Journal of Theoretical and Applied Information Technology
15th September 2021. Vol.99. No 17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ENHANCING DISTRIBUTED AGILE TEAMS' ADOPTION DURING THE COVID-19 PANDEMIC USING TAGICK ACTIVITIES

[1*]MONA RASMY, [1]OSAMA ABU-ELNASR AND [1]SAMIR ELMOUGY

[1] Department of Computer Science, Faculty of Computers and Information, Mansoura University, 35516, Egypt

E-mail: [*]monamahmoudrasmy@gmail.com, mr_abuelnasr@mans.edu.eg and mougy@mans.edu.eg

ABSTRACT

Due to the viral outbreak of the coronavirus (COVID-19) pandemic, organizations have adhered to social distancing and/or lockdown measurements. Project teams have shifted from direct communication in the workplace to remote working. Obviously, such sudden changes have led the organizations to face many challenges. A number of these challenges are related to the communication and cooperation among the team members and also to fulfilling the code documentation process that helps the team later in the maintenance phase. In this paper, the researchers propose a modification to the Scrum methodology called the Distributed Scrum (Di-Scrum). It is designed based on suggesting activities called the TAGICK activities (Timekeeping; Aggregation; Groupthink; Interconnectedness; Continuous documentation; and Knowledge transfer). Moreover, each activity supports the agile principles and rules. Additionally, these activities help the team members improve their performance. They also enable them to overcome all the obstacles mentioned above which they face while working remotely. We have used a questionnaire to evaluate the suggested activities. The questionnaire is filled in by 40 different employees in four software development companies applying the Di-Scrum model. The results of the evaluation indicate the effectiveness of the TAGICK activities with remote teams. They have led to enhancement of group communication, cooperation and ability of continuous documentation.

Keywords: *Distributed agile, Distributed Teams, Work Remotely, COVID-19, Software Development, Communications Theory, Scrum Methodology*

1. INTRODUCTION

Software Development (SD) relies on a set of methodologies, practices and tools. In order to improve business performance and project management, SD goes through specific phases called the Software Development Life Cycle (SDLC). The phases of SDLC include defining the requirements and designing the solutions, then, starting implementation, moving to verification and, finally, providing maintenance [1]. Moreover, SD relies on different methodologies, such as the traditional methodologies and the agile methodology for designing programs and taking into account the type of requirements, their clarity and the extent of their change [2], [3]. The agile methodology achieves customer satisfaction and delivers product quality that meets the customers' requirements [4], [5]. It also includes many frameworks, such as Scrum [6], Extreme Programming (XP) [7], Lean [8] and Kanban [9]. These entire frameworks are methods for applying the principles and values of the agile methodology [10]. The agile methodology is more suitable for a dynamic and co-located work environment.

Nowadays, the world is facing a number of challenges due to the outbreak of the Coronavirus (COVID-19). Actually, this epidemic has led to social separation. Consequently, organizations have switched to remote work. However, the agile methodology has a limited support for the distributed development environments [11]. This may result in poor communication, lack of documentation, development team loss of the self-organization feature, irregular working hours and work environment and making quick decisions without documenting it.

Therefore, the present paper proposes the Distributed Scrum (Di-Scrum) model in an attempt to address the above-mentioned problems. The paper is organized as follows: Section 2 focuses on the Scrum model and communication theories,

since they are the basis of the proposed model. Section 3 presents a brief review of the various software development life cycle methodologies and their interactions with the distributed teams. In Section 4, the proposed Di-Scrum model and TAGICK activities are described in details. Moreover, section 5 includes the research design and evaluation method of the proposed model. The results are presented and discussed in Section 6. Finally, Section 7 demonstrates the conclusions.

2. LITERATURE REVIEW

2.1 Scrum

The Scrum methodology is one of the most popular agile techniques. It is defined as an iterative, incremental and rapid process of software project management [12]. This methodology makes it easier for teams to respond quickly and effectively to change through providing techniques and practices for project management [13]. Moreover, the Scrum framework consists of three components: roles, meetings and artifacts [14].

The Scrum team mainly plays three roles: The Product Owner, the Development Team and the Team Master. At the outset of the process, the Product Owner lists the product requirements and the product backlog according to their priorities, in terms of maximizing business value and return on investment [15]. Then, the Product Owner meets the team members to split user stories into sprints, with a two to four weeks repeat. After that, sprint planning begins with the Product Owner describing the scope and tasks. At the same time, the team provides a list of the work that can be done during a single sprint, decides on the complexity of the tasks "story points" and estimates the time each task needs. Every day, during 15 minutes, the team conducts the daily Scrum meeting. The Team Master asks the team members three questions related to their work done the previous day, their next activity planned for today and the obstacles to achieving the tasks.

In addition, the agile team consists of five to nine individuals with different skills. The Team Master is responsible for solving problems, improving teamwork and ensuring working according to the agile principles and values [14]. At the end of the sprint phase, a sprint review meeting is held to present the completed work and take feedback. Finally, a retrospective meeting is held about the lessons learned by the team [15]. Actually, the agile teams need to interact and communicate efficiently in order to finish tasks on time and avoid failure that could jeopardize the investments made [12], [13].

2.2 Communication theories in project management

The importance of communication in project management is hidden in achieving organizational goals and delivering a high-quality product. Therefore, cooperation must take place among all the members of the project to define goals and share responsibilities [16]. Moreover, communication is considered one of the most valuable aspects of the agile methodologies [17]. Any communication defect is a factor that may lead to the failure of the project [17], [18]. Table 1 presents the communication theories in project management, their principles and their strategies [19].

Table 1: Communication theories in project management

Communication Theory	Theoretical Principles	Theoretical Strategy
Diffusion Theory(DT)	Humans are creatures of habits. It is difficult for individuals to change their habits or choices to think in a moment [19]. Some individuals can easily adapt to changes while others take time to adjust [18].	This theory is applied during the planning stage. Sudden changes in plans may confuse some team members, resulting that they may not work properly as usual. Hence, plans should be specific and comprehensive to avoid confusion among the team members [20].
Groupthink Theory(GT)	The group adopts a single mind to achieve maximum results and avoid distracting opinion [21]. The cohesion among team members is necessary so that collective thinking is not difficult to achieve [20].	To apply this theory, the project manager needs to reach a common ground with the team members to make sure that they all target the same goal. Moreover, most importantly, the team members must feel they are part of the group and achieving the project goals is more important than their personal differences [20].
Social Information Processing Theory(SIPT)	This theory emphasizes that relationships via the internet can be as powerful as face-to-face relationships [21]. Interpersonal communication forms strong bonds through	To apply this theory, project managers need to pay more attention to their words and to conveying messages properly despite the lack of evidence of social context [20].

	the very few clues that individuals can get from online conversations [21].	
Communication Accommodation Theory(CAT)	This theory argues that an individual adjusts to the person with whom he is communicating [22]. When team members interact with each other, the individual attempts to get involved with others through changing speech patterns, body language and gestures [22].	To apply this theory, it becomes a duty for project managers to communicate well with their team members. However, they must try to avoid excessive housing that can occur in four ways: exaggeration, sensuality, dependency and adjustment among groups [20].

