University of Sindh Journal of Information and Communication Technology (USJICT)
Volume 5, Issue 1, March 2021
ISSN-E: 2523-1235, ISSN-P: 2521-5582
Website: http://sujo.usindh.edu.pk/index.php/USJICT/

Constant Pressure Mitigation Techniques in Agile Software Development: An Empirical Survey in SME's Pakistan

Muhammad Ibrahim[1], Muhammad Kamran[2]

Department of Software Engineering, University of Engineering and Technology Peshawar[1]
Department of Software Engineering, Abasyn University Peshawar[2]
Muhammadibrahim493@yahoo.com[1], aaafridiii@gmail.com[2]

Abstract: Due to their proved advantages in terms of agility, flexibility and reaction many software developer companies utilizing agile development methods. However, the adoption of such methods presents major obstacles for companies. This study generally examines the dynamics of software development teams operating in Pakistan with Agile. We have surveyed the SMEs in Pakistan because there are few previous research available to address their concerns. In the challenges faced by software development teams, we focused on main problem areas. The study explores main challenges to development teams with agile processes in small and medium-sized enterprises and direct the software companies using agile approaches in Pakistan or other similar countries with a set of guidelines.

Keywords: Agile Practitioners; Team Pressure; Separate backlogs; Agile Sprints; SMEs; Pakistan;

I. INTRODUCTION

Agile techniques are based on an iterative or incremental approach in which analysis, implementation, integration and test cycles are split into smaller projects [1, 2, 3, 4]. For small and Medium scale projects, it is proven fact that agile is more effective for software than conventional waterfall model [5].

Although the software industry adopted agile methods for over a decade and began to increase in 2001 with the Agile manifesto [6]. It began to gain popularity in Pakistan only a decade later [7]. We are aware that agile practitioners in Pakistan fail to switch from conventional development models to Agile based on the previous experience of their first writer in the software development field [8]. The struggle of the industry by adopting agile development practices following research interviews with Pakistani agile practitioners is well understood [9]. We thus perform this exploratory work to define, analyze and possibly solve their issues. No previous research is sufficient to examine and investigate their problem areas within Pakistan's IT Industry.

Since, Scrum promotes a continuous supply in short periods to improve the quality, which increased stress levels of the practitioners that degrade the quality of products [10].

The contribution of this research is an investigation that software developers in Pakistan or similar developing countries can use to identify their stress situations and then try to mitigate those stressed situations with a set of guidelines. In order to carry out this research, we have designed the following research questions (RQs):

RQ1: What are the pressure situations in the implementation of Agile Software Development Methods within SMEs of Pakistan?

RQ2: What are the challenges that software development teams are facing in the constant pressure situation of agile implementation?

RQ3: How to mitigate constant pressure of the development team while implementing agile?

II. LITERATURE REVIEW

Models of the software development assist the organization, range and implementation of software projects in terms of time and within budget [11]. Agile methodologies for developing software have been developed to solve the issues raised by traditional development methods [12]. The main objective of using Agile methods is to ensure a more adaptable, flexible and responsive life cycle for development [13]. Agile approaches have been shown to improve product quality and efficiency as well as to enhance consumer engagement, adaptability and the rapid implementation of software projects [14]. A set of values and concepts have been established for software projects, Agile Manifesto [15]. In order to give guidance and clarification on the application of agile values in software projects, multiple methods have since been defined (e.g., lean, extreme programming and Scrum) [14].

Agile development is based on self-organized teams that work together to adapt to the changing requirements of the customers [16]. Nonetheless, routine stand-ups, sprint scheduling and retrospective practice are most commonly used in Agile processes, with a percentage of 90%, 88%, and 85% respectively [17].

While guidelines and practices on Agile software development are defined, companies, particularly SMEs, tend to "cheer" Scrum and XP selected practices from the full range of available practices [12] [18].Since the agile manifestation in 2001, agile software development has been

an active area of research, but the research literature fails to address specific challenges faced by Pakistan practitioners.

Some research focused on investigating the challenges facing software development practitioners in smaller software firms in specific regions like Sub-Continent, where literature contributions were made to areas like agile awareness and challenges of adoption.

