

Towards Scrum Based Agile Framework for Global Software Development Teams

SHAGUFTA SHAFIQ*, YASER HAFEEZ*, SADIA ALI*, NAILA IQBAL*, AND MUHAMMAD JAMAL**

RECEIVED ON 19.06.2018 ACCEPTED ON 19.02.2019

ABSTRACT

Organizations are moving their workforce to guarantee matters of trade and profit where accounts are expected to be less expensive and provide better quality products with the accessibility of skilled labor. With the evident benefits, distributed teams often face challenges with reliable and efficient communication-coordination patterns that lead towards project failure. In the rapidly growing software industry, it becomes critical for any business to have better management activities to acquire right processes and practices. Agile methods are considered as a remedy for mitigating distributed teams' challenges, with their consistent focus on communication. Here, we intend to propose a tailored Scrum based solution for distributed teams that empowers anticipated communication and coordination throughout the development process. In the first phase of framework development we have evaluated the state of agile practices from both literature and industry; based on collected evidence and suggestive guidelines a scrum based agile framework was formulated. The Scrum is a project management method in Agile Methods family. Scrum is an iterative process that asks for incremental development of the product, includes cross functional development and testing teams. Evaluation results show that the proposed framework contributes positively to improve development process quality. We have performed theoretical as well as empirical evaluation to validate proposed framework. We have eight project managing rising aspects in DSD (Distributed Software Development) to evaluate the proposed framework.

Key Words: Distributed Software Development, Distributed Scrum, Agile Practices, Scrum of scrums, Global Software Development.

1. INTRODUCTION

System engineering is a set of activities, from planning to development performed by a group of people in collocated as well as in distributed settings. In Collocated development, all the team members are placed in one building or in an office where they can interact and communicate with each other to

achieve a target. In contrast, DSD has team members dispersed at different geographical places and work together to achieve a goal. DSD has become the need of time to access global market, which helps to acquire required system resources in less cost with a higher productivity rate [1-2].

Authors E-Mail: (shaguftashafiq786@yahoo.com, yasir@uaar.edu.pk, sadiaalief@gmail.com, nailaiqbal760@gmail.com, jamal@uaar.edu.pk)

* University Institute of Information Technology, Pir Mehr Ali Shah Arid Agriculture University, Rawalpindi, Pakistan.

** Department of Mathematics & Statistics, Pir Mehr Ali Shah Arid Agriculture University, Rawalpindi, Pakistan.

In today's global economy, most of the development companies have shifted towards the DSD and are continuously growing with time. However, despite the benefits it provides, it also brings numerous difficulties such as, physical separation, lack of project visibility, dialect separation, and loss of correspondence that causes unsatisfied clients and leads to project failure [3]. Many of the DSD problems have communication issues at their core because of the geographical distance. Communication could be an issue even when the teams are distributed in the same region and increases difficulties when there are time zones differences, language barriers, and cultural differences [4]. To deal with the increased risk, distributed teams must monitor the development process closely [5].

Agile methodology is an alternative to traditional software development approaches that empowers teams to develop quality product on time and within budget [6]. Agile teams follow incremental and iterative methodology by focusing on creating and delivering product features in a short period. Agile methodologies are supplanting traditional plan-driven software development methods [7]. Agile methods focus more on development activities instead of detailed documentation, develop several small versions of the same product in short cycles. The customer then reviews developed application and provides feedback to improve the system features. This helps to produce high quality products in distributed paradigm [8-9].

Agile methods have already been in-practice to develop high quality software products in collocated software development successfully. But agile practices should be tailored accordingly to integrate within the distributed settings because they are primarily being applied for small collocated teams rather than distributed teams [10-11]. Here we have proposed an integrated framework for distributed setting by tailoring the most frequently applied

agile methodology i.e. Scrum to utilize benefits and suggested strategies for higher product quality.

DSD is the development of software in situations where stakeholders are dispersed at various geographical locations. It is becoming widely acknowledged in the software industry [13]. DSD offers numerous benefits in software industries like cost reduction, access of high quality works over the fringe accessibility of staff to perform development round the clock, closeness to the market and clients. Regardless of the benefits in DSD, it has also created several inherited challenges for practitioners and analysts, like topographical separation, time zone contrast, social distinction, administration, and group building issues that lead to the communication, coordination, control, team management, knowledge management, development and system integration problems [4,12-13,14-20]. The measures should be taken to manage and mitigate these issues by monitoring the system development closely [5]. Project managers should support the project development process by ensuring project visibility, to improve product quality, motivate team members and to try to establish better communication patterns to optimize interaction between stakeholders which improves team capability [1].

Agile, is a lightweight people-oriented rather than plan-oriented software development, comprising of distinctive philosophies focused around iterative and incremental techniques where requirement advances throughout the development process [14,21]. Agile methods are constantly gaining popularity because of its continuous change management approach in the development process [17,22], and it follows the values and principles that encourage anticipated communications throughout the developmental process that increases the project success rate.

The different agile methods are Scrum, Lean, XP (Extreme Programming) etc. The Scrum method manages the projects and overcomes the issues of XP as only focus on the ongoing changes but not manage the projects. Lean optimize theoretical framework and principles for development process and ignore the team roles and communication. Therefore, for the project management, team coordination and change management during software development especially in GSD (Global Software Development) environment Scrum method is more suitable.

The Scrum is a project management method in Agile Methods family. Scrum is an iterative process that asks for incremental development of the product, includes cross functional development and testing teams. Scrum deals with risks, benefits, and opportunities by focusing on people and communication as against process and documentation [9]. Scrum activities promote good communication, and with some minor adaptation it can help disperse teams to deliver right system, at accurate time, with the right quality [24].

Scrum in DSD effectively increases visibility, flexible to modify and integrate requirements change, elects a team leader at each location, interface directly with key stakeholders due to remote reducing team members. Esteem time zone and cultural variances, acclimate effective communication scheme between remote teams and stakeholders, dependencies between team members and stakeholders and increased flexibility in the development process, provide proper training, recurrent demonstrations and retrospectives, supports early fault discovery and termination of failing products and provide documentation of every communication to produce high quality software products

From the literature, and many published case studies, we find strong evidence to the fact that agile methods and practices can be used to mitigate the problems and complexities in DSD. Researchers are trying to find out techniques and state of the art practices that enable the successful integration of scrum within the distributed environments.

Section 2 elaborates related work while section 3 explains the materials and methods. Section 4 presents the results and discussion and section 5 concludes specific research. Last of all, section 6 addresses the future work.

2. RELATED WORK

The purpose of this section is to produce literature evidence for the successful integration of agile methods in distributed environments to improve software quality.

Integration of Agile techniques with the DSD might be effective in enhancing software development quality by moderating some of its difficulties. Many researchers suggest that agile principles are helpful in minimizing DSD challenges. [1,4,25-30].