3. RELATED WORK

Over the past years, many software engineering researchers were interested in publishing researches on different ways to develop agile methodologies. In [23], Sarhan S. et al. proposed a group of activities for early detection of risks, improvement of team performance and better handling of changing requirements. In [24], El Allaoui et al. proposed a method for generating test cases from the UML sequence diagrams integrated with the Scrum agile process. This was done through introducing two new AndroMDA cartridges. The first cartridge is for M2M transformation. It takes UML 2 sequence diagrams as input and produces U2TP sequence diagrams. The second cartridge is for M2T transformation. It takes U2TP sequence diagrams as input and generates test cases. These suggestions relied on teams working in co-located environment. Furthermore, these studies were not devoted to investigate distributed teams.

On the other hand, many researchers turned to studying how agile methodologies deal with distributed teams, clarifying their strengths and weaknesses. In [25], S. Morrison-Smith and J. Ruiz studied the collaboration challenges that virtual teams face and the recent strategies used to handle them. They found that the challenges facing distributed teams fall into five categories: geographical distance, time distance, perceived distance, dispersed team formation and diversity of workers. They proposed a strategy to better support collaborative tasks in virtual teams through applying four design implications: helping to establish common ground and business standards; facilitating communication; introducing mechanisms to achieve work transparency; and providing lightweight and familiar technology design.

In [26], A. Winter examined a number of challenges caused by shifting to virtual working. They included managers and employees are clueless about how projects can be continued in full-time virtual teams and the related new advantages or problems that hinder/improve project execution. As a result, the researcher made suggestions to meet these challenges. Furthermore, in [27], Yermolaieva and Sofiia surveyed 53 participants to assess the most common communication challenges that agile teams face during work. The researchers also developed recommendations to overcome communication problems based on communication theories.

In [28], M. E. Badiale et al. studied thoroughly the positive and negative effects of communication factors such as technology, culture and trust during the COVID-19 pandemic. They found that combining simultaneous and asynchronous communication is important and the methods of communication among the members change according to the cultural backgrounds. Their results also showed that communication becomes effective when there is trust among the team members.

In, [29] Marek et al. conducted a survey to verify the status of the agile software development teams during the Corona pandemic period and its effect on their work. They found that the Covid-19 pandemic had little impact on the Product Backlog planning and the visibility they define done and the frequency of the release. Despite this, they found that a few teams changed their work organization to face the changes related to the Product Backlog and Vision.

4. DISTRUBUTED SCRUM (DI-SCRUM) MODEL BASED ON TAGICK ACTIVITIES

Due to the current situation and the spread of the Coronavirus, organizations have turned to remote work and distributed teams have become the norm. Scrum activities keep the entire team focused on the sprint targets. However, it becomes difficult to implement them in distributed teams. This section explains how the Di-Scrum methodology makes activities more effective when working remotely, enhances communication and collaboration among team members and increases the level of documentation, as shown in Figure 1. Moreover, in this paper, the "TAGICK" activities are a set of

suggested activities aim at fitting the agile methodology to the distributed teams and removing any sense of distribution. In addition, these activities can add dynamic value to the distributed agile teams and enhance the quality of their advanced programs.

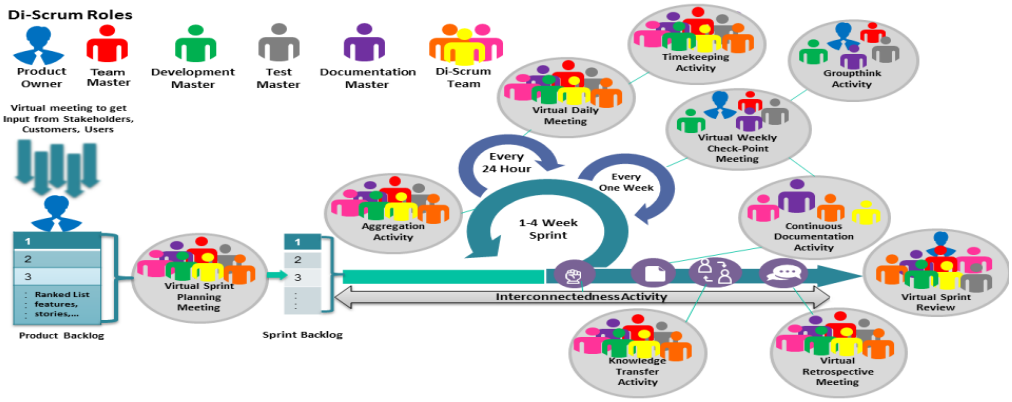


Figure 1: Di-Scrum framework based on the TAGICK activities

4.1 Di-Scrum Collaboration Tools

The Di-Scrum collaboration tools are essential to facilitating remote working. The Di-Scrum team uses rapid planning tools to collect stories/requirements, report and manage issues and track progress and quality. Additionally, the Di-Scrum team adopts the virtual multiple Scrum boards and Jira automation rules. There are two boards. First, the high-level board is for managers and team masters. It displays main user stories, tasks, and bug reports. It also provides a quick overview of how the current sprint is running. Second, the low-level board is useful for development and quality assurance teams. It presents the detailed sub-tasks.

Additionally, the Di-Scrum team applies Slack for unplanned asynchronous internal communication. Finally, it uses Zoom conferences for planned communication in formal meetings simultaneously.

4.2 Di-Scrum Roles

The Di-Scrum team is a Scrum team that works fully or partially remotely. For the success of the Di-Scrum team, new approaches to accrediting Scrum must be implemented. Besides, in order to achieve easy communication and cooperation among the team members, the traditional roles of the Scrum team need to be adjusted to meet the challenges of remote working as shown in Table 2.

