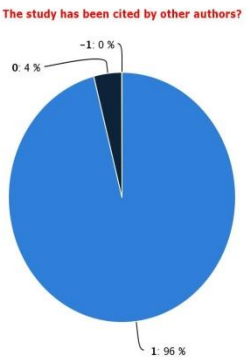

Figure 3.8: The study has been cited by authors

Q# 15: Empirical Validation of the agile techniques applied in large size projects.

Options	Response Percentage	Responses
Case study	51%	26
Experiment	6%	3
Survey	16%	8
Experience Report	22%	11
Observational Study	31%	16
Action Research	0%	0
Other	4%	2

Total Responses: 51 Answered Responses: 51 Skipped Responses: 0

Mean: 3.804 Std. Deviation: 1.996 Satisfaction Rate:32.323 Variance: 3.986 Std. Error: 0.246

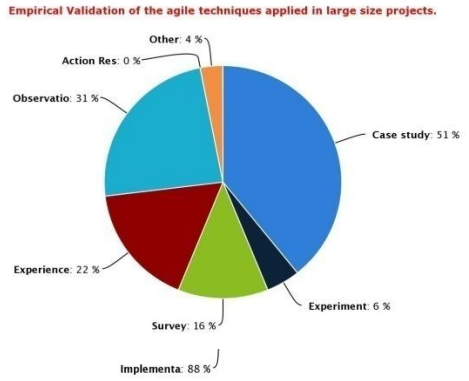

Figure 3.9: Agile phases where agile techniques are used for scalability

Q# 13: Data characteristics?

Options	Response Percentage	Responses
Academia	51%	26
Industry	49%	25
mixed	0%	0
Goverment	0%	0

Total Responses: 51 Answered Responses: 51 Skipped Responses: 0

Mean: 1.49 Std. Deviation: 0.5 Satisfaction Rate:16.34 Variance: 0.25 Std. Error: 0.07

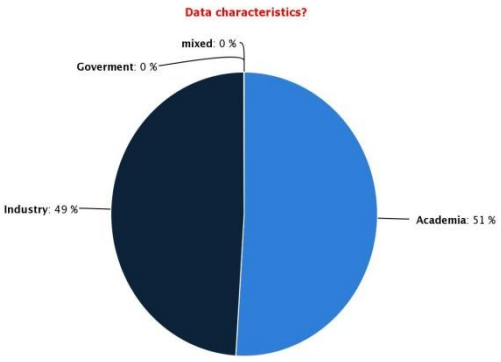

Figure 3.10: Data characteristics

Q# 14: Research detects agile limitations for large size project?

Options	Response Percentage	Responses
Documentation	24%	12
Time period	22%	11
Budget	14%	7
Human resources	14%	7
Coordination	33%	17
Distributed enviorment	25%	13
Other	65%	33

Total Responses: 51 Answered Responses: 51 Skipped Responses: 0

Mean: 9.353 Std. Deviation: 5.066 Satisfaction Rate:62.833 Variance: 25.661 Std. Error: 0.507

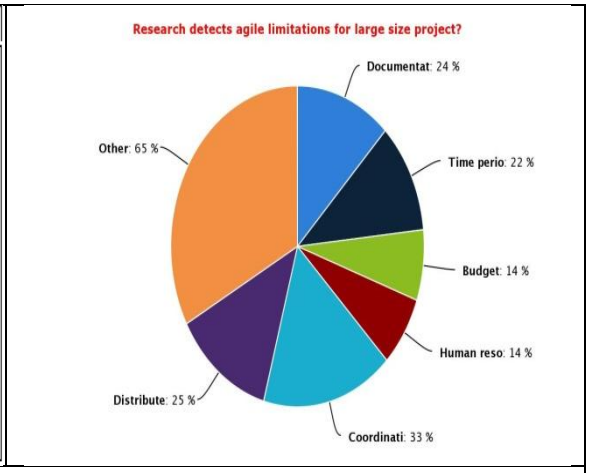

Figure 3.11: Research detects agile limitations for large size projects

3.21 Summarizing Systematic Literature Review

In this SLR performed on agile scalability and adaptability three research questions were made on basis of these three questions research strings were designed using PICOC structure to extract research papers from different data bases including IEEE, Google Scholar, ACM and Science Direct .Search protocol was designed for setting studies rules regulations to follow for summarize and concrete results after analysis.

Inclusion and Exclusion criteria was applied on these selected data bases of papers on basis of set protocol .Papers were selected iteration wise against each research question from these finalized data bases 51 papers were selected ,these selected papers were analyzed ,reviewed and data was extracted based on questions designed in data extraction form .

The gathered data is statistically analyzed and according to this analysis research papers selected for study were taken between 2009 -2011 ,out of 100 percent 35 percent papers were published in journal 61 percent in conference and 4 percent were thesis publications .

While 37 percent paper's publisher was IEEE, 4 percent ACM, Science Direct was publisher of 18 percent and 43 percent papers were published by Google Scholar.

Form all these selected papers 67 percent papers in detail describe agile adoptability and scalability, 33 percent papers partially discussed the issue .From papers that were answering about agile techniques application 71 percent studies in detail describe the agile techniques application on large scale projects,27 percent studies partially describe the techniques application and just 2 percent papers are not describing any technique at all.

In SLR 79 percent studies provide clear results of agile application on large size projects, 29 percent are partially providing results of applications, according to analysis, main factor we found was that 88 percent techniques were applied in implementation phase for agile scalability.

Research detect agile limitations for large size projects this question was answered by different researchers and 24 percent researchers said documentation is a limitation for agile practices in large scale projects,22 percent were saying about time period as a limitation for agile approaches in large scale projects,14 percent were saying about budget overflow issues in large scale projects while applied agile techniques,14 percent were talking about human resources related problems in large scale projects while applied agile approaches .

In SLR conducted 33 percent were talking about team coordination and communication issues and 25 percent were saying that distributed teams are creating limitation for agile applications in large size projects. From here we extract detail problems analysis; current strategies present and their limitations .This SLR is providing base to our proposed approach formulation.

Chapter 4 Selection of Proposed Approach

4.1 Introduction

Grounded Theory is an inductive research methodology introduced by Barney Glaser and Anslem Strauss [65]. The Grounded Theory Institute defines Grounded Theory as "the systematic generation of theory from systematic research" (Grounded Theory Institute). In this research methodology researchers gather qualitative data and systematically discover a theory derived directly from the data by performing set of systematic steps.

The reason why chosen Grounded Theory as a research method; there three reasons. First, it is appropriate to be used in areas/topics that have yet to be explored or where a new perspective might be valuable and literature doesn't cop the limitation of agile scalability and adoptability.

Second, Grounded Theory help us to study social interactions and mutual behaviors of people in the context of solving problems, whereas Agile scalability for large size projects is especially a human centered ,team size centered and intensive communication oriented process. Third, Grounded Theory boosts us up to answer the questions. In this research article grounded theory let us to formulate a theory emerge from systematic analysis of qualitative data collected by conduction of a systematic literature review. Grounded theory has been used in distributed development environment as well in distributed agile development to formulate a problem [66].

4.1.1 Grounded Theory Example Problem Statement

According to the SLR perform it is found that researchers are writing excessively about the limitations of agile approaches while talking about agile applicability to large size projects, on the same time it is also found that In software engineering, the agile

methodologies are used to handle the current challenges in software industry such as low cost, time deadlines continuously changing requirements of the market and uncertain situations and much more.

Agile is beautifully answering these challenges while applied on the small and medium level projects, but it has some limitations when it is applied on large size projects. Here the general problem found is "Scalability of the agile approaches for large size projects".

4.2 Hypothesis Generation

We will take hypothesis for answering the research question found after the complete SLR, with the help of this hypothesis we will try to find the solution for the identified problem. Hypothesis generation is on the basis of the SLR studies.

4.2.1 Formulated Hypothesis

H0: Agile approaches are scalable for large size projects.

H1: Agile approaches are not scalable for large size projects.

4.3 Selection of Proposed Approach

The proposed approach for solution is chosen systematically. In first stage all the existing limitations faced by the large size projects are studied and gathered through proper SLR conduction. In second stage all the remedial work present in form of the Frameworks /models /guidelines are studied and gathered.

In third phase all these Frameworks/models/guidelines are analyzed critically so that can highlight the existing remedial work deficiencies. Standard for analyzing these Frameworks /models /guidelines are the SLR analysis results in which the general limitations faced by agile techniques are:

- Co-ordination & communication

- Distributed teams
- Team size
- Documentation
- Time period

While the specific technical problems faced by the agile techniques when applied to the large size projects according to Deepti Mishra and Alok Mishra are:

- Challenges in regard to realize continuous testing
- Increased maintenance effort with an increase in the number of releases
- Management overhead due to the need for coordination between teams
- Detailed dependencies are not discovered on detailed level due to lack of focus on design
- Long requirements engineering duration, due to complex decision processes in requirements engineering
- Requirements priority lists are hard to create and maintain.
- Waiting times in the process, specifically in design waiting for requirements
- Reduction of test coverage due to shortage of projects and lack of independent testing
- Increased configuration management effort [37].

So according to the five general and nine technical specific issues the exiting work will be analyzed. After the analysis the limitations will be highlighted .After that from SLR the Agile technique (s) will be selected to proposes a solution to overcome these found limitations faced by agile techniques while applied in large size projects.

4.4 Analyzing existing agile adoptability efforts for large scale projects

Once the general and specific limitations are highlighted in second phase the existing work for agile scalability is analyzed on basis of these limitations found, the existing Frameworks /models /guidelines are analyzed one by one.

4.5 Development process for complex software project

In literature review discussed a presented conceptual frame work for the development process of the complex projects using agile methods.

The main entities of the frame work includes a:face to face meeting with the customer b:Analysis of information c:workshop d:Requirements repository e: preliminary architectural design f:Itearative incremental development g:testing by the customer all these phases perform some specific task such as face to face customer meeting with the customer will gather the users stories for requirements gathering ,analysis of information is for finalizing the requirements from that users stories ,workshop here is basically referring to the formal training Scholar and meeting of the staff and that also helps in finalizing the requirements from the captured stories from the customers .

Here the requirement repository concept is given which shows some central access point for all of the team members to access their data and share among other teams' members, specifically here in the frame work the central repository is storing the requirements finalized from the users' stories.

The next phase identifies the preliminary architectural design which shows how to convert the finalized requirements in to the design shape .After the design phase is the incremental and iterative development phase here the design is shaped into some develop product or some completed work shape this phase is followed by the testing by the customer here the customer is the authority to finalized the correctness of the developed module or work product ,during the testing phase the customer will also give the feedback about the product deliver and his/her feedback will be added again in the requirements phase to add the missing quality and components to the product [37].

4.5.1 Limitations found in the frame work for Development process

As we already discussed that the current work will be analyzed base on the five general issues and nine technical issues faced by large size projects while talking about agile scalability.