An analysis of literature by Sriram on agile procedures and GSD shows that there were a range of techniques presented using agile procedures, which were integrated, combined and tailored correctly for optimal productivity [31]. A systematic literature review [1] concluded that agile practices could be used in a distributed setting more effectively by adding suitable strategies to support GSD teams. The GSD services and resources are used throughout the world due to higher software qualities e.g. America, Sweden while DSD utilize the resources and services within country from all over the cities.

Hossain et. al. [1] develop a conceptual framework by classifying GSD risks and mitigation advice by suggesting strategies for listed risks including agile as a contemporary benefit for distributed teams. Another study examined multiple case-studies in which practitioners tried to integrate Scrum methods in DSD context to achieve leverage assistances of agile and initiate distinctive benefits over traditional method. Findings suggested that Geographical distance-based coordination challenges mitigated through synchronous and asynchronous communication tools with scrum-based activities of frequent communication and collaboration, whereas socio-cultural distance-based challenges were minimized by frequent meetings supported by scrum practices [4].

Ramesh et. al. [25] reported that agile methods can be effectively used with disperse teams and most of the scrum practices with their core agile principles support to overcome the communication issues in distributed settings. Brown et. al. [29] discussed inspiring elements of agile to support large scale development processes by comparing agile methods with plan driven conventional approaches. Study results demonstrate that agile standards and principles could lead the development process towards a quality software product.

Santos et. al. [32] Investigated the use of agile practices in GSD context and the findings suggest that agile approaches could be used to improve project cost and scope of quality product development. Melo et. al. [33] identified significant motivating factors in distributed agile teams and highlighted how agile features support distributed teams to be more productive and successful by developing an integrated team management framework. Rothman et. al. [34] Analyzed the benefits and challenges of using Agile in distributed settings

and suggested that agile can be a suitable solution for GSD and advised that measures should be taken to improve team management to accomplish the needs of distributed settings.

Agile practices serve as a performance improvement factor in distributed settings; studied articles highlight the effectiveness of agile practices in distributed settings which contribute to the project visibility, reacting to change, optimizing interaction, improving coordination, productivity and result in quality product development. As many efforts have been made to scale agile practices methods and roles to meet the challenges of distributed settings, we have utilized the literary contribution to formulate a complete process model for system development.

There is the communication gap between the team members and stakeholder of two popular paradigms i.e. GSD and DSD due to variant location of team members and stakeholder, different communication methods, improper task allocations, limited explorer to resources, different time and cultural ambiguities and no proper management of development process for earlier faults detections. Therefore, in this research study, we intend to develop a framework for distributed setting by optimizing scrum and scaling agile practices in DSD and GSD settings to achieve maximum performance and efficiency.

3. MATERIALS AND METHOD

After the review of prior work dealing with DSD challenges we present and propose a framework to improve software development in DSD using scrum. At the end, a case study is conducted to evaluate the proposed framework.

3.1 Scrum Based Agile Framework

In the SBAFe (Scrum Based Agile Framework), the overall process is divided into on-site and off-site project teams.

On-site team manages planning for the project for the forthcoming sprint(s) and provides technical and supervisory help for distributed teams as shown in Fig. 1.

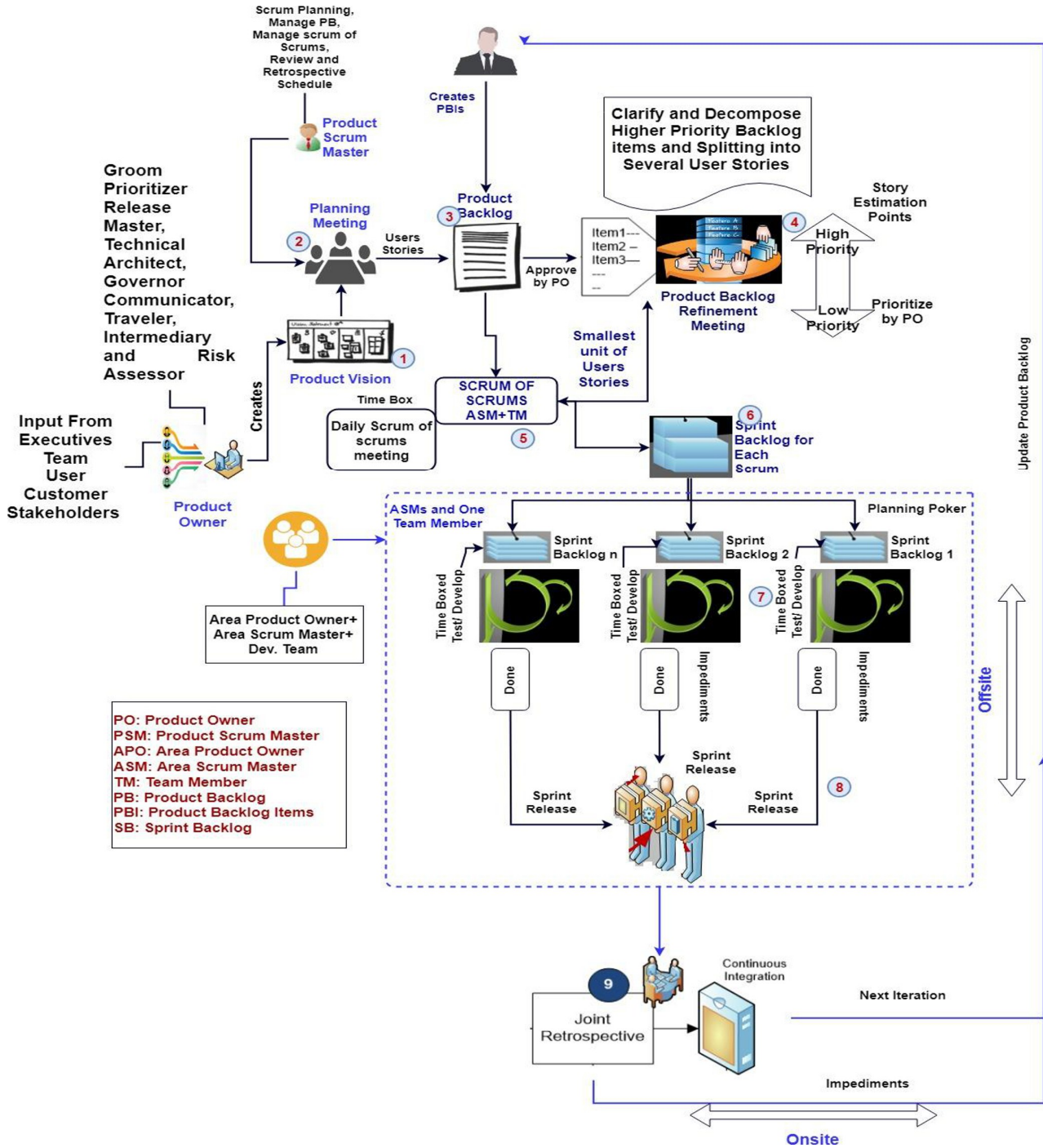


FIG. 1. SCRUM BASED AGILE FRAMEWORK

The framework is divided into planning, development and integration phase. Planning would be an on-site activity, as development will be at multiple distributed sites.