III. METHODOLOGY

Data has been collected for this survey through online emails from different professionals working in different SMEs within Pakistan. The questionnaire has 2indexes. First index is about organization information, which is kept confidential. Based on 1st index, the practitioners have responded to the 2nd index of the questionnaire. There were 5 questions in the 1st index of questionnaire. These were Open-Ended questions, and the respondent organizations were asked to provide information about organization. 2nd index of the questionnaire has 10 questions. These were Close-Ended questions, and each question has 5 legends, ranges from Strongly Agree (SA, value assigned is 1) to Strongly Disagree (SD, value assigned is 5). The questionnaire was mailed to 70 different companies across Pakistan and got 53 responses with a response rate of 75.7%. The confidence level was 95%, with margin of error of 10%. Company size of the respondents are given below:

TABLE 1: COMPANY SIZE OF RESPONDENTS

Developers	Frequency	Percentage
1 to 5 People	20	38.46
6 to 10 People	24	46.15
11 to 20 People	6	11.54
21 to 50 People	1	1.92
Total	52	100.00

IV. DATA ANALYSIS

Data analysis has been made using SPSS. Bar graphs are obtained in order to explain the results of the survey. The frequency table is also generated to verify the results of the bar graph. Frequency table explains frequency of the respondents, their percentage, valid percentage and cumulative percentage. The legends used for each response are SA for Strongly Agree, A for Agree, N for Neutral, D for Disagree, and SD for Strongly Disagree. The following analysis of the survey has been made.

1. Attending meeting every day is stressful

There are several continuous meetings throughout the agile software development life cycle. Everything before the development is planned. The question asked from the respondents that "attending everyday meeting is stressful", the responses are:

TABLE2: ATTENDING MEETING EVERY DAY IS STRESSFUL

Legends		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	SA	9	17.3	17.3	17.3
	A	19	36.5	36.5	53.8
	N	14	26.9	26.9	80.8
	D	8	15.4	15.4	96.2
	SD	2	3.8	3.8	100.0
	Total	52	100.0	100.0	

On the bases of this table we got the following bar graph.

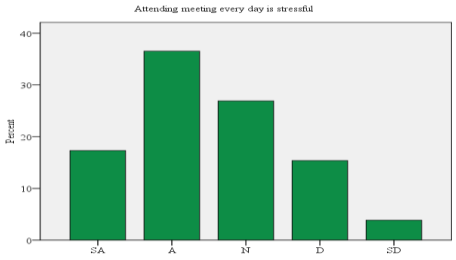


FIGURE 1: ATTENDING EVERYDAY MEETING IS STRESSFUL

According to these responses, most of the respondents are agreed that attending everyday meeting increase stress. This means that as Agile is based on regular meetings, but it creates stress for the development teams. This is because the team must first prepare for the meeting, which can be treated as an additional workload, for each day of meetings. If the structured planning is not needed, then it is likely that the team will skip important items in the discussion. A wide range of respondents (i.e. 26.9%) are neutral in this regard. Suggestion will be made later on in the results portion.

2. Generating Every Day Report is Stressful

For everyday meeting there is a need of generating formal reports as well. Generating reports for each meeting can affect the teams. The question asked, "Generating Everyday Report is Stressful". The survey got following interesting report.

TABLE3: GENERATING EVERYDAY REPORTS IS STRESSFUL

Legends		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	SA	14	26.9	26.9	26.9
	A	21	40.4	40.4	67.3
	N	11	21.2	21.2	88.5
	D	4	7.7	7.7	96.2
	SD	2	3.8	3.8	100.0
	Total	52	100.0	100.0	

For the above table, the bar graph is:

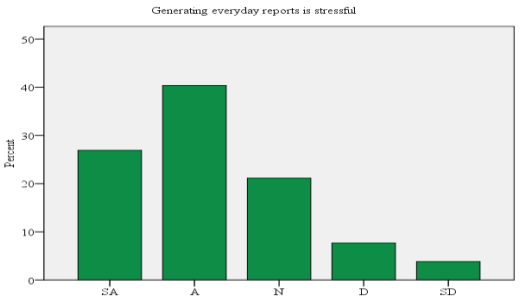


FIGURE 2: GENERATING EVERYDAY REPORT IS STRESSFUL

This result is strengthening the argument that has been made in the previous question. The point was that teams felt stressful at daily meetings due to preparations prior to daily meetings. For everyday meeting, formal reports should be generated at the end of the day which is more stressful for every day. Most of the respondents agreed that generating any kind of report on daily bases increases stress for the developing teams. When it comes to SMEs, this problem becomes more dominant because of limited resources.

3. Small Intervals of Time for Each Release during the Development Process Create Stress

Small releases are considered to be the sole of Agile methods. For small releases, especially in SCRUM, small intervals are been made, often called Sprint. Each interval delivers a milestone or deliverable. For each deliverable, there is a sperate planning, preparing separate backlog, fixing sprint goals, daily stand-up meetings, product backlog refinement meetings, conducting a sprint review, and sprint retrospective. This can be performed for each sprint iteratively. The question was asked, "Small interval of time for each release during the development process create stress", and we got the following results.