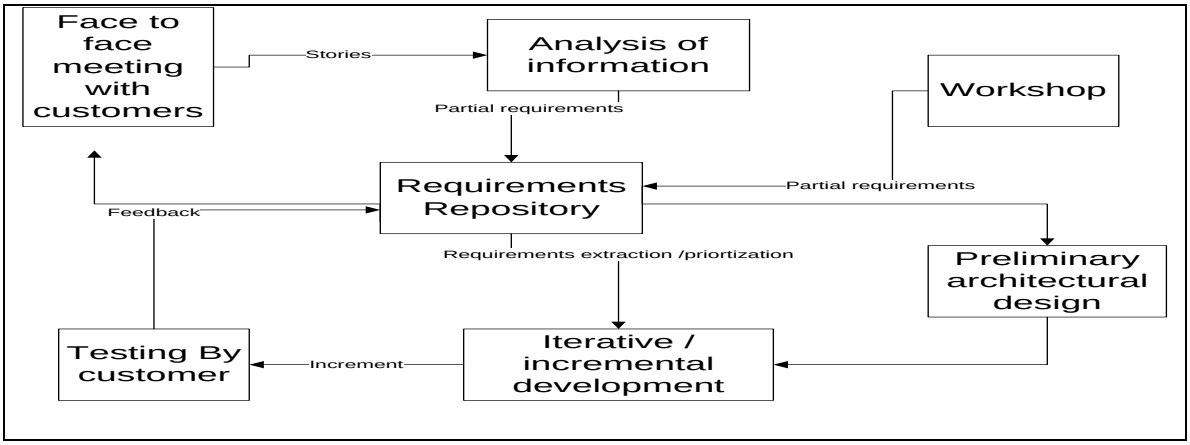

Figure 4.1: Development process for complex software project [37]

4.5.2 Analyses based on five general issues

We analyze the above mentioned frame work according to the five general issues and found some critical limitations of the frame work when applied to large size projects Limitations of the frame work are:

- If the project is of large size and the teams and customers are distributed than face to face meeting with the customer is not possible.
- If the project is of large size than the team ,size will automatically increases and it is not possible to conduct face to face meeting of the whole team at once in large set up or to conduct their workshop at once for analyzing and finalization of the requirements .
- The repository is specific just for the requirements sharing while in large size projects with greater number of the teams the sharing between teams must not only have to be of requirements but of every phase information .
- The testing is by customer all right here but what if the customer is not only one person and the customers are distributed than how the feedback will be managed and gathered and used.
- The very important thing this frame is not representing any type of the role who is going to work on which phase.

- In the frame work the communication and coordination ways and techniques are missed.

4.5.3 Analyses based on nine technical issues

The limitations found in the mentioned work based on nine technical issues are:

- The testing authority is only customer in this Framework so some technical bugs can remain unhandled and causes lack of independent testing.
- If the project is handled by distributed teams the configuration management efforts will increases.
- Maintenance efforts will automatically grow with the size of team and project.
- With increased in team size and project scale the design issues will increased and just preliminary design is not sufficient for covering the detailed designs issues
- Ambiguous coordination and communication methods cause delays and waiting times in requirements finalization.

4.6 Frame work for Agility, flexibility and adoptability

In literature review we studied about designing an Information Systems Development Course to Incorporate Agility, Flexibility, and Adaptability" the authors Chuan-Hoo Tan and Wee-Kek Tan are discussing about the agile flexibility and adoptability .They are discussing a frame work for agile scalability and adoptability.

The basic idea given by them is the combination of the different agile techniques .in the software process flow step they have showed the combination of the agile techniques combination. In first phase named planning they are applying combination of agile techniques XP and FDD and in planning phase they gate familiarity with the existing systems ,gathered the users stories ,develop the planned about the user stories development ,all over planning is done here with the finalized step of iteration planning .In second phase it is about development and here the SCRUM technique is applied with daily SCRUM process implementation ,along with mix of XP and FDD.

In third phase the executable release is produce with proper evaluation and here XP and SCRUM are used .The last phase is about post evaluation and review ,this is the phase which can provides requirements shocks and additional stories development and in fact these factors leads towards the updating of overall model. In this last phase the agile technique named XP is used.

4.6.1 Limitations found in the frame work for Agility, flexibility and adoptability

As it is already discussed that the current work will be analyzed based on the five general issues and nine technical issues faced by large size projects while talking about agile scalability.

4.6.2 Analyses based on five general issues

We analyze the above mentioned frame work according to the five general issues and found some critical limitations of the frame work when applied to large size projects .

Limitations of the frame work are:

- In planning phase the roles including in planning's not clear, customer is involved or not it is not clear.
- How the distributed teams will take part in planning it is not clear.
- How the planned activities will be stored and it is not clear.
- XP and FDD are applicable on different scenarios here the application and purpose of both application is not clear in the software development process flow.
- In second or development phase the distributed teams coordination and communication is not visible.
- Mix of SCRUM, XP and FDD is applied but the process is just representing the scrum an FDD phenomenon.

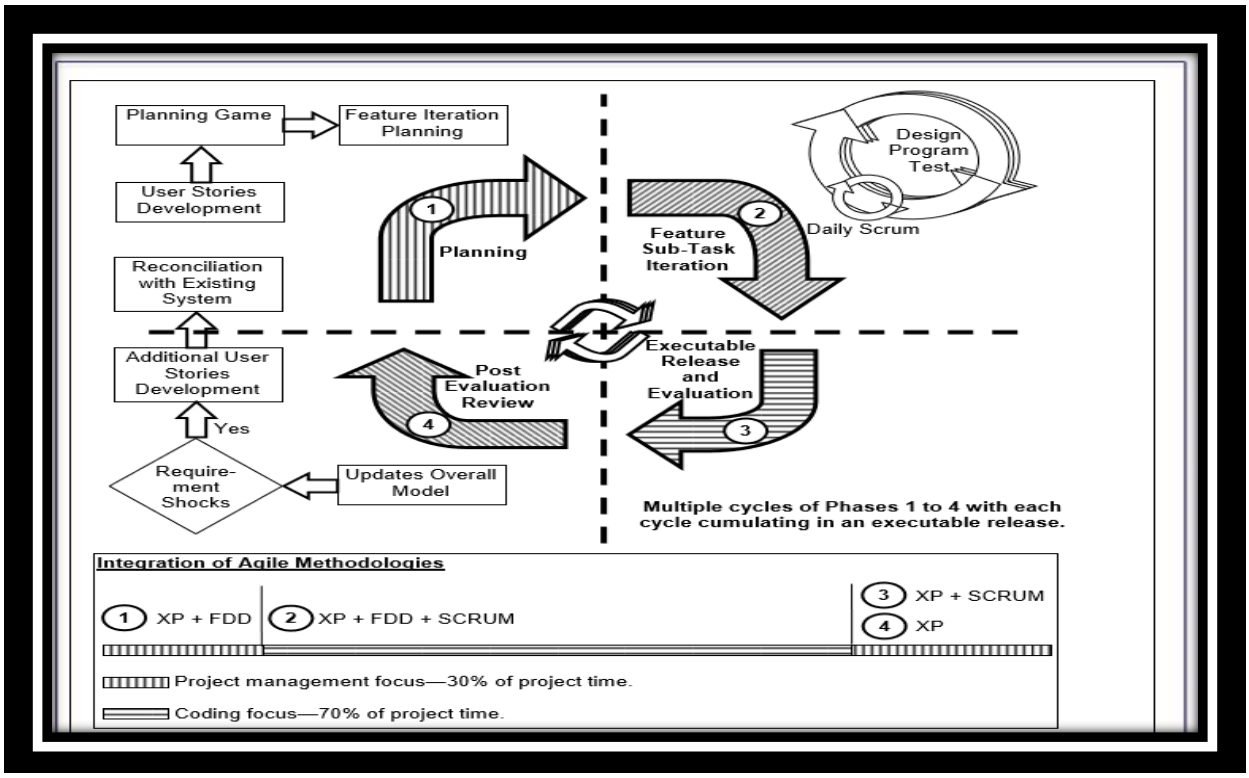

Figure 4.2: Frame work for Agility, flexibility and adoptability [44].

- How much team size is involved this Framework doesn't say anything about it.
- How much time is assigned for scrum meeting and sprint activity it is not clear from the frame work.
- Executable release time period is not visible.
- Post evaluation roles are not mentioned.
- Post evaluation for distributed team work will be maintained is not addressed.
- The post evaluation record sharing mechanism is not clear.

4.6.3 Analyses based on nine technical issues

- In the complete model is not cleared that who is responsible for the testing?
- Increased maintenance efforts with increase of number of team members and size of project are not clear how it will be handled?
- Coordination overhead is overcome how it is not visible in all frame work.
- The detailed level design is missing.

- Requirements engineering phase is basically given the name of planning with addition of features development planning.
- The planned or gathered requirements sharing and storing concept is missing.

4.7 The distributed agile development method

In literature review we studied about Agile Framework for Globally Distributed Development Environment (The DAD Model)" REHAN AKBAR, MUHAMMAD HARIS and MAJID NAEEM are talking about the distributed environments and globally distributed teams and are pointing that the communication gap is a big limitation for agile methods adaptations and for covering the problem they are suggesting a Framework that focus on the treatment of the communication problems.

In the model every phase is dependent on is previous phase requirements gathered turned into the featured list, from the featured list the tasks are prioritized and after that the prioritized tasks are assigned to specific team members and the assigned tasks are transformed to the development phase by coding and testing phases .the gathered requirements and features list are properly maintained in shape of the document. DAD model work incrementally iteration wise.

4.7.1 Analyses based on five general issues

- Model is not clearly showing the detailed coordination and communication mechanism for distributed teams.
- The model is not showing the size of distributed teams.
- Time period required for each phase or for complete process flow is not mentioned.
- The finalized documents sharing among teams are not visible.

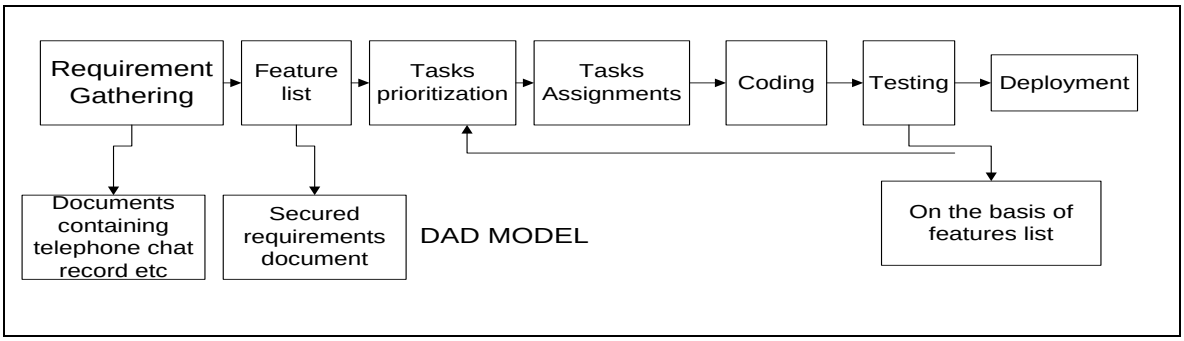

Figure 4.3: The distributed Agile Development Method [46]

4.7.2 Analyses based on nine technical issues

- Who is testing, customer is involve in testing or not? The testing roles are not clear.
- How the management overhead is managed in large setting in distributed development environment the process flow is not talking about it.
- Roles dependencies are not clear.
- The waiting time and reduction delays mechanism is not clear as in the large size projects the time waste and delays reduction is major issue.