Table 2: Di-Scrum roles

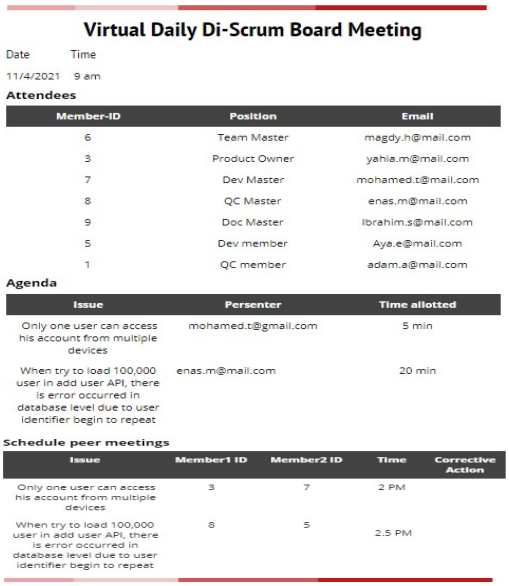
Di-Scrum Role	Responsibility
Product Owner	Communicates with stakeholders and gathers requirements remotely to build product backlog. He must be committed and involved with the team in virtual meetings where he directs the team to present a successful project to the stakeholders.
Team Master	Organizes meetings and activities that the distributed team carries out. He handles obstacles. He also facilitates communication among the team members and continuous communication with the master of each team to ensure that the team follows the steps of Di-Scrum correctly.
Development Master	Represents the development team in the weekly meeting to achieve the theory of diffusion. Moreover, his responsibility includes managing the Dev-team, preparing the agenda of the meetings for his team, documenting meeting decisions and following up on the implementation of these decisions.
Test Master	Represents the test team in the weekly meeting to achieve the theory of diffusion. His responsibility also includes leading the test channel for his team and following up on the workflow.
Documentation Master	Documents the project code and divides it into groups of API. Each API is divided into a set of methods. Moreover, every method has input, output and a detailed explanation of the code. He represents the documentation team in the weekly meeting to achieve the theory of diffusion.
Development Team	Includes developers, designers, writers, testers, etc. The responsibility of the team is to follow the plan and be committed to communication, cooperation and self-regulation.

4.3 Di-Scrum Activities (TAGICK)

4.3.1 "T" Timekeeping activity

The Timekeeping Activity is an iterative activity. It organizes and prepares the team to focus during the virtual daily meeting by the end of the working day. It also assists the team with regard to simultaneous group and individual communication, sharing of information and goals and passing of ideas amongst the team members. This activity is based on two communication theories: the (DT) and (CAT) theories.

Moreover, each team member displays what has been accomplished and the issues faced. Then, the Team Master organizes the members and links them to the digital Scrum board to facilitate discussion of tasks and set a clear agenda, as shown in Figure 2. Therefore, they can discuss them every day in the meeting. This activity is aimed at a team of developers, testers, analysts and designers. After this meeting, individual "peer meetings" takes place among the team members to participate in resolving any technical issues. This practice fosters collaboration on technical issues or approaches. As a result, it leads to increased group bonding. This activity matches the agile value: "people and interactions over processes and tools."



Virtual Daily Di-Scrum Board Meeting

Date Time

11/4/2021 9 am

Attendees

Member-ID	Position	Email
6	Team Master	magdy.h@mail.com
3	Product Owner	yahia.m@mail.com
7	Dev Master	mohamed.t@mail.com
8	QC Master	enas.m@mail.com
9	Doc Master	Ibrahim.s@mail.com
5	Dev member	Aya.e@mail.com
1	QC member	adam.a@mail.com

Agenda

Issue	Persenter	Time allotted
Only one user can access his account from multiple devices	mohamed.t@gmail.com	5 min
When try to load 100,000 user in add user API, there is error occurred in database level due to user identifier begin to repeat	enas.m@mail.com	20 min

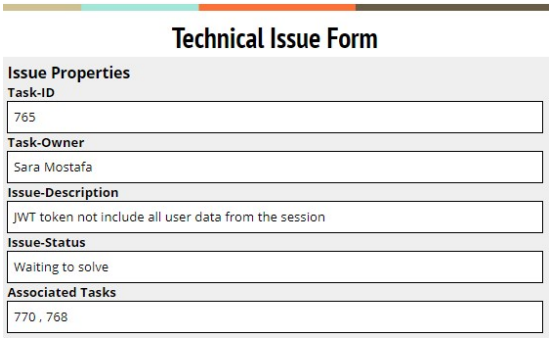
Schedule peer meetings

Issue	Member1 ID	Member2 ID	Time	Corrective Action
Only one user can access his account from multiple devices	3	7	2 PM	
When try to load 100,000 user in add user API, there is error occurred in database level due to user identifier begin to repeat	8	5	2.5 PM	

Figure 2: Virtual daily Di-Scrum meeting agenda

4.3.2 "A" Aggregation activity

In this activity, during the sprint, all the technical issues that the development team and the testing team face are grouped into one form shared among the team members. Besides, any member can add a solution to the problem. He can also search for a similar problem and use its solution and communicate with the one who solved it, as clarified in Figure 3. Thus, this makes it easier for the issue owner to experiment with different solutions and choose the most appropriate one. Then, he changes the issue status from "waiting for a solution" to "solved". This file facilitates dealing with the issues without interrupting another member to solve them at inappropriate time. Moreover, this sheet is shared via Slack. In addition, when any team member updates or edits this sheet, the rest of the team is notified. This sheet is a repository of technical issues or approaches; and solutions can be referred to at any time. This activity matches the agile principle**: "**Attention to technical detail and design enhances agility".



Technical Issue Form

Issue Properties

Task-ID

765

Task-Owner

Sara Mostafa

Issue-Description

JWT token not include all user data from the session

Issue-Status

Waiting to solve

Associated Tasks

770 , 768

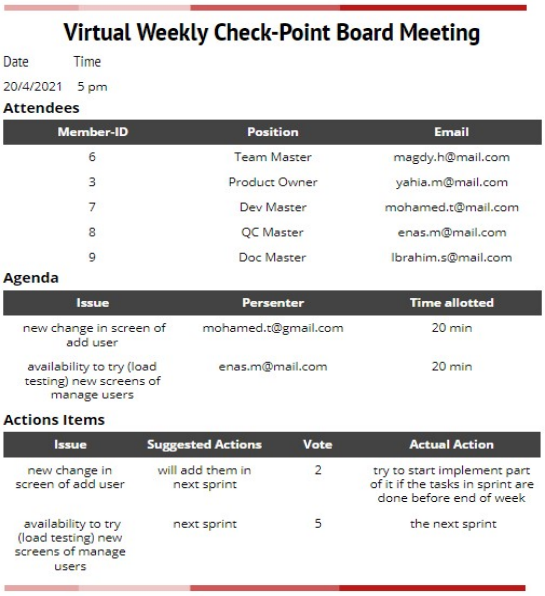
Figure 3: Technical issue form

4.3.3 "G" Groupthink activity

In this activity, according to the groupthink theory, the Team Master organizes the virtual checkpoint meetings weekly. The attendances, who are allowed to participate in the weekly virtual meetings, are: The Product Owner, the Developer Master, the Tester Master and the Documentation Master.