- Project initiation starts with communicating with stakeholders, gaining an understanding of needs and assessing solutions. The Product Owner creates a vision for the project. Developing the vision provides the information to create a business case and decides how the project ought to proceed. Product owner uses his technical knowledge to define the product high level roadmap, key user scenario and desired system functionality.
- Management and planning meeting is a key driver of any successful business, so it must be aligned with corporate goals, because it influences all the stakeholders and the entire organization. The agenda of this meeting is to examine contemplation for converting the vision to a release plan.

Project planning team will work together to establish the following:

- Approving and defining the project objectives in detail
- Support the development of a project release plan
- Starts discussing and sketching the early solution
- Agreeing project controls, deliverables and timelines
- Defining iteration duration, spring planning and another scrum meeting schedule

- Appropriate composition of the cross-functional teams,
- Allocating and managing resources,
- Monitoring project progress,
- Regularly update product backlog and knowledge base for the remote sites.

Product owner(s), scrum master(s) including area scrum masters conduct meetings (scrum master is the motivator, or facilitator, coach who actively participate in solving project development complexities and synching task and activities while project manager not participate in project development and involve in the planning and execution of project). Planning and project team are the essential participants of this on-site meeting. The proceedings of this meeting will be recorded and updated in the knowledge base for the remote sites.

- Product Owner is responsible for product backlog meeting in organization to develop a priority sequence called PBI (Product Backlog Items). The conditions to define these priorities can include the ROI (Return on Investment) for the development effort, knowledge of the development team's abilities, and the technical value of items for future development. The high priority PBIs deals early in the developmental cycles by the planning and project team.
- PB refinement meetings are organized to address the technical understanding of the user requirements/user stories, listed in the PB, with respect to development perspective. PB items map and split into small manageable, independent, negotiable, valuable estimates, small, and testable tasks (Fig. 2).

- The SoS (Scrum of Scrums) meeting is the most important meeting in establishing coordination and communication between the remote sites. First sprint SoS would be different as regular sprint SoS meetings, as it would be like a planning meeting for all distributed teams, lead and supervised by product scrum master and attended by all scrum masters with one technical team member (selected by all team members to represent them in planning meeting). Product scrum master shares the refined PB with remote team members and distribute high priority PBIs among the teams aligned with their resources and expertise. The output of the first SoS sprint meeting would be a team’s sprint backlog. Regular SoS meetings will share the progress and solutions of impediments if any, during the developmental cycle of each team. PB status, sprint task’s status is updated through the knowledge base, after each SoS meeting to improve project visibility and helps to share knowledge among the distributed

teams. The SoS uses when there is disburse environment to help the synchronization and time management or scheduling otherwise traditional scrum uses.

- Sprint is time box, test and development activity. Sprint schedule is defined by the project planning team during Inter-team discussion and planning phase.
- During the sprint planning meeting, APO (Area Product Owner) defines sprint goals while ASM (Area Scrum Master) shares PB, project plan along with sprint tasks and the team discuss how to accomplish defined sprint goal by establishing plan and define effort estimation. Development team consists of cross-functional and self-organized group of people, responsible for the success or failure of the sprint. During sprint, daily standup meetings are the heartbeat of sprint cycle with the purpose to describe and analyze impediments and update the backlog task’s

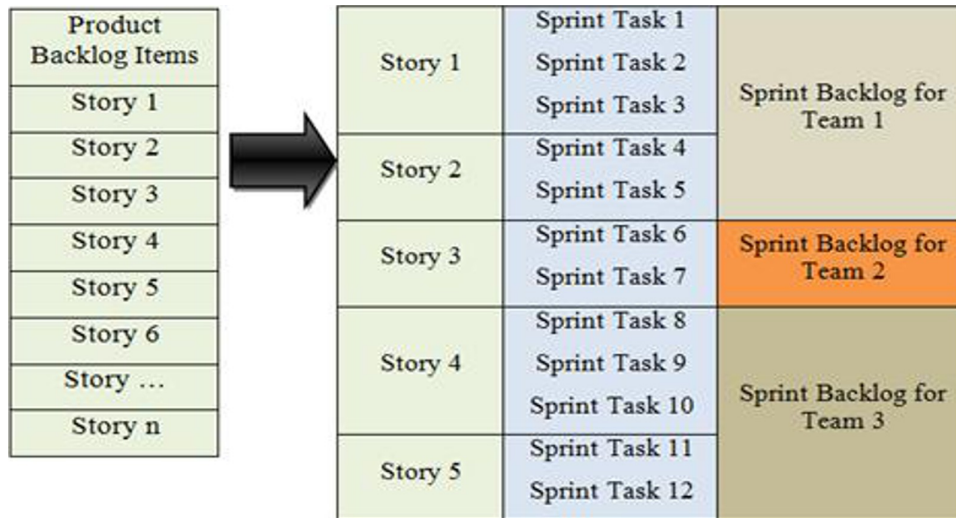


FIG. 2. USER STORY TASKS DISTRIBUTION FOR EACH SPRINT BACKLOG PROCESS

status. ASM and team attend the meeting to update tasks status and share, if any, impediments occur during task completion. After the meetings, knowledge base is updated with the task boards and burns down charts accordingly. Once development and testing phase is completed, sprint review will be performed. It is kind of informal demo, in which team presents developing features to the APO. APO is the person responsible to verify the product features as per the defined acceptance criteria. After the demo, retrospective meeting is to be held to analyze the success and failure of the sprint goal. The findings are added to the knowledge base for the future scrum teams to use. At last, verified sprint features are forwarded to the on-site project team to perform integration/testing and joint demo.

- Once the on-site project team receives completed sprints features; they perform integration testing for joint sprint review to the product owner, as per PO (Product Owner) feedback, change requests and impediments added to the PB for the next iteration. Lastly, Retrospective meeting is held to analyze the success and failure of the project. Distributed teams will also be the part of joint sprint demo and retrospective meetings through the define means, in Fig. 3, as part of the project team.

3.2 Framework Composition (Framework Roles, Activities and Artifacts)

Framework roles, practices and artifacts are listed in Fig. 3.

3.2.1 Roles

Stakeholders: Stakeholders provide information to the PO about features, and the work the Scrum Team needs to do. They also regularly review deliverables and discuss them with the PO to confirm that the work of the Scrum Team will meet their expectations and deliver business value. Finally, they need to work with the Scrum Team during the entire release.

Product Owner: The PO is central to the communication between customer and development teams. He defines and owns the PO, prioritize requirements in the product and/or sprint backlog, makes sure that PB continuously evolves and grooms with time. They are also responsible to define the product acceptance criteria [36].

Product Scrum Master: This person is responsible for setting the overall rhythm, supporting the process and ensuring that the scrum team adheres to scrum values, practices, and rules. The PSM (Product Scrum Master) coordinates with area PSM, ensures that meetings run smoothly, organizes SOS and makes sure that all the teams have what they need to be successful. If the Scrum team faces an issue that prevents progress, the PSM is responsible for ensuring its quick resolution.