4.8 Hybrid technique for applying agile at large size projects

In Literature review we discussed hybrid remedial approach for the large size project development. Researchers suggested a hybrid development method for managing larger scale projects that are expected to live for many releases. The hybrid technique employs agile for all new development work. That is the best way to create new purpose in a timely and creative way. [55]

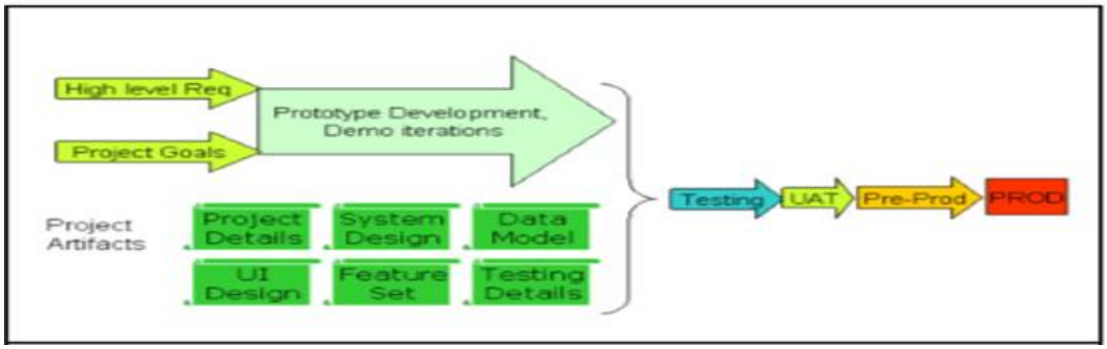

Figure 4.4: Hybrid technique for applying agile at large size projects. [55]

The model is basically handling the design issues in beginning the high level requirements and project goals combine produce the development demo iterations .the demo iterations includes project detail system designs users interface design ,system design ,feature set ,data model and testing details and at the end the testing is prior to the production .

4.8.1 Analyses based on five general issues

a. How high level requirements and project goals are gathered from the distributed teams is not mentioned.
b. Scheduling of tasks /activities is not mentioned.

4.8.2 Analyses based on nine general issues

a. Testing roles are not clear.
b. Large team's structure /distribution, scheduling of activities overhead maintenance is not clear.
c. Roles dependencies are not clear.
d. Requirements ranking and sharing all around the teams concept is week.
e. In large projects the time delay and waiting time reduction is major problem while here the time delay and waiting time reduction is not mentioned.

Table 4.1: Comparison of all Frameworks /models /guidelines with respect five general problems

FIVE GENERAL ISSUES FACED BY LARGE SIZE PROJECTS					
FRAMEWORK/MODEL	**COORDINATION & COMMUNICATION**	**DISTRIBUTED TEAMS**	**TEAM SIZE**	**DOCUMENTATION**	**TIME PERIOD**
Development process for complex software project	±	-	-	±	+
Frame work for Agility,	±	-	-	±	+

flexibility and adoptability					
The distributed Agile Development Method	±	+	-	±	+
Hybrid technique for applying agile at large size projects	±	-	-	±	+

Note regarding tables: In tables : the symbol ± represents the partial inclusion of the solution of problem in the stated Framework /model while the symbol + represents the full inclusion of the solution of problem while the symbol – represents the exclusion of the solution of problem in the stated Framework .

Table 4.2: Comparison of all Frameworks /models /guidelines with respect nine technical problems

NINE TECHNICAL ISSUES FACED BY LARGE SIZE PROJECTS				
FRAMEWORKS/MODELS				
PROBLEMS	Development process for complex software project	Frame work for Agility, flexibility and adoptability	The distributed Agile Development Method	Hybrid technique for applying agile at large size projects
Challenges in regard to realize continuous testing	±	±	±	+
Increased maintenance effort with an increase in the number of releases	-	±	-	-
Management overhead due to the need for coordination between teams	-	±	+	-

Detailed dependencies are not discovered on detailed level due to lack of focus on design	-	±	-	-
Long requirements engineering duration, due to complex decision processes in requirements engineering	±	±	-	±
Requirements priority lists are hard to create and maintain.	+	±	-	+
Waiting times in the process, specifically in design waiting for requirements	-	-	-	-
Reduction of test coverage due to shortage of projects and lack of independent testing	-	+	±	+
Increased configuration management effort	-	-	-	-

4.9 Moving towards the solution

Looking at the existing remedial work we found some Frameworks/models/guidelines for scaling the agile approaches for large scale systems along with some specific agile technique used for the implementation of these Frameworks/models/guidelines.

In analyses we also found the limitations faced by these approaches, after detail analyses of the current literature we come to the point that in large scale environments

agile two approaches are showing fruitful results while applied at large scale projects these approaches are SCRUM OF SCRUM and LEAN.

We also found that these approaches separately addressed partial problems of scalability so we in our research work combined the two approaches together for better and extended results.

4.10 Choosing Scrum Process as a base process for scalability

After conducting the detail SLR it is found that SCRUM is the technique which is applied by many researchers and organization for the scalability purpose.

SCRUM is somehow providing some inbuilt characteristics which helps the researchers to make scalable agility for the large size projects .we are not saying that all other agile techniques are failed to apply on large size projects no it is not so at all, agile others flavors are also applied for this purposed but according to our judged limitations scrum process is doing well in large setups.

4.11 Choosing scrum of scrum for large scale projects

The scrum of scrum is covering limitations of large size projects: After detailed study of scrum we found its flavor named scrum of scrum in this approach the issues of the large setups like distributed teams and large size teams are handled these two issues are handled beautifully as the large teams are divided into small teams and distributed teams are handled by help of making more project owners, scrum masters etc.

The scrum of Scrum uncovered limitations for large size projects: As already discussed scrum of scrum is answering the questions such as large team size in large size projects, and distributed teams hierarchy but leaving some limitations unaddressed.

The uncovered limitations are communication and coordination mechanism between large size and distributed teams plus the less documentation rule of scrum and early time to market for large size projects.

Scrum of scrum is also showing limitations while talk about Increased maintenance effort with an increase in the number of releases and detailed dependencies are not discovered on detailed level due to lack of focus on design along with Long requirements engineering duration, due to complex decision processes in requirements engineering and Requirements priority lists are hard to create and maintain at same time it is not handling waiting times in the process, specifically in design while waiting for requirements.

Table 4.3: Analyzing scrum of scrum technique for large size projects with respect to five general issues of large size projects

SCRUM OF SCRUM AND FIVE GENERAL ISSUES FACED BY LARGE SIZE PROJECTS					
Framework/Model	Coordination & communication	Distributed teams	Team size	Documentation	Time period
SCRUM OF SCRUM	±	-	+	±	+

Table 4.4: Analyzing scrum of scrum technique for large size projects with respect to nine technical issues of large size projects

NINE TECHNICAL ISSUES FACED BY LARGE SIZE PROJECTS	
PROBLEMS	**SCRUM OF SCRUM**
Challenges in regard to realize continuous testing	+
Increased maintenance effort with an increase in the number of releases	+
Management overhead due to the need for coordination between teams	+
Detailed dependencies are not discovered on detailed level due to lack of focus on design	-

Long requirements engineering duration, due to complex decision processes in requirements engineering	±
Requirements priority lists are hard to create and maintain.	-
Waiting times in the process, specifically in design waiting for requirements	-
Reduction of test coverage due to shortage of projects and lack of independent testing	+
Increased configuration management effort \	±

So scrum of scrum is providing us the mix results after application on large size projects, showing some scalability factors as well as some limitations.

4.12 Choosing lean for large scale projects

The second approach that we selected from the literature review is LEAN based software development according to the current work done lean is providing significant results when applied to the large size projects as it is basically originated from Toyota.

Lean production is an assembly-line methodology developed originally for Toyota and the manufacturing of automobiles. It is also known as the Toyota Production System or just-in-time production.

Lean production principles are also referred to as lean management or lean thinking. Lean Thinking is embodied by the following seven principles, first published in the book, Lean Software Development: An Agile Toolkit.

1. Eliminate waste
2. Amplify learning
3. Decide as late as possible
4. Deliver as fast as possible
5. Empower the team

6. Build integrity in team

7. See the whole

By SLR we found that all software projects in large environments faces the waste elimination problems ,the learning amplification faces problem ,decision about finalizing the thing matter a lot ,the speed of the product development matters a lot ,in large setup the team authority and empowerment matters a lot ,how the team is integrated and managed is the main concern in the large scale projects and in large projects and distributed teams see on the whole concept is very important to handle the thing accurately .

4.13 Lean based scrum of scrum concept for agile scalability and adoptability in large size projects

We identified the scrum of scrum limitations and applications for large scale projects as well we studied the lean principles made for large setup development.

We are going to proposed lean based scrum of scrum methodology for the large size projects .so that the identified five general and nine technical issues can be addressed by the extended version of lean based scrum of scrum technique.

Basically we choose the scrum of scrum as a based process for the large size projects development and we imbedded the lean principles in this process flow where required for handling the existing limitations.

We not only embed the lean concepts in scrum of scrum but also introduce the central knowledge based repository mechanism for handling the large size projects issues .so we are proposing remedial frame work for the agile scalability by the combination of lean and scrum of scrum approach.

4.14 Lean based scrum of scrum proposed frame work for agile scalability and adoptability explanation

The proposed frame work is consisting of four stages. Basically these four stages are representing the scrum of scrum internal four basic process flow sequence.

Product backlog phase : In this stage the scrum method first phase is executed named product backlog, here the combination of scrum and lean is applied as in beginning input from executive team ,customer ,stakeholder and end users is gathered the lean waste reduction principle will be apply by removing slow or poor communication. Delays and waiting time reduction will be also eliminated to speed the communication process .The information gathering and principle application will be done by the product owner. With multiple and diversified team the communication and coordination will be performed by video conferencing.

The product owner also apply lean waste reduction principle to the gathered data and before storing it into the product back log will eliminate detail requirements from it and also insufficient requirements from product backlog in this way the product owner is applying waste reduction principle of the lean approach into scrum .

Amplified learning concept is adopted by the product owner when repeating the same process for second iteration which is also principle of the lean approach.

At the end the product backlog is stored in the central knowledge sharing repository so multiple distributed teams can easily access it without any communication, coordination and team size problem faced.

Sprint planning meeting phase: Second stage of the frame work is about the sprint meeting in which all the members (team) of scrum will be included .In sprint meeting the team selects starting at top as much as it can commit to deliver by end. After sprint meeting the proper documentation of finalized and plan activities is made according to the finalized requirements from the product backlog. The communication ,coordination distributed and large size team problem is solved by the video conferencing /meeting .The final document of the planned activities is stored on the central knowledge

repository and from here is accessed by all teams and members ,this phenomena introduce the lean approach in scrum process flow and enhance the knowledge sharing ,team integrity and empowerment and on the same time reduce waiting time and time delays .All these principles help the agile scalability for large size projects .

Sprint Backlog phase: The third stage is sprint backlog. In sprint backlog the cut down chart is maintained here the selected stories after spring meeting is shaped into a cut down chart showing the roles, responsibilities, detailed time period of each schedule activity .In the beginning will Eliminate Highly Detailed stories /insufficient stories and remaining will be store in the sprint backlog .cut down chart is stored in the central knowledge sharing repository.

Sprint Review phase: This stage is starting from the sprint daily meeting this meeting will depends on the amount of activities completed, in this meeting the sprint review phase is also included in which some shaped product demo will be presented and the complete and incomplete features are separated the completed features will give some finished product while the incomplete features will be transferred back to the product backlog. A retrospective phase is also present after the product demo in this phase all the shortcomings are observed and saved ,limitation are judged and for future its remedies are also planned .Sprint retrospective is also store in central repository and sprint review demos are conducted by video conferencing if the teams are divided and distributed .

Table 4.5: Analyzing proposed Framework for agile scalability for large size projects with respect to five general issues of large size projects

LEAN BASED SCRUM OF SCRUM FRAMEWORK AND FIVE GENERAL ISSUES FACED BY LARGE SIZE PROJECTS					
Framework/Model	**Coordination & communication**	**Distributed teams**	**Team size**	**Documentation**	**Time period**
LBSOS FRAMEWORK	+	+	+	+	+

Table 4.6: Analyzing proposed Framework for agile scalability for large size projects with respect to nine technical issues of large size projects.