In this meeting, the attendees produce a document, through filling the sheet shown in Figure 4, based on discussing the following: the tasks accomplished during the week; the plan reviewed; whether all the stories will be completed within the specified time for the sprint; whether there are changes in the stories or in requirements; the need to delete stories or migrate them to a new race; and whether new stories shall be added.

In addition, the diffusion theory and taking the right decision are adopted. This activity matches the agile principles: **"**collaboration between the business stakeholders and developers throughout the project**"** and **"**accommodate changing requirements throughout the development process**".**



Virtual Weekly Check-Point Board Meeting

Date Time
20/4/2021 5 pm

Attendees

Member-ID	Position	Email
6	Team Master	magdy.h@mail.com
3	Product Owner	yahia.m@mail.com
7	Dev Master	mohamed.t@mail.com
8	QC Master	enas.m@mail.com
9	Doc Master	Ibrahim.s@mail.com

Agenda

Issue	Persenter	Time allotted
new change in screen of add user	mohamed.t@gmail.com	20 min
availability to try (load testing) new screens of manage users	enas.m@mail.com	20 min

Actions Items

Issue	Suggested Actions	Vote	Actual Action
new change in screen of add user	will add them in next sprint	2	try to start implement part of it if the tasks in sprint are done before end of week
availability to try (load testing) new screens of manage users	next sprint	5	the next sprint

Figure 4: Virtual Weekly Check-Point Meeting Agenda

4.3.4 "I" Interconnectedness activity

This activity aims to enhance internal communication among the members of the Di-Scrum team. It deals with the problems of communication, cooperation and difficulty of applying pair programming among team members when working remotely. Furthermore, slack has been relied upon as the main focus of internal communication. It has been also linked with Jira to automate managing and organizing work in the project. This activity applies (CAT) and (SIPT) theories. It creates new slack channels instead of internal emails during the development, test and documentation processes, such as follows:

Sprint – It is a public channel for all member of the project. They use it to share retrospective sprint reports and urgent communications about the current sprint that needs immediate attention from managers in order to be discussed at the weekly meeting. The Team Master leads this channel.

#Masters– It is a special channel for the Team Masters and Product Owners.

#Dev– It is a special channel for the developers. It is for providing communication and collaboration among the developers, reporting issues, seeking advice from others, and sharing code snippets. Through this channel, the team shares a problem memory sheet, suggests solutions and communicates asynchronously. The Development Master controls this channel.

#QC– It is a special channel for the Quality Control team to communicate and collaborate with regard to product quality and to notify about the completion of the user's story. The Test Master is responsible for this channel.

#Doc– It is a special channel for the documentation team. Communication takes place through it to ensure documentation quality and solve the problems facing the documentation teams. The Documentation Master is responsible for this channel.

This "I" Activity matches the agile principle: "Self-organizing teams encourage great architectures, requirements, and designs".

4.3.5 "C" Continuous documentation activity

This activity has two approaches. In the first approach, the technical documentation team documents the code and divides it into groups of API. Each API is divided into a set of methods. Furthermore, each method has input, output and a detailed explanation of the code, using the pseudo code, to make reading this documentation easy for any new member or for the rest of the team members. This documentation is intended for developers.

In the second approach, the code reviewer or the developers document the code. In addition, choosing the appropriate approach for documenting the code depends on the size of the project and whether there are additional resources to be exploited.

Furthermore, the documentation is done after the code comes out from the QC team to ensure that there are no Bugs. Continuous documentation leads to the documentation of the sprint code at the end of the sprint without delaying or disrupting the developers. Teams working remotely for long periods discussing how a code unit works or how a team member deals with technical debt need this activity. Moreover, good documentation speeds up the development and maintenance processes and reduces the money and time spent referencing the developer who wrote the code. Figure 5 shows how to document the code in an easy-to-read way for developers. This activity matches the agile principle: "agile processes to support a consistent development pace".

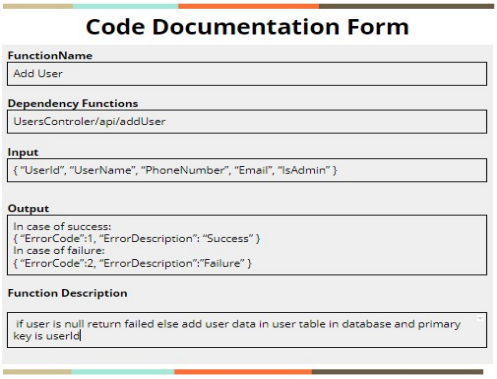


Figure 5: Code Documentation Form

4.3.6 "K" Knowledge transfer activity

This activity is done at the end of the sprint planning. The team votes for a new technology that will be used in another sprint or a new course. Then, one member of the team learns it during the sprint. After that, he shares it with the rest of the team at the end of each sprint following the sprint review meeting. This activity is good for three reasons. First, it enables team members to share skills among each other. Second, it contributes to overcoming the difficulties that the team will face in the upcoming races. Third, it increases cooperation among the team members. This activity matches the agile principle: **"**regular reflections on how to become more effective**".**

5. RESEARCH METHEDOLOGY

This research approach relies on the quantitative method with an analytic survey approach to evaluate the impact of the proposed activities on software development. The evaluation is done by applying the Di-Scrum model in four software companies in Egypt. These companies were working in co-located environment. However, they suddenly shifted to work remotely due to the outbreak of the coronavirus (COVID-19) pandemic. It is a realistic and in-depth study of how the proposed activities influence the distributed team and product quality. The Six-Star Project Management Model [30] is used to check the validity and efficiency of the proposed model on the software companies and the success of its projects. Usually, Project success factors are time, cost, and scope but it is not sufficient to measure the extent of project evaluation. Therefore, PMBOK 4.0 introduced an advanced triple-entry model based on six factors, namely, schedule, scope, budget, risk, resources, and quality as shown in Figure 6, which are critical to the success of the project [31].