Project and Planning Team: This team is a well skilled group of people, including management and planning experts, and is responsible to create a product roadmap. It organizes the scrum teams and provides technical support to PO in creating and prioritizing the product backlog. Working with product scrum master in estimating user stories, refining backlog items, developing a release plan, mapping stories into sprint backlogs and defining sprint length, team size and team velocity for each sprint. This team may include requirement engineers, designers, testers, cost, scheduling and quality assurance experts.

Area Product Owner: In a distributed scrum the developer does not have direct access to the PO. This could be managed by technical area PO. He coordinates with the PO and provides necessary support to the development team to achieve a sprint goal and at the end of sprint to verify the sprint is deliverable.

Area Scrum Master: In distributed environments, these area scrum masters set in the scrum teams to provide

local expertise and provide further details with direct supervision needed by the team. ASM, again a technical person, works interactively with the team members, help them to smoothly run the process activities and provide support in achieving the defined sprint goal.

Test and Development Team: It is a cross-functional team including programmers, testers, and designers responsible for sizing the effort required to deliver each



FIG. 3. FRAMEWORK ROLES, ACTIVITIES AND ARTIFACTS

requirement in sprint backlog and work together to achieve sprint goal. This self-organizing development, test and team as a whole are responsible for the success or failure of the project.

3.2.2 Artifacts

Vision: Vision is a high-level view of business opportunity. It describes the functional requirements and stakeholder needs. PO is creating a vision of what the product or project will be by communicating with the stakeholders, gaining an insight and understanding of needs and evaluating the solutions.

Product Backlog: PB is the master list of work for the entire project. It contains the prioritized requirements and technical details of both business value and development efforts. The initial PB is almost more than enough to start development cycle, containing a simple and precise description of desired functionality called user stories. This PB grows and refines as the development process continues.

Sprint Backlog: The sprint backlog is a list of tasks identified by the planning team for each scrum team, which needs to be completed during the sprint. During the SoS meeting, the prioritized PBI selected for each team depends on the team resources and expertise and identifies the tasks necessary to complete each user story.

Sprint's Deliverable: Part of the product is delivered by a team after the test and development phase of each sprint.

Product's Deliverable: Integrated part of the product after each sprint

Knowledge Base: The use of a repository, to guarantee the sharing of the knowledge and a common view of the

project. This provides visibility of activity and artifacts developed by the teams. All the findings related to work blockers and solution guidelines from meetings are added to the scrum knowledge base for the scrum teams to use in future. This shared base can be accessed by any team member at any time for support and guidance.

Contained information:

- (i) All releases per project
- (ii) All sprints per release
- (iii) All sprint backlog items per sprint
- (iv) All product backlog items per project
- (v) All sprint backlog items, per product backlog item
- (vi) What files are currently checked out project wise?
- (vii) The last check-in of the entire project
- (viii) The sprint burn down Charts
- (ix) The product burn-down Charts

The scrum artifacts help in PB, burn down, sprint backlog, task allocations with documentation and archival records and list of process executed means that these artifacts maintain overall progress report and manage development process with acceptance packages. Whereas, traditional software development has no acceptance packages and progress-maintained report for synchronization of project development.

3.2.3 Practices

Scrum meetings are important for each team member to remain informed about project status and changes relevant to his/her projects or parts.

In our proposed approach, product planning meeting is attended by the PO, PSM and product planning/management team. The PO describes the product features

to the team. The participants ask enough questions that they can produce a list of high-level features and recorded as PB items those further discussed in PB refinement meeting.

The backlog refinement meeting's purpose is to prioritize, estimate and split user stories. Prioritization depends on the PO description and then management and planning team split user stories into multiple tasks for the development teams and develop a scheduling, costing and effort estimation plan.

SoS is a means of scaling Scrum to large, multi-team projects. In a SoS environment, each Scrum team will typically send the scrum master and specific team member(s) to attend the SoS meeting where each ASM highlights its team progress; and PSM helps to coordinate and collaborate different teams. SoS meeting includes PSM, ASM and team member(s). Project's first SoS meeting collectively identifies and distributes PBI among different teams aligned with each team's resources and expertise. SoS provides a solution for large-scale product development organizations to connect the team's SoS to their product organization.

TSP (Thrift Savings Plan) identifies the tasks that need to be performed to achieve the desired functionality during sprint. The purpose of the Sprint planning meeting is for the scrum team to understand the project vision, discuss the sprint tasks and plan on how to complete those tasks with APO and ASM. APO defines the sprint goal and ASM facilitates the scrum team to accomplish those tasks. Each scrum team holds its own daily scrum meetings, including ASM, APO and team. A very short and time box meeting highlights progress, issues and what is to be done next.

Task boards are used to make backlogs visible to all the teams. During meetings these boards are updated

throughout the sprint. Definition of done is defined as a list of criteria that must be met before the sprint review meetings. Sprint retrospective is a dedicated period at the end of each sprint to deliberately reflect on current progress and find ways to improve the process activities. Joint review meeting after the integration testing of product features is held onsite with the product owner and is supervised by the product scrum master. Remote teams join review meetings through the defined ICT (Institute of Communication Technology) mediated strategies, listed in Fig. 4.

Retrospective meeting is usually followed with a review meeting. In this, PSM, ASM and Team member(s) are asked to identify the things that need to improve, things that need to stop and the things that need to continue during the sprint process to improve the effectiveness of process. PSM can lead this meeting and develop an initial list of ideas and then teams will commonly vote on specific items to focus on during the upcoming sprint(s). The suggestive

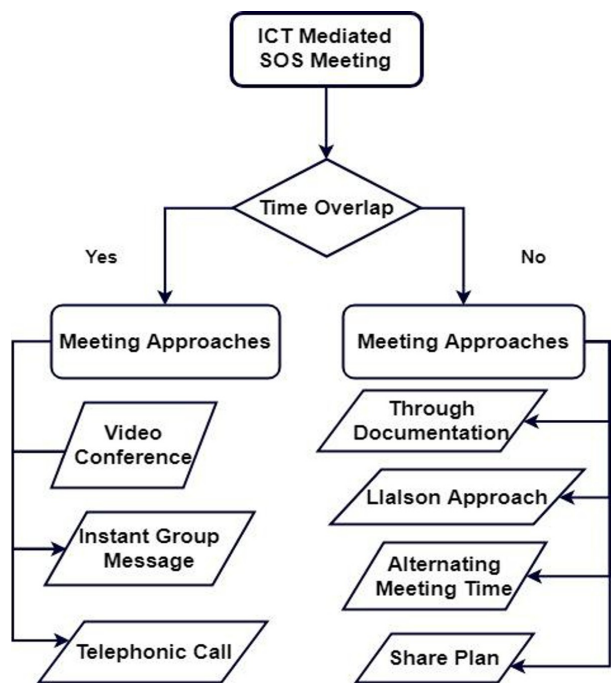


FIG. 4. ICT MEDIATED MEETING STRATEGIES

guideline will be updated on knowledge base and usually the next retrospective starts by reviewing the list of things selected for attention in the prior sprint retrospective meeting.