NINE TECHNICAL ISSUES FACED BY LARGE SIZE PROJECTS	
PROBLEMS	**LEAN BASED SCRUM OF SCRUM FRAME WORK**
Challenges in regard to realize continuous testing	+
Increased maintenance effort with an increase in the number of releases	+
Management overhead due to the need for coordination between teams	+
Detailed dependencies are not discovered on detailed level due to lack of focus on design	-
Long requirements engineering duration, due to complex decision processes in requirements engineering	+
Requirements priority lists are hard to create and maintain.	+
Waiting times in the process, specifically in design waiting for requirements	+
Reduction of test coverage due to shortage of projects and lack of independent testing	±
Increased configuration management effort \	±

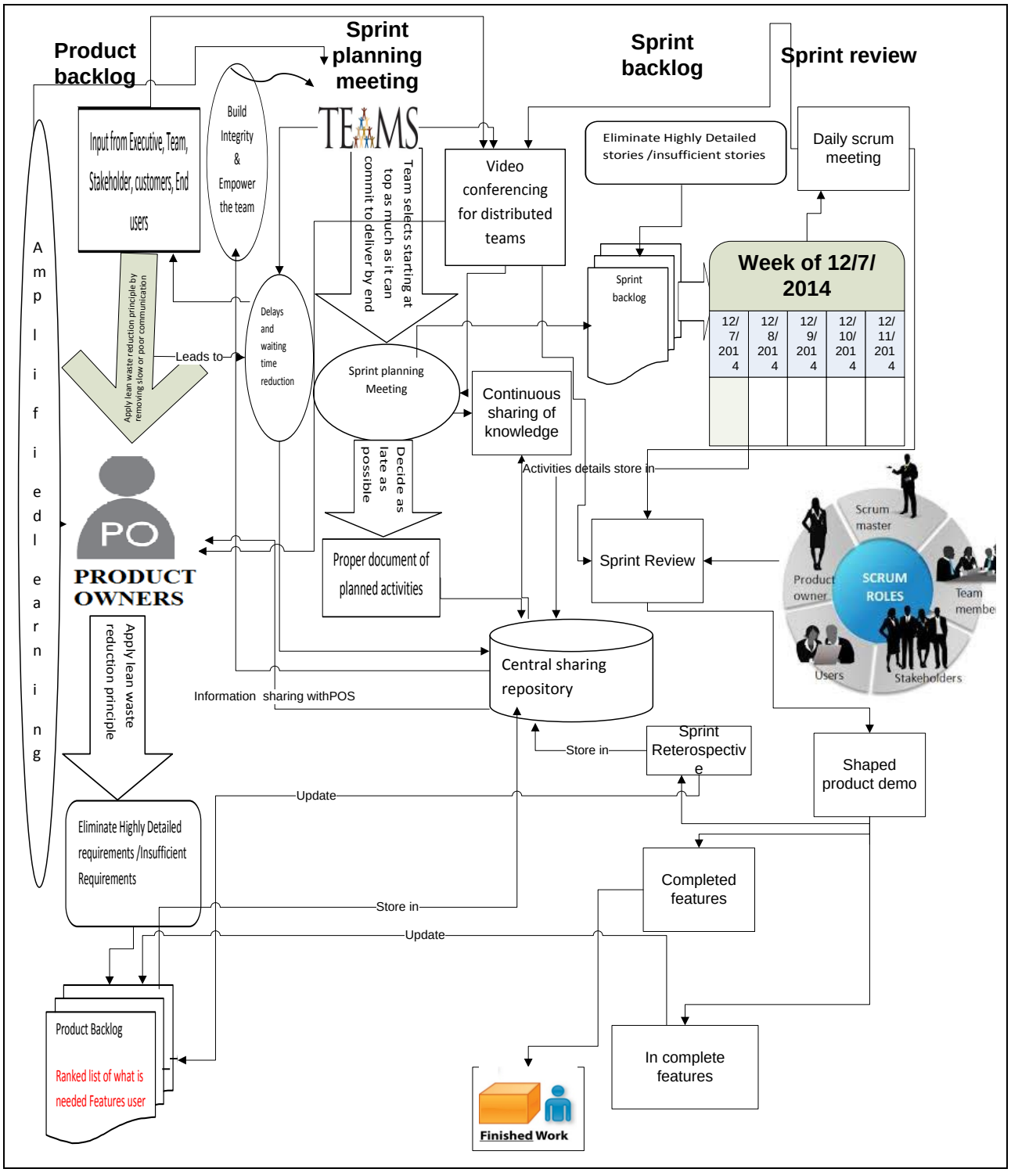

Figure 4.5: proposed frame work for Agile Scalability

4.15 Summarizing proposed Approach Methodology

Proposed approach is started by Grounded theory technique where data is gathered analyzed statistically from this analyzed data hypothesis is generated and in detail literature is reviewed for analysis of existing agile scalability techniques the existing techniques are analyzed based on five general limitations found in literature review and nine technical limitations referenced from Mishra and Mishra paper .Existing Frameworks/ models and guidelines were evaluated according to these limitations captured in SLR .

After identification of the limitations faced by already existing techniques proposed approach is stated which is based on lean based Scrum of Scrum methodology? for covering limitations of agile approaches for large size projects lean principles are imbedded in Scrum of Scrum methodology .Before finalizing the remedial approach Scrum of Scrum technique is individually analyzed based on five general and nine technical issues of agile scalability it is found that it is showing some enhance results but leaving some gaps unfilled so lean principles are imbedded in Scrum of Scrum process also central knowledge sharing repository and video conferencing concept is introduced .The new proposed idea is finalized in shape of remedial frame work named " Lean based Scrum of Scrum " Framework for agile scalability.

Proposed framework is analyzed according to five general issues faced by agile applications and nine technical issues faced by agile techniques while applied in large scale projects. It is observed that the proposed technique remains successful in handling most of the faced issues.

Chapter 5 Evaluation and Results of proposed Framework

5 Expert Reviews

To measure the credibility of the proposed Framework, it has to be verified by an appropriate evaluation method. Here we have selected expert review method to check out the credibility of the proposed Framework. This approach is suggested for our proposed Framework validation by Prof. Dr. Zahid Hussain Dean, Faculty of Science Quaid-e Awam University, Nawabshah, Pakistan and is Research Associate for Agile Scalability Institute for Software Technology,Technical University Graz,Austria and Dr. Sara Shahzad Assistant Professor Department of Computer Science University of Peshawar Pakistan and is Research Associate for Agile Scalability, Institute for Software Technology, Technical University Graz, Austria.

We ask experts to rate the Framework upon some certain parameters that were finalized in proposed methodology section. Framework was presented in Industry including different organizations dealing with large size projects such as NADRA, MTBC, UNO (UNIDO) and Pakistan Atomic energy Commission. Session of 3-4 weeks was taken by this organization for the evaluation of the proposed Framework.

Theses mentioned organizations trained software experts having more than 4 years' experience,5-10 years' experience, and 10-15 and more than 15 years' experience with roles like project managers, Directors, Assistant Directors, team leads evaluated the frame work.

This was all about industry we also evaluated our Framework from educationalist and researchers in which the important researchers and educationalists were from University of Manchester London, Institute for Software Technology, Technical University Graz, Austria, University Institute of Information Technology Arid agriculture Rawalpindi and Post Graduate College for Boys (ICB) Islamabad here it will also be added that all these educationalist were mostly having PHD degrees in software engineering and were on seniors posts in their departments having roles of Deans ,HOD,s or senior researchers .

In industry the frame work was presented directly while to foreigners' educationalists the Framework was sent by mail along with online survey created link and after on their requests the work was presented to them by Skype sessions and got their reviews.

The penal of the experts are educationist and industrialists. All of them are of five years of experience having more than five years, ten to fifteen years and more than fifteen years of experience. Some parameters are identified to rate the credibility of the Framework, survey form was designed to fill the reviews of reviewers according to Framework presented.

This form was consist of three parts ,Part one was about respondent information ,part two was about general limitations faced by agile scalability for large size projects and the third and final section was about nine technical issues faced by agile approaches while applied to large size projects .

5.1 Form Designed for Experts reviews survey

A form was designed for expert's reviews survey, first part was about validation of framework according to the five general issues identified in SLR about agile scalability and the second part was about nine technical issues of agile scalability. The Survey form designed for expert's reviews is attached in Annex B.

5.2 Statistical and Graphical results of Experts Reviews

Q# 6: Respondent's Job experience

Options	Response Percentage	Responses
less than 5 years	26%	16
5-10 years	49%	30
10-15 years	21%	13
more than 15 years	5%	3
Other	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 2.082 Std. Deviation: 0.812 Satisfaction Rate:26.21 Variance: 0.66 Std. Error: 0.103

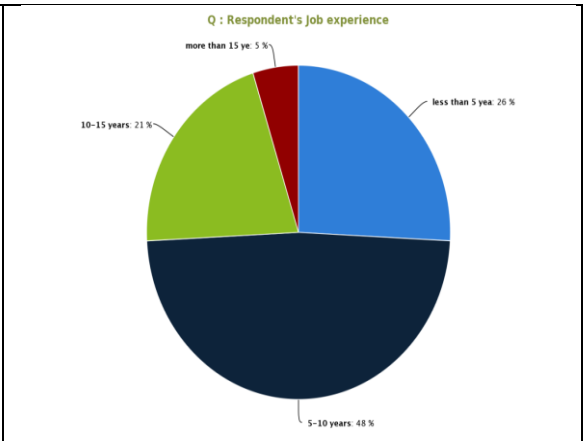

Figure: 5.1 expert reviewers' job experience

Q# 7: Respondent's Job nature

Options	Response Percentage	Responses
Project Manager	16%	10
Team Lead	26%	16
Software Engineer	48%	29
Acadamic Researcher	34%	21
Other? Please specify	21%	13

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 4.557 Std. Deviation: 1.868 Satisfaction Rate:53.09 Variance: 3.489 Std. Error: 0.198

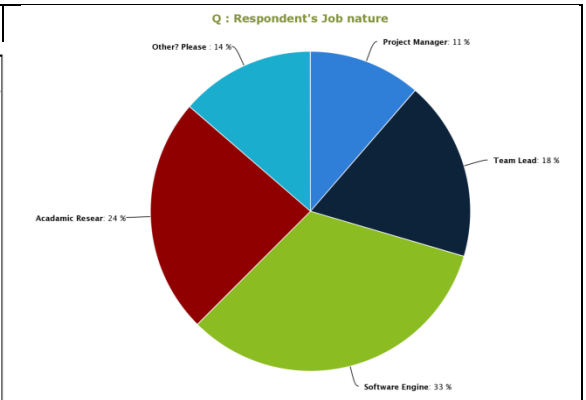

Figure: 5.2 expert reviewers' job nature

Q# 8: The framework is providing solution for Co-ordination & Comunication problems in large size projects ?

Options	Response Percentage	Responses
strongly agree	64%	39
agree	36%	22
neutral	0%	0
disagree	0%	0
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 1.361 Std. Deviation: 0.481 Satisfaction Rate:9.016 Variance: 0.231 Std. Error: 0.062

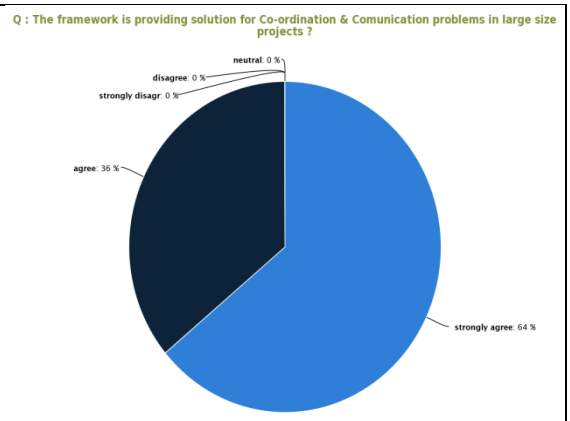

Figure: 5.3 Frameworks providing solution for communication & Coordination

Q# 9: The framework is providing solution for Distributed teams problem in large size projects ?