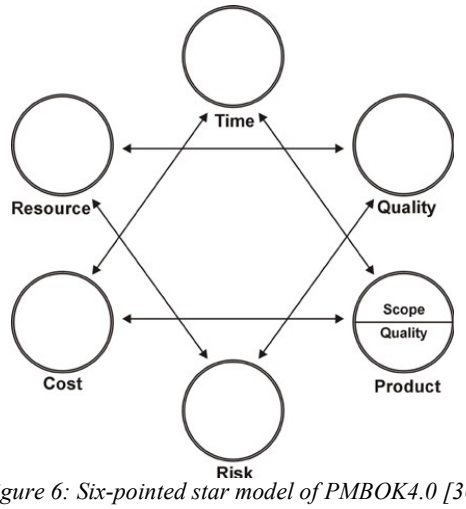


Figure 6: Six-pointed star model of PMBOK4.0 [30]

The questionnaire is divided into three parts. The first part is about general information. The second part includes the success hypotheses for each activity, as shown in Table 3. Finally, the third part covers the factors of the six-pointed star model, as shown in Table 4. In order to elicit the evaluators' opinions, we used a Likert scale from 1 to 5 where 1 refers to strong disagree, 2 refers to disagree, 3 refers to natural, 4 refers to agree and 5 refers to strong agree.

Table 3: Practices validation questions

H. No.	Success Hypotheses of Di-Scrum Model
H1	Increase communication effectiveness
H2	Enhance collaboration among team members
H3	Increase confidence among team members
H4	Save time and add value to teamwork
H5	Match agile principles and manifesto
H6	Enhance understanding of requirements
H7	Improve development quality and test effectiveness
H8	Increase customer satisfaction
H9	Increase rapidly accommodating changing requirements
H10	Reduce technical issues
H11	Enhance sharing interpersonal skills and knowledge via communication
H12	Improve team self-organization
H13	Increase team productivity
H14	Enhance Documentation and Decrease the cost of maintenance
H15	Increase knowledge transferred between team

Table 4: Factor-wise questionnaire used in the evaluating the efficiency of the proposed model

Factor	Q. NO.	Questions
Scheduling	1.1	With using the Di-Scrum model project, the status is clear enough to the project team.
	1.2	Project teams are able to satisfactorily and effectively accommodate changing requirements
	1.3	According to the schedule, the project is delivered on time
Scope	2.1	Clarity of project scope boundaries
	2.2	Effectiveness of the project management methodology to clarify the scope
Budget	3.1	The project has returned with an acceptable investment
	3.2	The budget covers the entire project
Risk	4.1	The ability to manage the risks and opportunities of the project
	4.2	The ability to meet business goals
Resource	5.1	Availability of human and material resources
	5.2	The project team is cooperative and trained enough to reach the expected results
	5.3	The available software tools are sufficient to complete the project
Quality	6.1	Quality requirements are achieved
	6.2	Quality Client satisfaction is achieved
	6.3	In general, the project is successful

Data have been collected through frequent structured and semi-structured online interviews with 40 employees from different teams and in different positions (project managers; senior software developers; junior software developers; quality control team; system analyst), as shown in Figure 7.

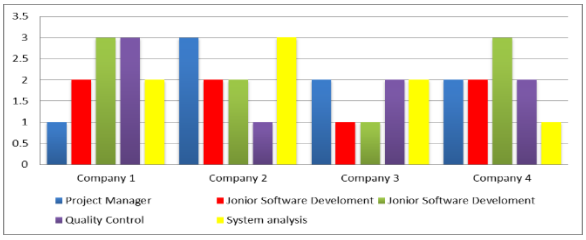


Figure 7: The evaluator's distribution indicated by roles and companies

6. RESULTS

In this section, the statistical analysis obtained from the questionnaire is presented and discussed. The questionnaire shows the extent to which the Di-Scrum model affects all the factors, whether specific to each activity, or on the whole project. In the beginning, we start by analyzing the results of the questionnaire on each activity separately. Then, we analyze the questionnaire responses to show how the management factors are related to each other and how Di-Scrum model affects each factor of the six-pointed star model. Analyzes show the frequency of "agree" addition to "strongly agree" voters for each factor.

6.1 Evaluation of the TAGICK Activates

6.1.1 Evaluation of "T" Timekeeping activity

This activity is evaluated based on measuring the validity of the hypotheses H1, H2, H3, H4, H5 and H6. Figure 8 shows the results of the "T" Timekeeping activity evaluation. Obviously, 80% of the evaluators agree to hypothesis H1. Moreover, 87.5% of them confirm hypotheses H2 and H6. Additionally, 85% of them go along with hypothesis H3; and 85.5% of them verify hypothesis H4. Finally, 90% of them approve hypothesis H5.

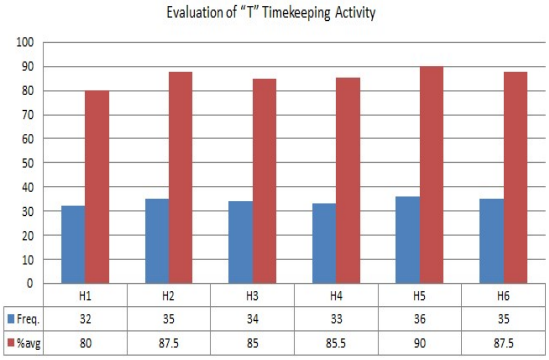


Figure 8: Improvement ratio of "T" Timekeeping activity in the Di-Scrum

6.1.2 Evaluation of "A" Aggregation activity

It has been evaluated according to the same six success hypotheses of activity "T" and H7. Figure 9 displays that 90% of the evaluators agree to hypothesis H1. Furthermore, 85% of them confirm hypotheses H2 and H4. Additionally, 95% of them approve hypothesis H3. Regarding the challenges facing the work teams and enhancing the progress of work, 80% of the evaluators agree to hypothesis H5. In addition, 75% of them verify hypothesis H6. Finally, 85% of them go along with hypothesis H7.

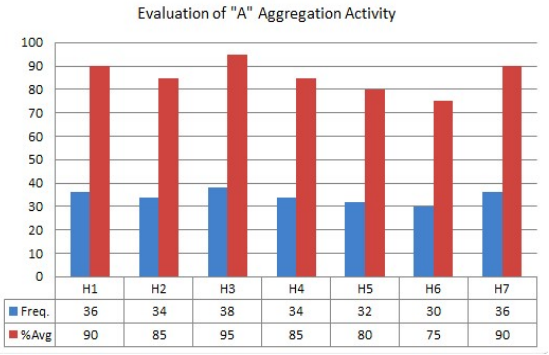


Figure 8: Improvement ratio of "A" Aggregation Activity in the Di-Scrum

6.1.3 Evaluation of "G" Groupthink activity

It has been evaluated according to the same six success hypotheses of activity "T", H8 and H9. Figure 9 shows that 80% of the evaluators agree to hypothesis H1. Furthermore, 87.5% of them confirm hypothesis H2. Moreover, 75% of them go along with hypothesis H3; and 80% of them approve hypothesis H4. Additionally, 85% of them agree to hypothesis H5. Besides, 75% of them approve hypothesis H6; and 86% of them verify hypothesis H7. Finally, 90% of them agree to hypotheses H8 and H9.