Scrum meetings become critical in distributed settings and need to be scheduled carefully to minimize the geographical and temporal distance.

It is difficult to organize and manage regular meeting in distributed settings. Most of the distributed teams communicate through different ICT mediated communication systems, like phone calls, instant messages, video conference, etc., but these modes of communication are suited only for the distributed locations due to no time distance as illustrate in Fig. 5. To ensure effective inter-team communication, for the non-overlapping time zones, on a regular basis, we have suggested ICT mediated meeting strategies presented in Fig. 5 to improve synchronous communication.

3.3 Evaluation

To validate proposed framework we have selected seven project management success factors in distributed settings to evaluate the proposed framework. These success factors were identified during the literature study [22,25,33-36].

These success factors are: **Project Visibility** (Able to provide better visibility to what is being done), **Project Capability** (More capable and faster in reacting to change, and flexibility), **Product Quality** (Software development quality improves), **Motivated People** (The people involved in the process are more motivated to work), **Interaction** (Inter team and inter organization, user communication and interaction at different locations with shared understanding), **Coordination** (Inter team coordination), **Productivity** (A functional piece of software produced). For evaluation of our framework we used different methods as described in **Table 1**.

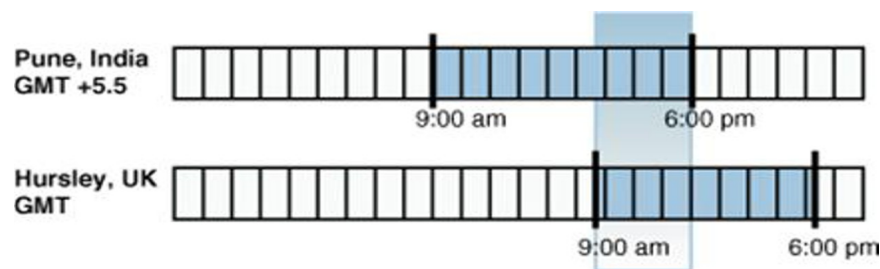


FIG. 5. DISTRIBUTED TEAMS WITH OVERLAPPING WORK HOURS

TABLE 1. EVALUATION METHODS

Evaluation	Evaluation Approach	Evaluation Focus	Data Collection	Number of Participants
I	Focus Group	Mapping of Agile Practices and DSD success factors	Semi Structure and structure interview through developed instrument	12
II	Case Study	Framework Evaluation Parameters Participant Observation	Framework Use	6
III	Industry Review	Framework Evaluation Parameters	Framework Evaluation Parameter	48

The complete detail and discussion of these methods we describe in the next section.

4. RESULTS AND DISCUSSION

We have performed theoretical as well as empirical evaluation to validate propose framework and there details are as follow.

Focus Group: A focus group was organized to validate the effectiveness of agile practices in producing high quality software products in distributed settings. Interview was attended by twelve practitioners representing four different companies. These were mid-sized to large multi-site companies from the local software industry. The companies were chosen due to their involvement in both national and international software projects.

The respondents were project managers, scrum masters, team leads, developers and site managers with relevant experience in DSD and management. Each interviewee had a master’s degree equivalent education in computer science or management information systems, with minimum five years of industry experience.

The evaluation was divided into an inductive and a deductive part. The first part was an expletive, semi

structured focus group interview that had as its objective to obtain data about the practitioners’ own experiences in DSD and agile. The second part was a structured focus group interview. At the end practitioners produced a list of agile practices they have used in practice. Participants were asked to mark each entry in the matrix according to their own experience, whether each agile practice supports the listed factors or not. After that, collected data was analyzed and summarized by calculating the mean value for each success factor. This is presented in Fig. 6.

Empirical Evaluation: For empirical evaluation of the framework, an evaluation instrument was developed and statistically tested to validate the results. Core features of the proposed framework have made the basis for questionnaire’s design. The instrument design is shown in Fig. 7.

The questions in the survey have been divided into three sections. The first section shows initial background information followed by a framework description on the second page, so the respondent gets an idea about the basic features of the proposed solution. Third section contains framework evaluation questions presented with five Likert scale.

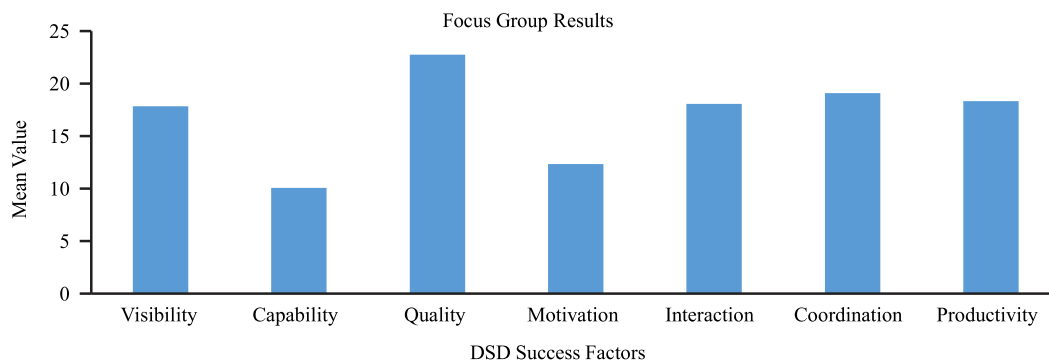


FIG. 6. FOCUS GROUP EVALUATION RESULTS

The following demographic information was collected:

- Location of the main site
- Type of organization
- Number of employees in organization
- Number of employees in their team
- Role in team
- Experience in Years
- Experience with agile practices in years

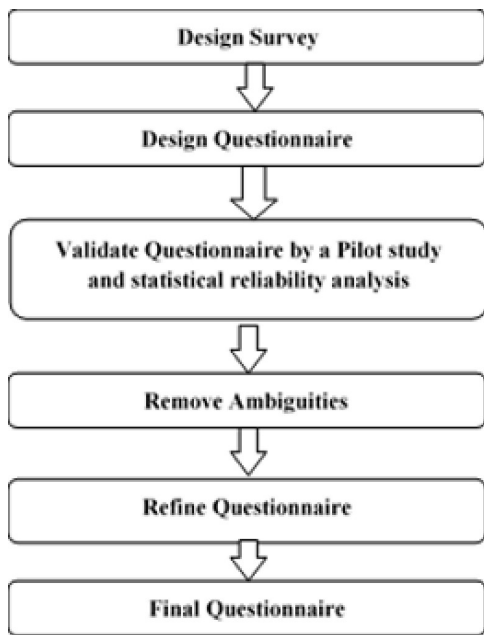


FIG. 7. INSTRUMENT DESIGN

Instrument Statistical Analysis Results: For the reliability of questioner, statistical analysis was performed. **Table 2** depicts the overall mean and SD (Standard Deviation) of the questions. Results show that our questions are unbiased and reliable as all questions divert 7.27 points from its mean value. Scales tested indicate satisfactory reliability (**Table 3**) as Cronbach’s Alpha (α) show internal consistency and dimensionality of all questions whereas 0.910 depict that highly consistent and unidimensional.