Options	Response Percentage	Responses
strongly agree	64%	39
agree	34%	21
neutral	2%	1
disagree	0%	0
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 1.377 Std. Deviation: 0.518 Satisfaction Rate:9.426 Variance: 0.268 Std. Error: 0.066

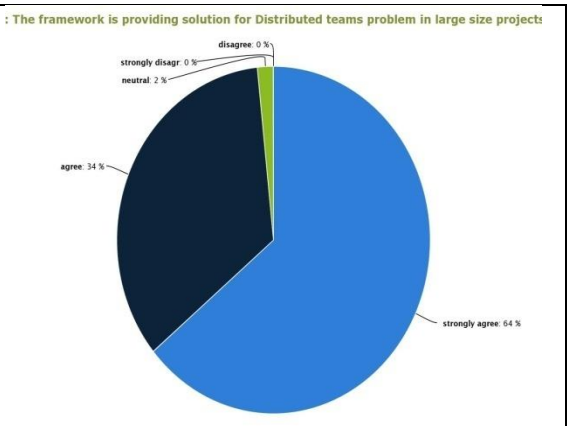

Figure: 5.4 Frameworks providing solution for Distributed teams problems

Q# 10: The framework is providing solution for Team size problems in large size projects ?

Options	Response Percentage	Responses
strongly agree	57%	35
agree	41%	25
neutral	2%	1
disagree	0%	0
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 1.443 Std. Deviation: 0.528 Satisfaction Rate:11.066 Variance: 0.279 Std. Error: 0.068

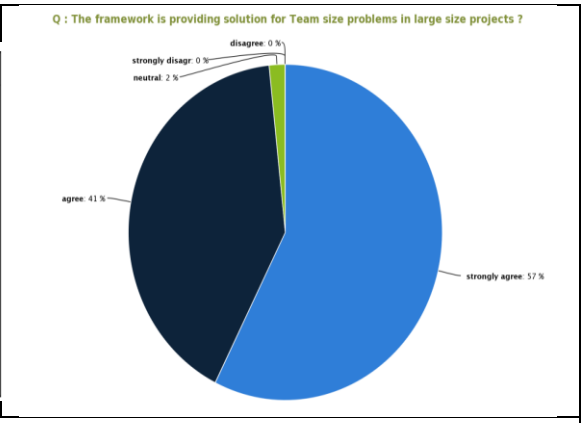

Figure: 5.5 Frameworks providing solution for team size problem

Q# 11: The framework is providing solution for documentation issues in large size projects ?

Options	Response Percentage	Responses
strongly agree	64%	39
agree	33%	20
neutral	3%	2
disagree	0%	0
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 1.393 Std. Deviation: 0.551 Satisfaction Rate:9.836 Variance: 0.304 Std. Error: 0.071

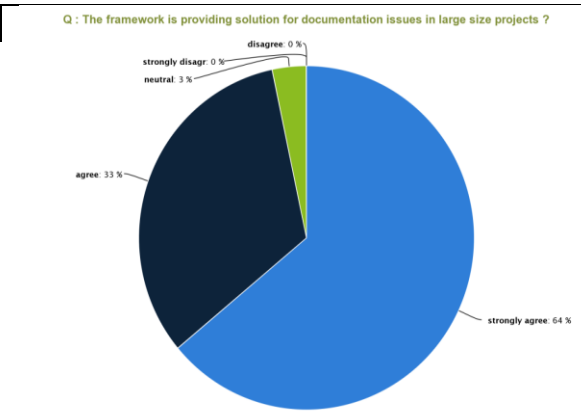

Figure: 5.6 Framework is providing solution for documentation problem

Q# 12: The framework is providing solution for early time to market issue (Time period) in large size projects ?

Options	Response Percentage	Responses
strongly agree	57%	35
agree	38%	23
neutral	5%	3
disagree	0%	0
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 1.475 Std. Deviation: 0.59 Satisfaction Rate:11.885 Variance: 0.348 Std. Error: 0.076

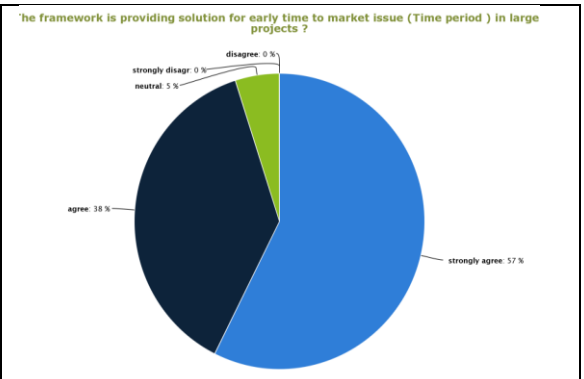

Figure: 5.7 Framework is providing solution for time to market problem

Q# 13: The framework is handling Challenges with regard to realize continuous testing?

Options	Response Percentage	Responses
strongly agree	30%	18
agree	64%	39
neutral	8%	5
disagree	0%	0
strongly disagree	0%	0

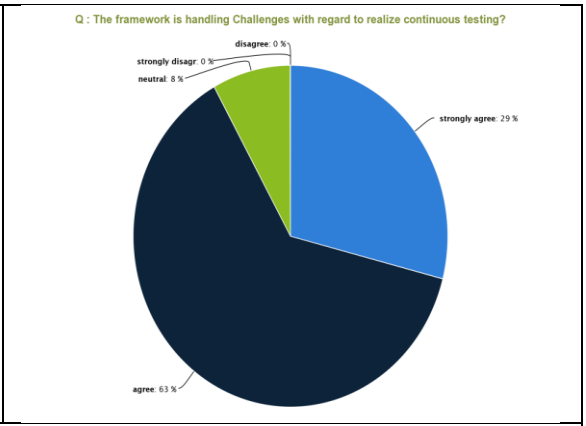

Figure: 5.8 Frameworks is handling challenges with regard to continuous testing

Q# 14: The framework is handling increased maintenance effort with an increase in the number of releases?

Options	Response Percentage	Responses
strongly agree	25%	15
agree	54%	33
neutral	21%	13
disagree	0%	0
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 1.967 Std. Deviation: 0.677 Satisfaction Rate:24.18 Variance: 0.458 Std. Error: 0.087

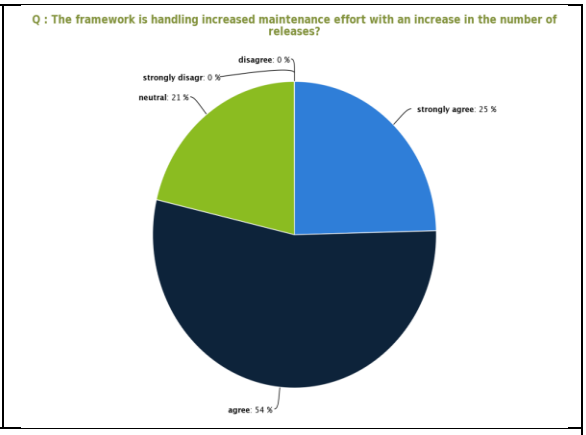

Figure: 5.9 Framework is handling increased maintenance effort with an increase in the number of releases

Q# 15: The framework is handling Management overhead due to the need for coordination between teams?

Options	Response Percentage	Responses
strongly agree	25%	15
agree	69%	42
neutral	7%	4
disagree	0%	0
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 1.82 Std. Deviation: 0.528 Satisfaction Rate:20.492 Variance: 0.279 Std. Error: 0.068

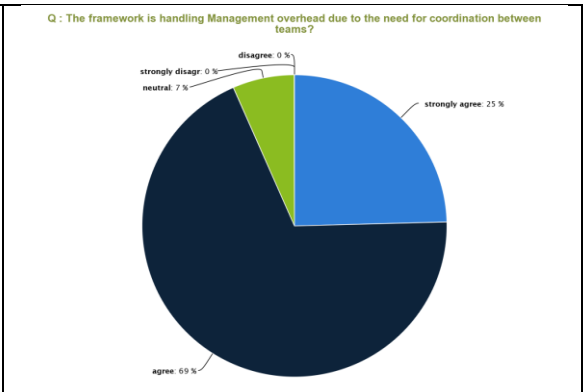

Figure: 5.10 Framework is handling management overhead

Q# 16: The framework is answering Detailed dependencies on detailed level due to good focus on design?

Options	Response Percentage	Responses
strongly agree	10%	6
agree	54%	33
neutral	31%	19
disagree	5%	3
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 2.311 Std. Deviation: 0.714 Satisfaction Rate:32.787 Variance: 0.51 Std. Error: 0.091

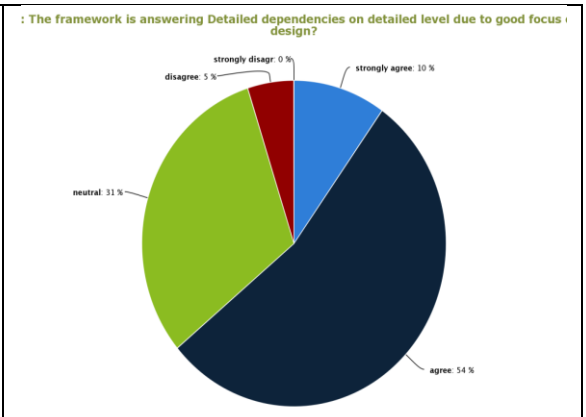

Figure: 5.11 Framework is answering detailed dependencies on detail level due to good focus on design

Q# 17: The framework is handling Long requirements engineering duration, due to complex decision processes in requirements engineering problem ?

Options	Response Percentage	Responses
strongly agree	20%	12
agree	64%	39
neutral	16%	10
disagree	0%	0
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 1.967 Std. Deviation: 0.6 Satisfaction Rate:24.18 Variance: 0.36 Std. Error: 0.077

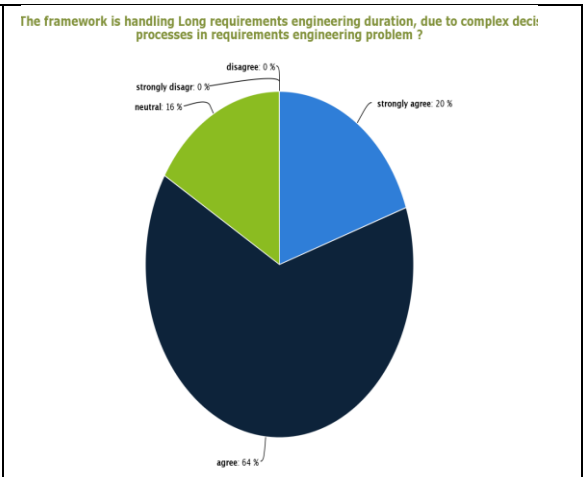

Figure: 5.12 Framework is handling long requirements engineering duration

Q# 18: The framework is handling Requirements priority lists creation and maintenance .?

Options	Response Percentage	Responses
strongly agree	13%	8
agree	84%	51
neutral	3%	2
disagree	2%	1
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 1.967 Std. Deviation: 0.472 Satisfaction Rate:23.387 Variance: 0.223 Std. Error: 0.06

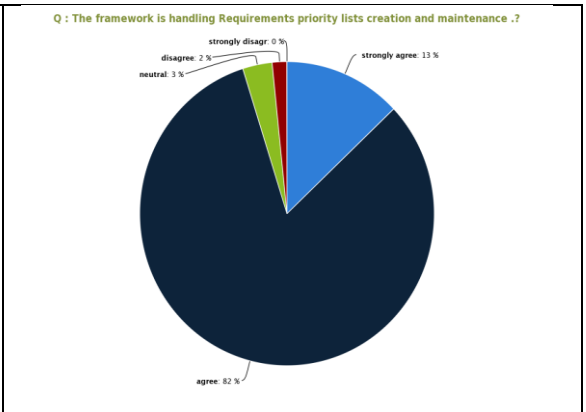

Figure: 5.13 Framework is handling requirements priority list creation and maintenance

Q# 19: The framework is reducing waiting times in the process, specifically in design waiting for requirements?