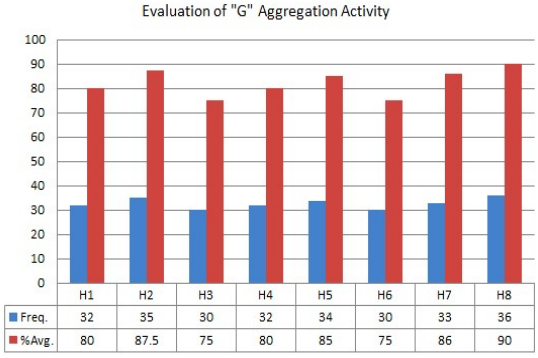


Figure 9: Improvement ratio of "G" Groupthink activity in the Di-Scrum

6.1.3 Evaluation of "I" Interconnectedness activity

It has been evaluated according to the same six success hypotheses of activity "T", H10, H11 and H12. Figure 10 clarifies that 95% of the evaluators agree to hypothesis H1. Additionally, 90% of them approve hypotheses H2, H10, H11 and H12. Moreover, 87.5% of them confirm hypothesis H3; and 85% of them go along with hypotheses H4 and H6. Finally, 80% of them approve hypothesis H5.

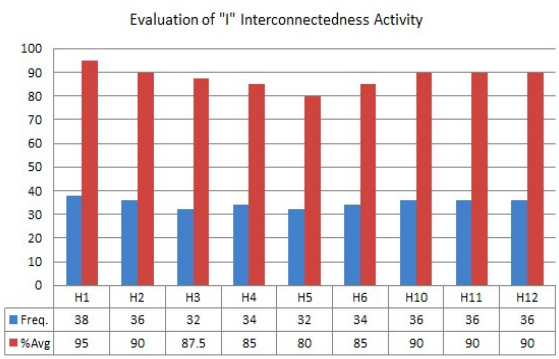


Figure 10: Improvement ratio of "I" Interconnectedness activity in the Di-Scrum

6.1.4 Evaluation of "C" Continuous documentation activity

It has been evaluated according to the same six success hypotheses of activity "T", H8, H13 and H14. Figure 11 explains that 90% of the evaluators agree to hypotheses H1, H2, H8, H13 and H14. Besides, 95% of them confirm hypothesis H3. Moreover, 85% of them agree to hypotheses H4 and H5. Finally, 80% of them go along with hypothesis H6.

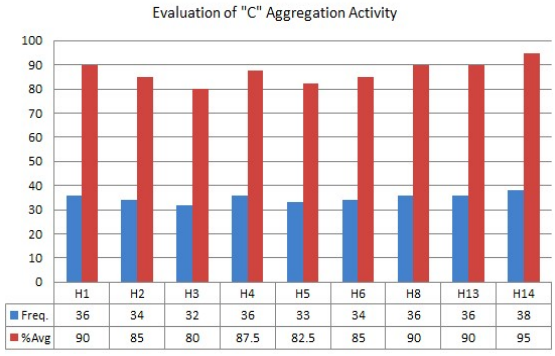


Figure 11: Improvement ratio of "C" Continuous documentation activity in Di-Scrum

6.1.6 Evaluation of "K" Knowledge transfer activity

It has been evaluated according to the same six success hypotheses of activity "T" and H15. Figure 12 shows that 85% of the evaluators agree to hypotheses H1 and H5. Moreover, 87.5% of them confirm hypothesis H2. Furthermore, 80% of them agree to hypothesis H3. In addition, 85% of them approve hypotheses H4 and H5. Besides, 82.5% of them agree to hypothesis H6. Finally, 90% of them agree to hypothesis H14.

Journal of Theoretical and Applied Information Technology
15th September 2021. Vol.99. No 17

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

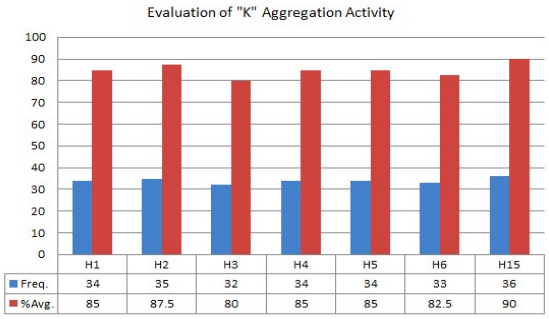


Figure 12: Improvement ratio of "K" Knowledge transfer activity in the Di-Scrum

6.2 Evaluation of the Di-Scrum Model

Table 5 summarizes the responses to the six-star model from the respondents. Concerning the proposed model, the values of "Disagree" and "Strongly Disagree" are much lower than those of "Agree" and "Strongly Agree," as shown in the table. In addition, Figure13 shows the frequency analyses of "Agree" plus "Strongly Agree.

Table 5: The response of questionnaire count concisely

Factor	Questions	Strong Disagree	Disagree	Natural	Agree	Strong Disagree	R.C
Scheduling	With using the Di-Scrum model project, the status is clear enough to the project team (1.1)	0.00%	0.00%	20.00%	33.33%	46.66%	40
		0	0	6	10	14	
	Project teams are able to satisfactorily and effectively accommodate changing requirements (1.2)	0.00%	6.66%	23.33%	33.33%	43.33%	40
		0	2	7	10	13	
	According to the schedule, the project is delivered on time (1.3)	0.00%	3.33%	16.00%	40.00%	40.00%	40
		0	1	5	12	12	
Scope	Clarity of project scope boundaries (2.1)	0.00%	6.66%	26.66%	33.33%	33.33%	40
		0	2	8	10	10	
	Effectiveness of the project management methodology to clarify the scope (2.2)	3.33%	10.00%	13.33%	43.33%	30.00%	40
		1	3	4	13	9	
Budget	The project has returned with an acceptable investment (3.1)	6.66%	3.33%	23.33%	36.66%	30.00%	40
		2	1	7	11	9	
	The budget covers the entire project (3.2)	0.00%	6.66%	16.00%	50.00%	26.66%	40
		0	2	5	15	8	
Risk	The ability to manage the risks and opportunities of the project (4.1)	6.66%	6.66%	30.00%	40.00%	16.00%	40
		2	2	9	12	5	
	The ability to meet business goals (4.2)	0.00%	6.66%	33.33%	33.33%	26.66%	40
		0	2	10	10	8	
Resource	Availability of human and material resources (5.1)	0.00%	6.66%	20.00%	43.33%	30.00%	40
		0	2	6	13	9	
	The project team is cooperative and trained enough to reach the expected results (5.2)	0.00%	10.00%	20.00%	30.00%	40.00%	40
		0	3	6	9	12	
	The available software tools are sufficient to complete the project (5.3)	0.00%	3.33%	20.00%	30.00%	46.66%	40
		0	1	6	9	14	
Quality	Quality requirements are achieved (6.1)	0.00%	6.66%	23.33%	46.66%	23.33%	40
		0	2	7	14	7	
	Quality Client satisfaction is achieved (6.2)	3.33%	10.00%	16.00%	43.33%	26.66%	40
		1	3	5	13	8	
	In general, the project is successful (6.3)	0.00%	6.66%	6.66%	46.66%	40.00%	40
		0	2	2	14	12	