Industrial Evaluation: After finalizing the instrument, an online survey form was designed at www.surveymonkey.com. The questionnaire was then sent to credible defendants via email and proficient networks. Likewise, numerous companies involved in DSD were investigated, and then invited to contribute in the survey through the questionnaire. We received 45 responses. Before evaluating the collected data, we performed sample adequacy tests for primary component analysis to validate the collected data statistically. The results in Table 4 show that the set of variables is 0.776, which is categorized as ‘middling’ and close to 1.0 which indicates that our analysis is useful.

TABLE 2. SCALE STATISTICS

Mean	Variance	Standard Deviation	Number of Items
58.6250	52.907	7.27372	14

TABLE 3. RELIABILITY STATISTICS

Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	Number of Items
0.910	0.912	14

TABLE 4. KMO AND BARTLETT’S TEST

Kaiser-Meyer-Olkin Measure of Sampling Adequacy		0.776
Bartlett's Test of Sphericity	Approximately Chi-Square	385.769
	Difference	91.00
	Significance	0.000

Bartlett’s test of sphericity, tests the hypothesis that the correlation matrix is an identity matrix, implying that all variables are interrelated. If the significance value 0.000 of this test is less than our alpha (0.05) level, we reject the null hypothesis that the population matrix is an identity matrix. The significance value for this analysis leads us to reject the null hypothesis and conclude that there are correlations in the data set that are appropriate for component analysis is useful.

Quantitative results for each question are presented in graph in Fig. 8, showing summary of the online survey. X-axis illustrates effectiveness of our framework and y-axis illustrates number of respondent satisfaction level. 96% of the respondents agreed or strongly agreed with the evaluative statements in each question and less number of participants disagree with our framework because they were already tuned up and adopted scrum method for development. Therefore, participant easily adopted the benefits of scrum of scrums during GSD.

Case Study: For empirical validation of the proposed framework, multiple case studies are supposed to be conducted and evaluated to generalize the framework

effectiveness in different environments. However, due to the large scope of the framework and limited available resources, we have analyzed a case study for the evaluation purpose.

Case Profile: Sigma Studio is a case of worldwide trader provided design, delivery and consulting facilities in Dubai. The company, applying agile procedures to develop DSD projects, has been involved in software trade since 2000. In 2003, GSD was espoused to lead large offshore reserve pools within limited resources. Therefore, the company employed an offshore IT facilities benefactor to a main development team in the UK. This company operates in 3 countries and has 150 employees. Other project details are described in Table 5. Proposed framework has been applied on a pilot project of the logistic domain to evaluate the performance and the utility of the approach.

Data Collection: Data has been collected from six practitioners of the selected company through a structured interview. Interview questions focus on overall process evaluation as well as distinctive feature validation based on framework evaluation parameters. Results in

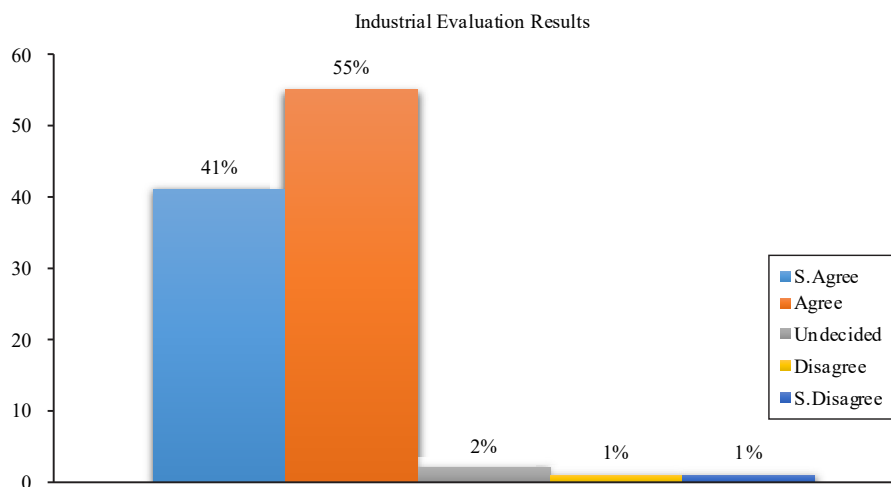


FIG. 8. ONLINE SURVEYS RESULT SUMMARY

Fig. 8 shows, how optimized practices of the scrum for distributed environments, contribute to the success factors of DSD environment.

During our experiment each customer was provided with over-all indications about anticipated system, and they had to interconnect for fulfilment of project requirements using our framework. After that time, each team presented with the fully tracked user stories that the PO had written with the customer approval on the knowledge base. The team members enable communication on a weekly basis through SoS meetings. Four sprints were carried out with the proposed framework-based process activities.

Finally, we obtained participants' personal thoughts about scrum-based product development process for distributed settings from post-experiment questionnaire.

The results were based on the stakeholders' satisfaction with project visibility, team capability, product quality, team Interaction, coordination, and productivity in the distributed context.

The company applies the proposed framework along with the proposed strategies for temporal distance to mitigate the risk associated with project management. Fig. 9 indicates the good amount of satisfaction of participants with the proposed solution. X-axis illustrates effectiveness of agile practices and y-axis illustrate number of participants satisfaction level. Four participants strongly agree and two agree over the effectiveness of agile practices to mitigating the concerned risks. It is astounding that not even a single participant disagreed with the effectiveness of agile practices.

TABLE 5. PROJECT DETAILS

Data Source	Company: Project	Experience in Years
Document Retrospection	5 (3 Week) Iteration	5
Interview	2 Project Managers	3
	1-Developers	2
	2-Scrum Master	2
	1-Product Owner	1
Total:		6

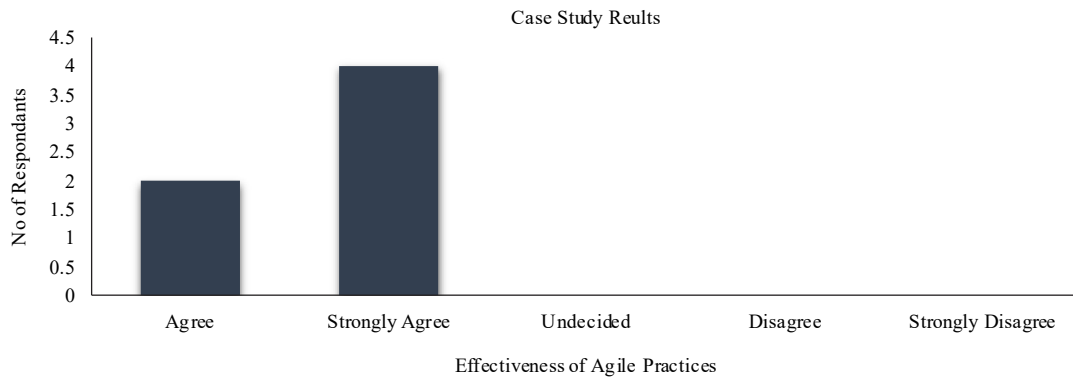


FIG. 9. CASE STUDY RESULTS

The overall estimation revealed that our framework is useful in enabling listed success factors related to the geographical distribution of projects. Therefore, the overall evaluation results that our framework outperform from traditional DSD methods as well from XP and lean development method.