Options	Response Percentage	Responses
strongly agree	21%	13
agree	59%	36
neutral	20%	12
disagree	0%	0
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 1.984 Std. Deviation: 0.64 Satisfaction Rate:24.59 Variance: 0.41 Std. Error: 0.082

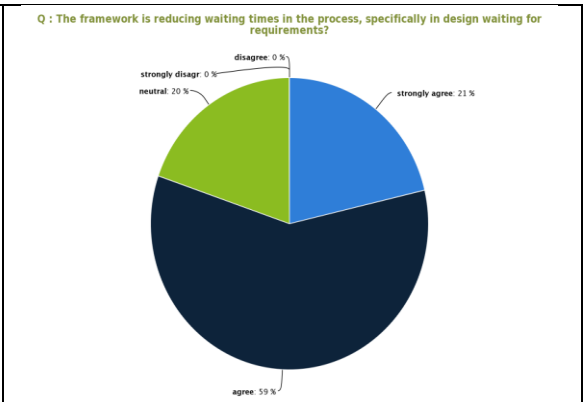

Figure: 5.14 Framework is reducing waiting times in the process

Q# 20: The framework is dealing with reduction of test coverage and independent testing problem?

Options	Response Percentage	Responses
strongly agree	20%	12
agree	66%	40
neutral	16%	10
disagree	0%	0
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 2 Std. Deviation: 0.596 Satisfaction Rate:24.194 Variance: 0.355 Std. Error: 0.076

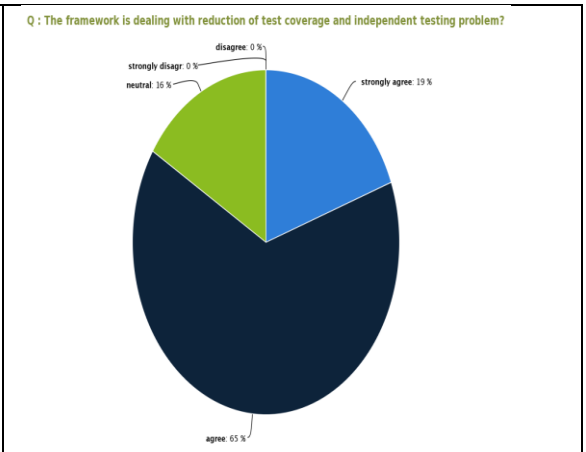

Figure: 5.15 Framework is dealing with reduction of test coverage and independent testing problem

Q# 21: The framework is giving Increased configuration management effort for large size projects ?

Options	Response Percentage	Responses
strongly agree	13%	8
agree	72%	44
neutral	16%	10
disagree	0%	0
strongly disagree	0%	0

Total Responses: 61 Answered Responses: 61 Skipped Responses: 0

Mean: 2.066 Std. Deviation: 0.539 Satisfaction Rate:25.806 Variance: 0.29 Std. Error: 0.068

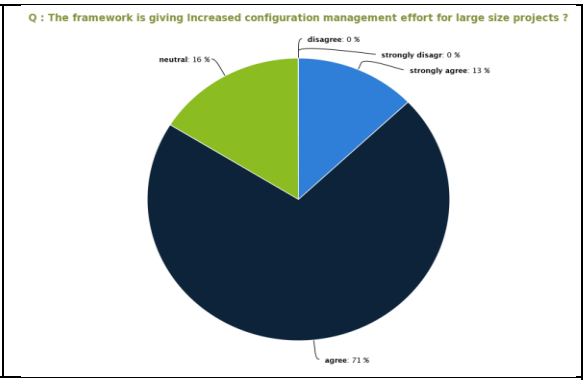

Figure 5.16 Framework is giving increased configuration management for large size projects

5.3 Summarizing Experts Reviews

61 experts have given the review after understanding the framework. After getting review from these experts, it is very easy to understand that frame work is providing solution to all five general issues faced by agile applications when applied to large scale projects as most of the experts are agreed with the framework for providing solution and remaining are strongly agreed there is no neutral or negative response for all five general limitations faced by agile techniques for large scale projects.

Here from expert reviewers review result it is cleared that communication & coordination, Team Size, distributed team problems, less documentation issue and time period issues are handled by the proposed framework for agile scalability and adoptability .While talking about the nine technical issues faced by agile techniques for large scale projects the results are also positive as majority of the experts are agreed with framework qualities for handling the issues faced by agile techniques in large projects, which depicts the general success of this framework.

In certain occasions, there is partial agreement, but the ratio of neutral or negative responses are negligible as compared to the agreed and strongly agreed responses. So, from the tables, graphs and a precise description, it is very easy to inculcate for a reader that this framework has the potential to be adapted by the industry for future endeavors. Until now, the framework has proved its worth.

Chapter 6 Conclusion and Future Work

6.1 Conclusion

The discussion on ability of agile practices to scale to "large" software development efforts has been widely debated in recent years [8]. Despite the fact that lean and agile software development has become mainstream recently, especially for larger-scale projects building complex products, the methodology leaves many architectural questions unanswered. [10].

Issues like proper documentation and training in large size projects is also a major issue while Appling the agile techniques [21].

Agile application in large size project not produced complete agility results, even agile technique name Scrum cannot be apply without making changes in it [27].

Scaling activities like scaling up agile is the problem according to the researchers the agile techniques such as Scrum and others create problems when used for large size projects [31].

Agile practices such as code refracting, minimal documentation etc. are not possible in large scale systems and here agile approaches have limitations [32].

Agile software development is about feedback and change in the large size projects if applied agile techniques continuous feedback and change will be required [33].

Peterson and Wohlin found that implementing agile and incremental practices in large-scale software development leads to benefits in one part of the process, while raising issues in another part of the process. For example, using small and coherent teams' increases control over the project, but leads to new issues at the management level where coordination of the projects has to take place [41].

The first challenge for large systems is the overall scope and complexity. Secondly large projects usually involve subcontracts with other businesses (sometimes competitors) for building or integrating substantial parts of the system. Third points about Reporting, governance, and compliance place additional burden on large projects. Fourth point is that in many large organizations, a separate group "Quality Assurance",(a.k.a. "Release Management", "Configuration Management") certifies software is fit for release, while in agility this concept is missing [52].

When talking about scalability of the agile process in literature, the two terms are scaling out and scaling up. Scaling up is concerned with using agile methods for developing large software systems that cannot be developed by a small team. Scaling out is concerned with how agile methods can be introduced across a large size projects with many years of software development experience [4].

The scaling factors will be team size, geographical distribution, Regulatory compliance, domain complexity, organizational distribution, technical complexity, organizational complexity, enterprise discipline [36].

There are many reasons that motivates towards agile techniques adoption in the large size projects. Agility provides ability to adapt quickly to change, market pressure demand short time frames and releases, ability to get instant feedback from customer, organizational process demand high quality bug free software [22].

For agile scaling work is already done to some extent researchers provided frameworks ,case studies ,experience reports and observational studies but still a gap is present that leads the researchers towards more research for designing a solid solution to the problem .

The research is also providing evidence that agile adaptation also give maturity to the organization standards and helps in the CMM levels achievement. From literature review it is also extracted that very little work is done on the current problem of agile scaling and adoption in large size projects. All these problems are motivating us for conducting research on the issue "Agile adoptability and scalability for large size projects "and suggestion of remedial framework for the Agile adoptability and scalability.

In this SLR performed on agile scalability and adaptability three research questions were made on basis of these three questions research strings were designed using PICOC structure to extract research papers from different data bases including IEEE, Google Scholar, ACM and Science Direct .Search protocol was designed for setting studies rules regulations to follow for summarize and concrete results after analysis.

Inclusion and Exclusion criteria was applied on these selected data bases of papers on basis of set protocol .Papers were selected iteration wise against each research question from these finalized data bases 51 papers were selected ,these selected papers were analyzed ,reviewed and data was extracted based on questions designed in data extraction form .

The study summarized that different researchers made efforts for agile scalability; different techniques of agile scalability are applied for covering agile limitations for scalability limitations.

The gathered data is statistically analyzed and according to this analysis research papers selected for study were taken between 2009 -2011 ,out of 100 percent 35 percent papers were published in journal 61 percent in conference and 4 percent were thesis publications .

While 37 percent paper's publisher was IEEE, 4 percent ACM, Science Direct was publisher of 18 percent and 43 percent papers were published by Google Scholar.

Form all these selected papers 67 percent papers in detail describe agile adoptability and scalability, 33 percent papers partially discussed the issue .From papers that were answering about agile techniques application 71 percent studies in detail describe the agile techniques application on large scale projects, 27 percent studies partially describe the techniques application and just 2 percent papers are not describing any technique at all.

In SLR 79 percent studies provide clear results of agile application on large size projects, 29 percent are partially providing results of applications, according to analysis,

main factor we found was that 88 percent techniques were applied in implementation phase for agile scalability.

Research detect agile limitations for large size projects this question was answered by different researchers and 24 percent researchers said documentation is a limitation for agile practices in large scale projects,22 percent were saying about time period as a limitation for agile approaches in large scale projects,14 percent were saying about budget overflow issues in large scale projects while applied agile techniques,14 percent were talking about human resources related problems in large scale projects while applied agile approaches .

In SLR conducted 33 percent were talking about team coordination and communication issues and 25 percent were saying that distributed teams are creating limitation for agile applications in large size projects.

From these results it is clear that researchers are working on agile scalability and adoptability for large size projects .They are trying to find the exiting limitations as faced by large scale projects while agile approaches are used .They are also working on remedial strategies for agile scalability problems compensation ion large projects.

Aim of this SLR conducted was performing a detail analysis of limitations of agility in large size projects and analyzing the existing remedial work and its limitations.

From here we extract detail problems analysis, current strategies present their limitations .We are statistically able to judge the problems ,their nature and effect on large size projects .This SLR is providing base to our proposed approach formulation .

After identification of the limitations faced by already existing techniques proposed approach is stated which is based on lean based Scrum of Scrum methodology? for covering limitations of agile approaches for large size projects lean principles are imbedded in Scrum of Scrum methodology .Before finalizing the remedial approach Scrum of Scrum technique is individually analyzed based on five general and nine technical issues of agile scalability it is found that it is showing some enhance results but leaving some gaps unfilled so lean principles are imbedded in Scrum of Scrum process

also central knowledge sharing repository and video conferencing concept is introduced .The new proposed idea is finalized in shape of remedial frame work named " Lean based Scrum of Scrum " framework for agile scalability.

Proposed framework is analyzed according to five general issues faced by agile applications and nine technical issues faced by agile techniques while applied in large scale projects. It is observed that the proposed technique remains successful in handling most of the faced issues.

Expert Reviews method has been chosen for validation purpose 61 experts have given the review after understanding the framework. After getting review from these experts, it is very easy to understand that frame work is providing solution to all five general issues faced by agile applications when applied to large scale projects as most of the experts are agreed with the framework for providing solution and remaining are strongly agreed there is no neutral or negative response for all five general limitations faced by agile techniques for large scale projects.

Here from expert reviewers review result it is cleared that communication & coordination, Team Size, distributed team problems, less documentation issue and time period issues are handled by the proposed framework for agile scalability and adoptability .While talking about the nine technical issues faced by agile techniques for large scale projects the results are also positive as majority of the experts are agreed with framework qualities for handling the issues faced by agile techniques in large projects, which depicts the general success of this framework.