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

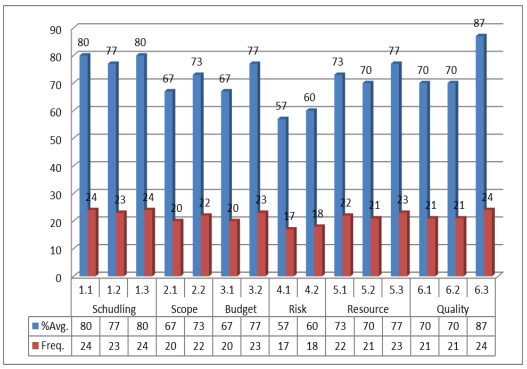


Figure 13: Frequency analyses of the six pointed star model

Table 6 shows the average of each Likert scale for all six-star factors. The average score for each factor are calculated according to the number of evaluators divided into number of the correlation of questions for each factor. For example, computing the cumulative average and the percentage for the schedule factor corresponds to Questions 1.1, 1.2, and 1.3. As it is shown in Table 7, 15.00%, 26.66%, and 32.5% of the respondents are neutral, agreed, and strongly agreed regarding the scheduling of the proposed model. Thus, the mean and the ratio of all factors indicate positive responses from the respondents.

Table 6: Average score of the six-pointed star along with Likert scale

Factors	1	2	3	4	5
Schedule	0.00	1.00	6.00	10.66	13.00
Scope	0.5	2.5	6.00	11.5	9.5
Budget	1.00	1.5	6.00	13.00	8.5
Risk	1.00	2.00	9.5	11	6.5
Resource	0.00	2.00	6.00	10.33	11.6
Quality	0.3	2.33	4.66	13.66	9.00

The quality factor is important factors of the success of the project, as it is directly related to the rest of the success factors of the project. To measure the extent to the quality factor related to each factor. In table 7, we have calculated the Pearson correlation values for the average degrees of all evaluators for the quality factor in relation to the average degrees of the other factors of the six-star model. Correlation is complete if the value is 1, while there is no correlation between the factors if the values are 0.

Table 7: Association between quality factor and project Management factors

Factors	Persons correlation with quality factors
Schedule	0.89
Scope	0.99
Budget	0.82
Risk	0.82
Resource	0.91

The correlation results in Table 8 show that the quality component is favorably connected with the other factors of the six-pointed star model that depicts project success. The factors are highly associated, with correlation values ranging from 0.82 to 0.99. Hence, we analyze the effect of individual factors of the proposed model in Table 8. In this table, we have calculated the mean and standard deviation of each factor of the six-pointed star model. Where there are 40 evaluators and 2 to 3 questions for each factor. Results denote that there are 15 questions refer to the six-pointed star model. Therefore, to compute the mean and standard deviation values we used the collected data. These measures explain the factor-wise levels of satisfaction. The results of the schedule factor display the extent of satisfaction with a factor relative to the other factors.

Table 8: The Six-Star factor's mean and standard deviation values

Factors	Mean	Standard Dev
Schedule	6.13	5.18
Scope	4	4.29
Budget	4	4.71
Risk	4	4.36
Resource	6	4.72
Quality	6	4.99

The proposed model has a positive impact to six success project factor “Schedule, Scope, Budget, Risk, Resources, and Quality”. It work to facilities and manage the communications, collaboration and continues documentation to agile distributed team. Although, in [28], they developed recommendations to develop collective boundaries, increase amount of communication, prioritize consensus higher than majority support, establish rules for interaction and conduct personality tests. But they did not apply these recommendations to real agile distributed teams. In [29], they studied thoroughly the positive and negative effects of communication factors such as technology, culture and trust during the COVID-19 pandemic. But they did not cover the collaboration and continues documentation points.

7. LIMITATIONS

This proposed model was conducted during the Corona pandemic and remote work. Therefore, its scope is limited as well as the data set. Similarly, survey responses were limited to only 4 companies. Suggesting the existence of a Documentation master was important, but it is expensive for some companies. The proposed activities depend on the presence of a strong connection to the Internet; if this condition is violated, it may be difficult to implement these activities. The proposed activities were applied in small-scale distributed software projects.

8. CONCLUSION

The emergence of the Corona pandemic has had a great impact on the field of software development. It has resulted in a shift from a co-located work environment to working remotely. In the present study, we suggest a modification to the Scrum methodology to make it suitable for a distributed environment and to guarantee project quality and customer satisfaction. This has led to designing the Di-Scrum methodology, which consists of a set of activities called the TAGICK activities (Timekeeping; Aggregation; Groupthink; Interconnectedness; Continuous Documentation; and Knowledge Transfer).

The Di-Scrum methodology, which is based on the TAGICK activities, is evaluated using two methods: conducting a questionnaire for four software development companies in Egypt and real case study. The questionnaire has been filled out by 40 employees in different job positions. The results of the questionnaire indicate that most of the evaluators agree there is a noticeable improvement in the Scrum methodology after its conversion to the Di-scrum model enhanced by the TAGICK activities. This has been obvious while studying the distributed teams and their ability to face challenges such as the lack of interconnectedness between means of communication, working time dispersion, lack of cooperation and teamwork, misunderstanding, poorly organized communication and lack of documentation of decisions. In addition, the ability of the team to interact remotely has been improved dramatically, while keeping the agile methodology's principles and rules.

The future work is applying the Di-Scrum model supported by the TAGICK activities to a larger scale projects in order to investigate the extent of their success and develop them. We can try to transform the organization and distribution of tasks from the manual way to the automated method to facilitate remote dealing with team members.