DSD has converted a common business reality. Additionally, recent business dynamics entail to develop systems rapidly. The results of these foremost trends, software companies have been motivated to combine agile and DSD to obtain aids of both.

Literature analysis highlighted the positive aspects of agile practices of continuous communication between stakeholder and multi-sites teams. Iterative development makes it easier for applicants with short-term aims. Sprint reviews also are very effective in improving communiqué as it aids stakeholders to share data approximately tasks and interdependency of requirements. Due to amplified communiqué and teamwork there is an enhancement in the software quality and team motivation [29].

The proposed framework is included on-site as well off-site activities. The reason of dividing the project activities in on-site and off-site is to reduce the planning issues that need direct communication between stakeholders. In fact, planning process gives a solid understanding of the process and help to accomplish something that is required to develop. Due to mentioned distances in DSD, it needs to have a detailed project planning meeting with essential stakeholders on-site. This could lead to the successful execution of the process that includes agile based requirements engineering practices, budget and cost estimation, resource allocation and required scheduling of team activities.

At the PB refinement meeting, user stories were broken down into small manageable tasks, with minimum task interdependencies, so when these tasks were distributed

among teams for the development, integration issues would be minimized.

Off-site phase includes the multiple test and development team with defined activities, which follows the on-site scrum process to deliver sprint goals. ICT mediated SoS meeting could be helpful in establishing a communication between on-site management and off-site development teams. Progress reports and updates were shared with each other to improve project visibility and inter-team coordination.

On completion of each sprint multiple team's deliverable forwarded to the on-site management teams to perform integration testing. After this, ICT mediated, joint review and retrospective meetings were arranged to refine the sprint process.

Formal communication and collaboration between managers and planning team is critical during off-site development sprints. On-site planning meetings resolve the interdependency among the tasks to minimize the need of formal communication between the distributed teams. Other than these planning meetings, scrum of scrums meetings resolves the impediments faced by distributed teams and provide facility for distributed teams to share their knowledge with each other.

Sprint backlogs developed during the first sprint SoS meeting, according to the team resources and expertise that help to accommodate the resource allocation and utilize team potential. The reason for having an extremely set number of individuals in SoS meetings is to make it more compelling and effective for different groups and their members. Area scrum master imparts their sprint objectives and technical team member to share their team progress by answering simple questions, including the impediments faced during development phase, and how they had to deal with said issues. The comparative analysis of our framework with existing methods illustrated in the Table 6.

TABLE 6. COMPARISON WITH EXISTING TOOLS

	Proposed Framework	Domia et. al. [22]	Shah et. al. [27]	Behm et. al. [28]	Daneva et. al. [30]
DSD and GSD	✓	✓	✓	✓	✓
Scrum of Scrums	✓	✗	✗	✗	✗
Complex	✗	✓	✗	✓	✓
Project Visibility	✓	✗	✗	✗	✗
Project Capability	✓	✓	✗	✗	✗
Reacting to Change	✓	✓	✗	✗	✓
Optimizing Interaction	✓	P	✓	✗	✗
Improving team Coordination	✓	✗	✓	P	✗
Proper Task Allocations	✓	✗	P	✗	✓
Scrum Team Interaction	✓	✗	✗	P	✗
Completely Compiled= "✓" Partially Compiled = "P" Not Compiled = "✗"					

5. CONCLUSION

This paper presents a framework that integrates and extends the theoretical concepts of DSD projects, their success and agile practices, showing how they mutually support each other. As DSD becomes the benchmark for cost-effective development, despite all the benefits, it faces many challenges that increase the project failure rate. On the other hand, agile becomes a de facto in developing quality products in collocated environments. Here we have designed a framework for the distributed team to mitigate most of the faced challenges in distributed environments. The basic idea behind this research is to integrate the effectiveness of the agile practices in the distributed environment.

Defined roles in framework help to increase inter-team communication activities and overall team coordination. Artifact sharing, and regular updates would help to improve project, team productivity and process quality. After evaluation of the framework discussed within this section, our assessment is that distributed agile does work. Because of this, stretching out Agile to dispersed

groups is possible in a manner that remains true to the values and principles of Agile.

6. FUTURE WORK

The motivation behind this investigation was to develop better understanding and provide guidance that agile performs in the distributed situations more effectively. It would be helpful in the ongoing development of a complete automated management process for DSD.

ACKNOWLEDGEMENT

The authors are grateful for the support of F3 Technology Rawalpindi and PSEB Islamabad, for vetting of data and survey to evaluate our SBAFe.

REFERENCES

- [1] Hossain, E., Babar, M.A., and Paik, H.Y., "Using Scrum in Global Software Development: A Systematic Literature Review", Proceedings of IEEE 4th International Conference on Global Software Engineering, Volume 8, No. 9, pp. 175-184, July, 2009.