In certain occasions, there is partial agreement, but the ratio of neutral or negative responses are negligible as compared to the agreed and strongly agreed responses. So from the tables, graphs and a precise description, it is very easy to inculcate for a reader that this framework has the potential to be adapted by the industry for future endeavors. Until now, the framework has proved its worth.

6.2 Hypothesis Truthfulness

From all this research work it is concluded that agile software engineering approaches are scalable for the large size projects and if applied the agile approaches intelligently the limitations faced by large size projects can be overcome.

6.2.1 Formulated Hypothesis Result

It is proved that H0 is true as “Agile approaches are scalable for large size projects”. And it is proved that H1 is false as it is not true that “Agile approaches are not scalable for large size projects”.

6.3 Future Work

The research work explains the Agile Scalability efforts and is proposing new framework for agile scalability for large size projects. Here the proposed approach is combination of lean and Scrum of Scrum.

Agile Scalability for large size projects can be tried by the combination of other agile techniques mix-up, agile scalability and adoptability can be used as maturity standards achievements for large size projects.

There are some other extensions possible for this research, work these include extension of the experimentation scale of the proposed framework for agile scalability and adoptability in future.

7 References

[1] Hamed,A.M.M.; Abushama,H.,(Aug. 2013). Popular agile approaches in software development: Review and analysis. Computing, Electrical and Electronics Engineering (ICCEEE), 2013 International Conference. pp.160, 166, 26-28.

[2] Miller, (2011). The Characteristics of Agile Software Processes. Granville Together Soft. ProceedinGoogle Scholar of the 39th Int'l Conf. and Exhibition on Technology of Object-Oriented Languages and Systems (TOOLS'01), 1530-2067/01 $10.00 © 2001 IEEE

[3] Dot Tudor and George A. Walte, (2006). Using an Agile Approach in a Large,Traditional Project. Proceedin Google Scholar of the conference on AGILE,

[4] Alistair Cockburn, (2001). Book Software Engineering. 1st ed.: Editors copyright Alistair Cockburn.

[5] Dan Turk and Robert France, (2002). Limitations of Agile Software Processes. in proceedingoogle scholar of the third international conference on extreme programming and flexible processes in software engineering (xp).

[6] ProjectVille T. Heikkil¨a,Maria Paasivaara and Casper Lassenius ,(2013). Scrum But,but Does It Matter? A Mixed-Method Study of the Planning Process of a Multi-Team Scrum. ACM / IEEE International Symposium on Empirical Software Engineering and Measurement.

[7] Brian Fitzgerald_,Klaas-Jan Stol_,Ryan O'Sullivan†,and Donal O'Brien† _Lero,(2013). Scaling Agile Methods to Regulated Environments: An Industry Case Study. The Irish Software Engineering Research Centre,University of Limerick,Ireland †QUMAS,Cleve Business Park,Monahan Road,Cork,Ireland.

[8] Moore,E.; Spens,J,(Aug. 2008). Scaling Agile: Finding your Agile Tribe. Agile, 2008. AGILE '08. Conference. pp.121, 124.

[9] N. Maiden, (2013). Extending Agile Processes with Creativity Techniques. IEEE Computer Society,B. Hollis ,City Univ. London,London,UK. 30, pp.: 78-84

[10] David Bustard, George Wilkie,Des Greer,(2013). The Maturation of Agile Software Development Principles and Practice. ECBS '13 Proceeding Google Scholar of the 20th Annual IEEE International Conference and Workshops on the Engineering of Computer Based Systems IEEE Computer Society Washington,DC,USA.

[11] Aziz Nanthaamornphong,Karla Morris,Damian W. I. Rouson,Hope A. Michelsen,(2013). A case study: agile development in the community laser-induced incandescence modeling environment (CLiiME). Proceeding SE-CSE '13 ProceedinGoogle Scholar of the 5th International Workshop on Software Engineering for Computational Science and Engineering. ISBN: 978-1-4673-6261-0, pp.9-18

[12] Lagerberg,L. Skude,T. ; Emanuelsson,P. ; Sandahl,K. ; Stahl,D.,(2013). The impact of agile principles and practices on large scale software development projects a multiple-case study of two projects at Ericsson. ACM / IEEE International Symposium on Empirical Software Engineering and Measurement.

[13] Ben Othmane,L.; Angin,P.; Weffers,H.; Bhargava,B.,,(2014). Extending the Agile Development Process to Develop Acceptably Secure Software, Dependable and Secure Computing,IEEE Transactions. 11 (6), pp.497, 509

[14] Antinyan,V.; Staron,M.; Meding,W.; Osterstrom,P.; Wikstrom,E.; Wranker,J.; Henriksson,A.; Hansson,J,(2014). Identifying risky areas of software code in Agile/Lean software

development: An industrial experience report. Software Maintenance,Reengineering and Reverse Engineering (CSMR-WCRE),2014 Software Evolution Week - IEEE Conference . (), pp.154, 163

[15] Scheerer,A.; Hildenbrand,T.; Kude,T.,(2014). Coordination in Large-Scale Agile Software Development: A Multiteam Systems Perspective. System Sciences (HICSS), 2014 47th Hawaii International Conference...pp.4780, 4788,

[16] Olsson,H.H.; Bosch,J.; Alahyari,H.,,(2013). Customer-Specific Teams for Agile Evolution of Large-Scale Embedded Systems. Software Engineering and Advanced Applications (SEAA), 2013 39th EUROMICRO Conference, pp.82, 89.

[17] Farrow,A.; Greene,S,(2008). Fast & Predictable A Lightweight Release Framework Promotes Agility through Rhythm and Flow. Agile, 2008. AGILE '08. Conference, pp.224, 228.

[18] Blau,B.; Hildenbrand,T,(2011). Product Line Engineering in Large-Scale Lean and Agile Software Product Development Environments - Towards a Hybrid Approach to Decentral Control and Managed Reuse. Availability,Reliability and Security (ARES),2011 Sixth International Conference ,pp.404,408 .

[19] Khan,M.I.; Qureshi,M.A.; Abbas,Q.(2010). Agile methodology in software development (SMEs) of Pakistan software industry for successful software projects (CMM framework). (ICENT), 2010 International Conference, pp.576, 580.

[20] Pearl Brereton a,*, Barbara A. Kitchenham a,David Budgen b,Mark Turner a,Mohamed Khalil c,(2006). Lessons from applying the systematic literature review process within the software engineering domain. Department of Computer Science,Durham University,Durham,Science Laboratories,South Road,Durham City,DH1 3LE,UKc Department of Computer Sciences,University of Khartoum,P.O. Box 321,Khartoum,Sudan.

[21] Subhas C. Misra,Vinod Kumar,Uma Kumar,(2009). Identifying some important success factors in adopting agile software development practices. Journal of Systems and Software. 82 pp.1869-1890

[22] Minna Pikkarainen & Outi Salo & Raija Kuusela & Pekka Abrahamsson, (2011). Strengths and barriers behind the successful agile deployment—insights from the three software intensive companies in Finland. Springer Science+Business Media.

[23] Frank K.Y. Chan,James Y.L. Thong ,(2009). Acceptance of agile methodologies: A critical review and conceptual framework. Department of Information Systems,Business Statistics and Operations Management,School of Business and Management,Hong Kong University of Science and Technology,Clear Water Bay,Kowloon,Hong Kong .

[24] Kuda Nageswara Rao,G. Kavita Naidu,Praneeth Chakka ,(2011). A Study of the Agile Software Development Methods,Applicability and Implications in Industry . International Journal of Software Engineering and Its Applications. 5.

[25] Mengjiao Shen; Wenrong Yang; Guoping Rong; Dong Shao, (2012). Applying agile methods to embedded software development: A systematic review," Software Engineering for Embedded Systems (SEES), 2nd International Workshop. Pp.30-36.

[26] Smits,H,(2007). The Impact of Scaling on Planning Activities in an Agile Software Development Center, 40th Annual Hawaii International Conference, pp.274c.

[27] Salo,O.; Abrahamsson,P,(2008). Agile methods in European embedded software development organizations: a survey on the actual use and usefulness of Extreme Programming and Scrum,Software,IET. 2 pp.58 and 64.

[28] Gabrielle Benefield, (2008). Rolling out Agile in a Large Enterprise. ProceedinGoogle Scholar of the 41st Hawaii International Conference on System Sciences.

[39] Taromirad,M.; Ramsin,R,(2008),An Appraisal of Existing Evaluation Frameworks for Agile Methodologies,Engineering of Computer Based Systems,ECBS 2008. 15th Annual IEEE International Conference and Workshop pp.418, 427.

[30] Laanti,M,(2008) Implementing Program Model with Agile Principles in a Large Software Development Organization,Computer Software and Applications,2008. COMPSAC '08. 32nd Annual IEEE International, pp.1383, 1391.

[31] Soundararajan,S.; Arthur,J.D.,(2009) A Soft-Structured Agile Framework for Larger Scale Systems Development," Engineering of Computer Based Systems,. ECBS 2009. 16th Annual IEEE International Conference and Workshop pp.187,195.

[32] E. Murphy-Hill,L. Williams,(2003),Agile software development: it's about feedback and change,North Carolina State University; Computer; 36(6): DOI: 10.1109/MC.2003.1204373 Source: IEEE Xplore,pp. 39- 43.

[33] Hajjdiab,H.; Taleb,A.S.,(2011) ,Agile adoption experience: A case study in the U.A.E, Software Engineering and Service Science (ICSESS),2011 IEEE 2nd International Conference on ,vol.,no.,pp.31,34,15-17 July 2011 doi: 10.1109/ICSESS.2011.5982247

[34] Mary Wu,(2011),Agile Method to Improve Delivery of Large-Scale Software Projects,Presented to The Faculty of the Department of General Engineering Son Jose State University.

[35] W. Ambler ,(2011). Methodologist for Agile and Lean,IBM Rational IBM agility@scale™: Become as Agile as You Can Be . 1st ed.

[36] Deepti Mishra and Alok Mishra,(2011). Complex software project development: agile methods adoption . journal of software maintenance and evolution: research and practice j. Softw. Maint. Evol.: Res. Pract. (2011) . (),pp.

[37] Reifer,D.J,(2003). XP and the CMM," Software. IEEE. 20 ,pp.e.g. 14-15.

[38] Paulk,M.C,(2011) ,Extreme programming from a CMM perspective,Software,IEEE ,vol.18,no.6,pp.19,26.

[39] Elssamadisy A,(2001),XP on a large project—A developer's view,ProceedinGoogle Scholar of XP/Agile Universe,Raleigh NC,2001.

[40] Petersen K,Wohlin C,(2009),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 2009; 82(9),pp. 1479–1490.

[41] Soundararajan S,Arthur JD,(2009). A soft-structured agile framework for larger scale systems development.. ProceedinGoogle Scholar of the 2009 16th Annual IEEE international Conference and Workshop on the Engineering of Computer Based Systems.ECBS. IEEE Computer Society: Washington DC,2009; 187195.

[42] Sidky A,Arthur J,(2007). Determining the applicability of agile practices to mission and life-critical systems. ProceedinGoogle Scholar of the 31st IEEE Software Engineering Workshop,(SEW 2007). IEEE Computer Society: Reading MA,2007; 3–12. DOI: 10.1109/SEW.2007.61.

[43]. Chuan-Hoo Tan ,Wee-Kek Tan ,Hock-Hai Teo Hock-Hai Teo ,(2013). Designing an Information Systems Development Course to Incorporate Agility,Flexibility,and Adaptability . Volume 26 Article 10 . (),pp.