REFRENCES

[1] A. M. Dima and M. A. Maassen,"From waterfall to agile software: development models in the IT sector, 2006 to 2018. impacts on company management", *Journal of International Studies*, vol.11, No.2, 2018 ,pp. 315-326.

[2] JAVANMARD, Mahdi; ALIAN, Maryam. "Comparison between Agile and Traditional software development methodologies" *Cumhuriyet Üniversitesi Fen Edebiyat Fakültesi Fen Bilimleri Dergisi*, vol. 36, no.3, 2015, pp. 1386-1394.

[3] M. A. Akbar, J. Sang, A. A. Khan, F. E. Amin, S. Hussain and et al, "Statistical analysis of the effects of heavyweight and lightweight methodologies on the six-pointed star model", *IEEE Access*, vol. 6, 2018, pp. 8066-8079.

[4] A. Martin, C. Anslow and D. Johnson, "Teaching agile methods to software engineering professionals: 10 years, 1000 release plans", International Conference on Agile Software Development, Springer, Cham, 2017, pp. 151-166.

[5] G. S. Matharu, A. Mishra, H. Singh and P. Upadhyay, "Empirical study of agile software development methodologies: A comparative analysis", ACM SIGSOFT Software Engineering Notes, vol. 40, no. 1, 2015, pp. 1-6.

[6] G. M. Kapitsaki and M. Christou, "Where is scrum in the current agile world?", 9th International Conference on Evaluation of Novel Approaches to Software Engineering (ENASE), IEEE, 2014, pp. 1-8.

[7] F. Anwer, S. Aftab, S. S. Shah and U. Waheed, "Comparative analysis of two popular agile process models: extreme programming and scrum", *International Journal of Computer Science and Telecommunications*, vol. 8, no 2, 2017, pp. 1-7.

[8] H. Alahyari, T. Gorschek and R. B. Svensson, "An exploratory study of waste in software development organizations using agile or lean approaches: a multiple case study at 14 organizations", Information and Software Technology, vol.105, 2019, pp. 78-94.

[9] H. Lei, F. Ganjeizadeh, P. K. Jayachandran and P. Ozcan, "A statistical analysis of the effects of Scrum and Kanban on software development projects", Robotics and Computer-Integrated Manufacturing, vol. 43, 2017, pp. 59-67.

[10] S. Kiv, S. Heng, M. Kolp and Y. Wautelet, " Agile manifesto and practices selection for tailoring software development: a systematic literature review", International Conference on Product- Focused Software Process Improvement, Springer, Cham, vol. 11271, 2018, pp. 12-30.

[11] G. Papadopoulos, "Moving from traditional to agile software development methodologies also on large, distributed projects", Procedia-Social and Behavioral Sciences, vol. 175, no. 2, 2015, pp. 455-463.

[12] D. E. Rush and J. C. Amy, "An Agile Framework for Teaching with Scrum in the IT Project Management Classroom", *Journal of Information Systems Education*, vol. 31, no. 3, 2020, pp. 196-207.

[13] R. Margjoni and M. McClure, "“Scrumming” the library materials budget: aserendipitous application of an agile project management framework", Charleston Library Conference, ,2019.

[14] A. Azanha, A. R. T. Argoud, J. B. Junior and P. D. Antoniolli, "Agile project management with Scrum: a case study of a Brazilian pharmaceutical company IT project", *International Journal of Managing Projects in Business,* vol. 10, no. 1, 2017, pp. 121-142.

[15] Y. Khmelevsky, X. Li and S. Madnick, "Software development using agile and scrum in distributed teams", Annual IEEE International Systems Conference (SysCon), IEEE, 2017, pp. 1- 4.

[16] P. A. Rodriguez, "Conceptual model of communication theories within project process", *INNOVA Research Journal*, vol. 2, no. 3, 2017, pp. 42-51.

[17] A. Al-Zaidi and R. Qureshi, "Global software development geographical distance communication challenges", *The International Arab Journal of* Process", (2017) *INNOVA Research Journal*, vol. 2, no. 3, pp. 42-51.

[18] T. j. Bond-Barnard, L. Fletcher and H. Steyn, "Linking trust and collaboration in project teams to project management success", International Journal of Managing Projects in Business, Vol. 11 No. 2, 2018, pp. 432-457.

[19] P. A. Rodriguez, "Conceptual model of communication theories within project process” *INNOVA Research Journal*, vol. 2, no. 3, 2017, pp. 42-51.

[20] Carrol, John. *Effective project management in easy steps.* In Easy Steps, 2012.

[21] B. Lauren, “Communicating project management: a participatory rhetoric for development teams”, Routledge, 1st Edition, 2018. [22] J. Dias, “Teaching operations research to undergraduate management students: the role of gamification", *The International Journal of Management Education*, vol. 15, no. 1, 2017, pp. 98-111.

[23] Sarhan, S., Abu El Soud, M., Bakry, N., Enhancing Agile Software Development Process Using Learn, Information, Change and Progress Activities, International Review on Computers and Software (IRECOS), 11(3), 2016, pp.239-248.

[24] El Allaoui, M., Nafil, K., Touahni, R., Introducing Model-Driven Testing in Scrum Process Using U2TP and AndroMDA, International Review onComputers and Software (IRECOS), Vol. 12, No. 1, 2017, pp. 30-39.

[25] S. Morrison-Smith and J. Ruiz, "Challenges and barriers in virtual teams: a literature review", *SN Applied Sciences*, vol. 2, 2020, pp.1-33.

[26] A. Winter, "Problems working in semi and full-time virtual teams: comparison of virtual team problems pre and post-Covid 19 epidemic", Bachelor's thesis, University of Twente, 2020.

[27] YERMOLAIEVA, Sofiia. Communication Challenges in Agile Teams from The Communication Theory Prospective. In: *Proceedings of the 2020 European Symposium on Software Engineering*. 2020. p. 88-95.

[28] M. E. Badiale, "The dynamics of communication in global virtual software development teams: a case study in the agile context during the Covid-19 pandemic", Master Thiesis,Uppsala University , 2020.

[29] Marek, Krzysztof, Ewelina Wińska, and Włodzimierz Dąbrowski. "The State of Agile Software Development Teams During the Covid-19 Pandemic." *International Conference on Lean and Agile Software Development*. Springer, Cham, 2021.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[30] Muhammad Azeem Akbar, Jun Sang, Arif Ali Khan, Fazal E-Amin et al. "Improving the quality of software development process by introducing a new methodology—AZ–Model", IEEE Access, vol. 6, 2018, pp. 4811-4823.

[31] L. Howard, et al. "A statistical analysis of the effects of Scrum and Kanban on software development projects." *Robotics and Computer- Integrated Manufacturing* (2015).