- [2] Herbsleb, J.D., and Moitra, D., "Global Software Development", IEEE Software, Volume 18, No. 2, pp.16-20, 2001.
- [3] Lanubile, F., Damian, D., and Oppenheimer, H.L., "Global Software Development: Technical, Organizational, and Social Challenges", ACM SIGSOFT Software Engineering Notes, Volume 28, No. 6, pp. 2, 2003.
- [4] Bannerman, P.L., Hossain, E., and Jeffery, R., "Scrum Practice Mitigation of Global Software Development Coordination Challenges: A Distinctive Advantage?", IEEE 45th Hawaii International Conference on System Science, Volume 7, No. 12, pp. 5309-5318, January, 2012.
- [5] Verner, J.M., and Abdullah, L.M., "Exploratory Case Study Research: Outsourced Project Failure", Information and Software Technology, Volume 54, No. 8, pp. 866-886, 2012.
- [6] Sidky, A., and Smith, G., "Becoming Agile in an imperfect World", Manning Publications, Greenwich, CT, USA, 2009.
- [7] Li, J., Moe, N.B., and Dyba, T., "Transition from a Plan-Driven Process to Scrum: A Longitudinal Case Study on Software Quality", Proceedings of ACM-IEEE International Symposium on Empirical Software Engineering and Measurement, Volume 10, No. 9, pp. 13, September, 2010.
- [8] Shrivastava, S.V., "Distributed Agile Software Development: A Review", arXiv Preprint arXiv:1006.1955, 2010.
- [9] Shrivastava, S.V., "A Framework for Risk Management in Globally Distributed Agile Software Development (Agile GSD)", Journal of Computer Science and Engineering, Volume 1, Issue 1, pp. 175-184, May 2010.
- [10] Reifer, D.J., Maurer, F., and Erdogmus, H., "Scaling Agile Methods", IEEE Software, Volume 20, No. 4, pp. 12-14, 2003
- [11] Shalinka, L., and Lai, R., "A Systematic Review of Requirements Change Management", Information and Software Technology, Volume 93, pp. 163-185, 2017.
- [12] Ali, S., Iqbal, N., and Hafeez, Y., "Towards Requirement Change Management for Global Software Development Using Case Base Reasoning", Mehran University Research Journal of Engineering and Technology, Volume 37, No. 3, Jamshoro, Pakistan, July, 2018.
- [13] Hashmi, A.S., Hafeez, Y., Jamal, M., Ali, S., and Naila, I., "Role of Situational Agile Distributed Model to Support Modern Software Development Teams", Submitted in Journal of Mehran University of Engineering & Technology, Volume 37, No. 1, Jamshoro, Pakistan, January, 2018.
- [14] Niazi, M., Mahmood, S., Alshayeb, M., Rehan, R.M., Faisal, K., and Cerpa, N., "Challenges of Project Management in Global Software Development: Initial Results", Science and Information Conference, pp. 202-206, October, 2013.
- [15] Saeed, K., Hafeez, Y., Ali, S., Iqbal, N., and Shahid, U., "Enabling the Usability Heuristics of Agile Base Systems to Improve Quality of Local Software Industry", Mehran University Research Journal of Engineering and Technology, Volume 38, No. 2, Jamshoro, Pakistan, April, 2018.
- [16] Gul, Z., Hafeez, Y., Hashmi, A.S., Ali, S., and Iqbal, N., "Introducing Team Coordination Framework Support Globally Distributed Teams for Pakistani Software Industry", Mehran University Research Journal of Engineering and Technology, Volume 37, No. 4, Jamshoro, Pakistan, October, 2018.
- [17] Hossain, M.E., "Scrum Practice Mitigation of Coordination Challenges in Global Software Development Projects: An Empirical Study", Ph.D. Thesis, The University of New South Wales Sydney 2052, Australia, 2011.
- [18] Bjarnason, E., Wnuk, K., and Regnell, B., "Are You Biting Off More than You Can Chew? A Case Study on Causes and Effects of Over Scoping in Large-Scale Software Engineering", Information and Software Technology, Volume 54, No. 10, pp. 1107-1124, 2012.
- [19] Kazmi, S.H.M., Hafeez, Y., and Ali, S., "Software Outsourcing Model for Risk Mitigation", IEEE International Conference on Computing, Mathematics and Engineering Technologies, pp. 1-11, 2018.

- [20] Ilyas, M., and Khan, S.U., "Software Integration Challenges in Global Software Development Environment: A Systematic Literature Review Protocol", *IOSR Journal of Computer Engineering*, Volume 1, pp. 29-38, 2012.
- [21] Sillitti, A., and Succi, G., "Requirements Engineering for Agile Methods" *Engineering and Managing Software Requirements*, pp. 309-326, Springer Berlin Heidelberg, 2005.
- [22] Domia, L., Moawad, R., and Kadry, M., "Supporting Tools for Requirements Change Management in Distributed Agile Development", *Future Computing and Informatics Journal*, Volume 2, No. 1, pp. 1-9, 2017.
- [23] Dorairaj, S., Noble, J., and Malik, P., "Knowledge Management in Distributed Agile Software Development", *IEEE Agile Conference*, pp. 64-73, 2012.
- [24] Dorairaj, S., Noble, J., and Malik, P., "Understanding Team Dynamics in Distributed Agile Software Development", *Agile Processes in Software Engineering and Extreme Programming*, pp. 47-61, Springer Berlin Heidelberg, 2012.
- [25] Ramesh, B., Cao, L., Mohan, K., and Xu, P., "Can Distributed Software Development be Agile?", *Communications of the ACM*, Volume 49, No. 10, pp. 41-46, 2006.
- [26] Paasivaara, M., Durasiewicz, S., and Lassenius, C., "Using Scrum in Distributed Agile Development: A Multiple Case Study", *IEEE 4th Conference on Global Software Engineering*, pp. 195-204, 2009.
- [27] Shah, Y.H., Raza, M., and Haq, S., "Communication Issues in GSD", *International Journal of Advanced Science and Technology*, Volume 40, No. 40, pp. 69-76, March, 2012.
- [28] Behm, B., Dikert, K., and Paasivaara, M., "Communication Practices in a Distributed Scrum Project", *arXivPrepr. arXiv*, 2013.
- [29] Brown, A.W., Ambler, S., and Royce, W., "Agility at Scale: Economic Governance, Measured Improvement, and Disciplined Delivery", *Proceedings IEEE International Conference on Software Engineering*, pp. 873-881, May 2013.
- [30] Daneva, M., Van Der Veen, E., Amrit, C., Ghaisas, S., Sikkell, K., Kumar, R., Ajmeri, N., Ramteerthkar, U., and Wieringa, R., "Agile Requirements Prioritization in Large-Scale Outsourced System Projects: An Empirical Study", *Journal of Systems and Software*, Volume 86, No. 5, pp. 1333-1353, 2013.
- [31] Sriram, R., and Mathew, S.K., "Global Software Development Using Agile Methodologies: A Review of Literature", *IEEE International Conference on Management of Innovation and Technology*, pp. 389-393, June, 2012.
- [32] Santos, M.D.A., Bermejo, P.H.D.S., de Oliveira, M.S., Tonelli, A.O., and Seidel, E.J., "Improving the Management of Cost and Scope in Software Projects Using Agile Practices", *arXiv preprint arXiv:1303.1971*, 2013.
- [33] Melo, C.D.O., Cruzes, D.S., Kon, F., and Conradi, R., "Interpretative Case Studies on Agile Team Productivity and Management", *Information and Software Technology*, Volume 55, No. 2, pp. 412-427, 2013.
- [34] Rothman, J., and Hastie, S., "Lessons Learned from Leading Workshops About Geographically Distributed Agile Teams", *IEEE Software*, Volume 30, No. 2, pp. 7-10, 2013.
- [35] Lagerberg, L., Skude, T., Emanuelsson, P., Sandahl, K., and Stahl, D., "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*, pp. 348-356, 2013.
- [36] Bass, J.M., "Agile Method Tailoring in Distributed Enterprises: Product Owner Teams", *IEEE 8th International Conference on Global Software Engineering*, pp. 154-163, August, 2013.
- [37] Schneider, S., Torkar, R., and Gorschek, T., "Solutions in Global Software Engineering: A Systematic Literature Review", *International Journal of Information Management*, Volume 33, No. 1, pp. 119-132, 2013.