[44] Jutta Eckstein ,(2009). Agile Software Development in the Large Encontro Ágil . Large Encontro Ágile. ISBN 0-932633-57-9 (),pp.233

[45] REHAN AKBAR,MUHAMMAD HARIS,MAJID NAEEM,(2008). Agile Framework for Globally Distributed Development Environment (The DAD Model) . 8th WSEAS International Conference on APPLIED INFORMATICS AND COMMUNICATIONS (AIC'08) . (),pp.

[46] Abbass Ghanbary ,(2010). Success of Agile Environment in Complex Projects. Edith Cowan University Research Online Australian Information Warfare and Security Conference.. (),pp.

[47] Begel,A.; Nagappan,N,(2007). Usage and Perceptions of Agile Software Development in an Industrial Context: An Exploratory Study. Empirical Software Engineering and Measurement,2007. ESEM 2007. First International Symposium. (),pp.255-264 ,doi: 10.1109/ESEM.2007.12 .

[48]. Isham M,(2008). Agile Architecture IS Possible You First Have to Believe. Agile,2008. AGILE '08. Conference. (),pp.484,489 ,doi: 10.1109/Agile.2008.16.

[49] John D. McGregor ,(2008). Agile Software Product Lines - A Working Session. 12th International Software Product Line Conference IEEE Computer Society Washington,DC,USA ©2008. ISBN: 978-0-7695-3303-2 ,doi:10.1109/SPLC.2008.60.

[50] Gul,E.; Sekerci,T.; Yuceturk,A.C.; Yildirim,U,(2008). Using XP in Telecommunication Software Development. Software Engineering Advances,2008. ICSEA '08. The Third International Conference . e.g. 32 (),pp.258,263. doi: 10.1109/ICSEA.2008.11

[51] Jianbin; Shi Tong; Fang Deying,(2009). A Heavy Weight IT Project Management Framework Based on Agile Theory. Management and Service Science,2009. MASS '09. International Conference . (),pp.1-5 ,doi: 10.1109/ICMSS.2009.5301737.

[52] VanderLeest,S.H.; Buter,A,(2009). Escape the waterfall: Agile for aerospace. Digital Avionics Systems Conference,2009. DASC '09. IEEE/AIAA 28th. (),pp.6 doi: 10.1109/DASC.2009.5347438.

[53] Lehman,T.J.; Sharma,A,(2011) ,Software Development as a Service: Agile Experiences,SRII Global Conference (RII),2011 Annual ,vol.,no.,pp.749,758,March 29 2011-April 2 2011 doi: 10.1109/SRII.2011.82.

[54] Read,A.; BrigGoogle Scholar,R.O,(2012) ,The Many Lives of an Agile Story: Design Processes,Design Products,and UnderstandinGoogle Scholar in a Large-Scale Agile Development Project, System Science (HICSS),2012 45th Hawaii International Conference on ,vol.,no.,pp.5319,5328,4-7 Jan. 2012 ,doi: 10.1109/HICSS.2012.684.

[55] S. Kalem,D. Donko and D. Boskovic ,(2013). Agile Methods for Cloud Computing . Faculty of Electrical Engineering,University of Sarajevo,Bosnia and Herzegovina . (),pp.

[56] Suprika V. Shrivastava ⇑,Urvashi Rathod ,(2015). Categorization of risk factors for distributed agile projects. Symbiosis Centre for Information Technology (SCIT),Symbiosis International University (SIU), Rajiv Gandhi Infotech Park,Hinjewadi,Pune 411 057,Maharashtra,India. Information and Software Technology journal homepage. 58 (),pp.

[57] Geir K. Hanssen a,b,*,Tor E. Fægri a ,(2007). Process fusion: An industrial case study on agile software product line engineering . SINTEF ICT,NO-7465 Trondheim,Norway b Norwegian University of Science and Technology,Department of Computer and Information Science,NO-7491 Trondheim,Norway . (),pp.

[58] A. Qumer,B. Henderson-Sellers ,(2008). A framework to support the evaluation,adoption and improvement of agile methods in practice. Faculty of Information Technology,University of Technology,Sydney,P.O. Box 123,Broadway 2007,Australia. (),pp.

[59] Xiaofeng Wanga,∗,Kieran Conboyb,(2012). Leagile" software development: An experience report analysis of the applicationof lean approaches in agile software development. Oisin Cawleyca Free University of Bozen/Bolzano,Italyb School of Information Systems,Technology and Management,UNSW,Sydney 2052,Australiac Lero,The Irish Software Engineering Research Centre,Ireland The Journal of Systems and Software . 85 (),pp.

[60] Stavros Stavru∗,St. Kliment Ohridski,(2014). A critical examination of recent industrial surveys on agile method usage . The Journal of Systems and Software 94 (2014) 87–97, pp.

[61] Guus van Waardenburg b,Hans van Vliet a,(2013). When agile meets the enterprise . a VU University Amsterdam,De Boelelaan 1081,1081 HV Amsterdam,The Netherlands b Avanade Netherlands,Versterkerstraat 6,1322 AP Almere,The Netherlands Information and Software Technology 55 (2013) 2154–2171.. (),pp.

[62] Jan Vlietland a,Hans van Vliet b,⇑,(2015). Towards a governance framework for chains of Scrum teams. The Netherlands b Department of Computer Science,VU University Amsterdam,Amsterdam,the Netherlands. Information and Software Technology 57 (2015) 52–65..(), pp.

8 Appendix A: Data Extraction Form

To extract the information from each primary study the following data extraction form was used:

Table 8.1: Data extraction form

Paper Title:	
Authors:	**Year of Publication:**
Reference Type: Journal/Conference/Thesis/Unpublished	**Publisher:** IEEE/ACM/Google Scholar/Science Direct
Quality Assessment	**(1) (0) (-1)**
1. Study provides detailed description of agile scalability?	
2. The study provides the guideline as how the agile techniques are used in large size projects?	
3. The study provides clear results after application of agile techniques in large size projects?	
4. The study has been published in a relevant journal or conference?	
5. The study has been cited by other authors?	
Data extraction for Questions	**Answers**
6. What methods have been employed by researchers to make Agile scalable for large size projects?	

1.1. Phase(s) of software process in which the Agile techniques are applied for scalability?	7. Requirement 8. Design 9. Implementation 10. Testing 11. Maintenance
1.2. Which Agile technique has been reported in this study?	12. Technique Name
1.3. Data characteristics	13. Academia 14. mixed 15. Industrial 16. Government
2. Research detects agile limitations for large size project?	17. Documentation 18. Time period 19. Budget 20. Human resources 21. Coordination 22. Distributed environment 23. others
3. Empirical Validation of the agile techniques applied in large size projects.	24. Case Study 25. Experiment 26. Survey 27. experience reports 28. observational study, survey 29. action research 30. No
Remarks:	

Detail of quality assessment 1-5:

Study provides detailed description of agile scalability? The possible answers to this question are: ‘‘Yes (+1)’’ if the paper provides detail description of agile scalability; ‘‘partially (0)’’ if the paper provides partial or not detail information about agile scalability; and ‘‘No (_1)’’ if the paper does not provide any information about agile scalability.

The study provides the guideline as how the agile techniques are used in large size projects? The possible answers to this question are: ‘‘Yes (+1)’’ if the paper provides information as how the agile techniques are used in large size projects; ‘‘partially (0)’’ if the paper provides partial or not detail information as how the agile techniques are used in large size projects; and ‘‘No (_1)’’ if the paper does not provide any information as how the agile techniques are used in large size projects.

The study provides clear results after application of agile techniques in large size projects?

The possible answers to this question are: ‘‘Yes (+1)’’ if the paper provides clear results; ‘‘partially (0)’’ if the paper provides partial or not detail results; and ‘‘No (_1)’’ if the paper does not provide any results.

The study has been published in a relevant journal or conference proceedings. The possible answers to this question are: ‘‘Very relevant’’ (+1), ‘‘Relevant (0)’’, and ‘‘Not so relevant (_1)’’.

This question will be rated by considering the order of relevance provided by the digital library, the CORE conference ranking (A, B and C conferences), and the Journal Citation Reports (JCR) lists.

The study has been cited by other authors. The possible answers to this question are: ‘‘Yes (+1)’’ if the paper has been cited by more than five authors; ‘‘partially (0)’’ if the paper has been cited by 1–5 authors; and ‘‘No (_1)’’ if the paper has not been cited. This question was rated by considering the Google scholar citations count.

9 Appendix B: Form Designed for Experts reviews survey

Table: 9.1 Survey form designed for experts reviews

Respondent's Name:
Respondent's Contact Email address:
Respondent's Qualification:
Respondent's Specialization:
Respondent's Organization:
Respondent's Job experience: ☐ less than 5 years ☐ 5-10 years ☐ 10-15 years ☐ more than 15 years
Respondent's Job nature ☐ Project Manager ☐ Team Lead ☐ Software Engineer ☐ Academic Researcher ☐ Other? Please specify :
Analysis of the proposed Framework, for agile scalability on the basis of five general limitations , faced by agile techniques while applied on large scale projects (Identified from SLR).
The Framework is providing solution for Co-ordination & Communication problems in large size projects? ☐ strongly agree ☐ agree ☐ neutral ☐ disagree

☐ strongly disagree

The Framework is providing solution for Distributed teams' problem in large size projects?

☐ strongly agree
☐ agree
☐ neutral
☐ disagree
☐ strongly disagree

The Framework is providing solution for Distributed teams' problem in large size projects?

☐ strongly agree
☐ agree
☐ neutral
☐ disagree
☐ strongly disagree

The Framework is providing solution for Team size problems in large size projects?

☐ strongly agree
☐ agree
☐ neutral
☐ disagree
☐ strongly disagree

The Framework is providing solution for documentation issues in large size projects?

☐ strongly agree
☐ agree
☐ neutral
☐ disagree
☐ strongly disagree

The Framework is providing solution for early time to market issue (Time period) in large size projects?

- ☐ strongly agree
- ☐ agree
- ☐ neutral
- ☐ disagree
- ☐ strongly disagree

Analysis of the proposed Framework for agile scalability on the basis of nine technical limitations, faced by agile techniques while applied on large scale projects. (Identified from SLR)

The Framework is handling Challenges with regard to realize continuous testing?

- ☐ strongly agree
- ☐ agree
- ☐ neutral
- ☐ disagree
- ☐ strongly disagree

The Framework is handling increased maintenance effort with an increase in the number of releases?

- ☐ strongly agree
- ☐ agree
- ☐ neutral
- ☐ disagree
- ☐ strongly disagree

The Framework is handling Management overhead due to the need for coordination between teams?

- ☐ strongly agree
- ☐ agree
- ☐ neutral
- ☐ disagree
- ☐ strongly disagree

The Framework is answering detailed dependencies on detailed level due to good focus on design?

- ☐ strongly agree
- ☐ agree
- ☐ neutral
- ☐ disagree
- ☐ strongly disagree

The Framework is handling Long requirements engineering duration, due to complex decision processes in requirements engineering problem?

- ☐ strongly agree
- ☐ agree
- ☐ neutral
- ☐ disagree
- ☐ strongly disagree

The Framework is handling Requirements priority lists creation and maintenance .?

- ☐ strongly agree
- ☐ agree
- ☐ neutral
- ☐ disagree
- ☐ strongly disagree

The Framework is reducing waiting times in the process, specifically in design waiting for requirements?

- ☐ strongly agree
- ☐ agree
- ☐ neutral
- ☐ disagree
- ☐ strongly disagree

The Framework is dealing with reduction of test coverage and independent testing problem?

- ☐ strongly agree
- ☐ agree
- ☐ neutral
- ☐ disagree

- [ ] strongly disagree

The Framework is giving increased configuration management effort for large size projects?

- [ ] strongly agree
- [ ] agree
- [ ] neutral
- [ ] disagree
- [ ] strongly disagree