TABLE 4: SMALL INTERVALS OF TIME FOR EACH RELEASE DURING THE DEVELOPMENT PROCESS CREATE STRESS

Legends		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	SA	7	13.5	13.5	13.5
	A	18	34.6	34.6	48.1
	N	16	30.8	30.8	78.8
	D	9	17.3	17.3	96.2
	SD	2	3.8	3.8	100.0
	Total	52	100.0	100.0	

The following bar graph explains this table.

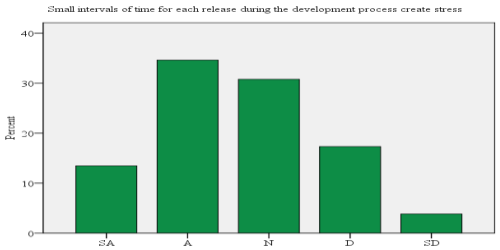


FIGURE 3: SMALL INTERVALS OF TIME FOR EACH RELEASE DURING THE DEVELOPMENT PROCESS CREATE STRESS

According to the results, most of the respondents agreed that small intervals can be stressful. The results show that 30% of the respondents are neutral this regard as well, but still on the agree side the respondent's ratio is greater. The conclusion is that most respondents have accepted the small intervals that generate a burden for SME's development teams.

4. Release in small intervals can increase productivity

As small intervals for small releases is the back bone of the agile methods. Although it creates stress for the development teams, but it increases the productivity. At the same time, the question is asked because if someone thinks that small intervals for the small releases creates stress, so it is understandable that small intervals can increase productivity. The question is asked that, "Releases in small intervals can increase productivity". The responses are:

TABLE 5: RELEASE IN SMALL INTERVALS CAN INCREASE PRODUCTIVITY

Legends		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	SA	12	23.1	23.1	23.1
	A	15	28.8	28.8	51.9
	N	11	21.2	21.2	73.1
	D	10	19.2	19.2	92.3
	SD	4	7.7	7.7	100.0
	Total	52	100.0	100.0	

The responses are shown in the following bar graph.

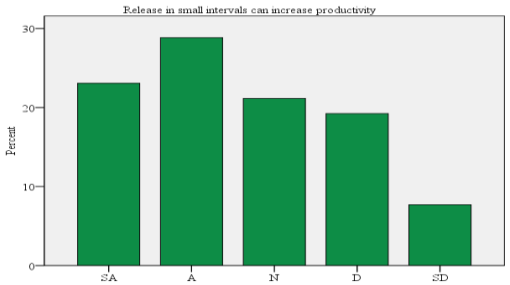


FIGURE 4: RELEASE IN SMALL INTERVALS CAN INCREASE PRODUCTIVITY

The responses show that most of the respondents agreed that small intervals can increase productivity. This encourages that although small releases increase stress but on the other side it increases the productivity of the development teams. The previous analysis contains an argument that can further enhance this finding, i.e. 30.8% respondents to question 3 were neutral regarding the question asked that small releases in small interval increase stress. So, a huge percentage of respondents were neutral.

5. Scrum rushing out the work

There is a deadline for the development teams regarding completion of the sprint. In a complex scenario, these deadlines increase the workload. The development teams are asked to submit their tasks as early as possible. Furthermore, sometimes due to limited resources, multiple tasks can also be assigned to the teams simultaneously. Sometimes, due to bad time estimation, as product backlog evaluates after each sprint, more user stories can be put to the prints which creates stress for the development teams. In order to get concrete evidence regarding this issue, we asked, "Scrum rushing out work".

TABLE 6: SCRUM RUSHING OUT THE WORK

Legends		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	Yes	36	69.2	69.2	69.2
	No	16	30.8	30.8	100.0
	Total	52	100.0	100.0	

The bar graph represented with two legends, Yes or No.

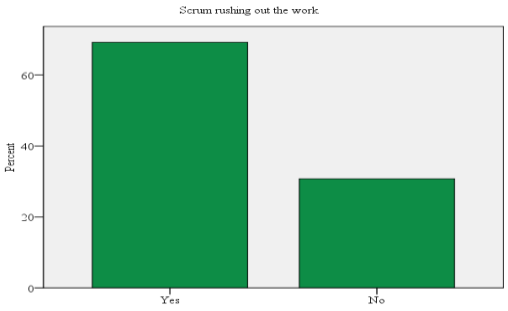


FIGURE 5: SCRUM RUSHING OUT THE WORK

69.2% of the respondents agreed with the argument that has been made. Only 30.8% respondents oppose this stance. This might be because they (respondents) are arguing that as they are continuously evolving the product backlog which helps the developing teams to implement user stories completely.

6. Rushing out effect the quality

It is cleared from the above question that Scrum rushing out the work. But this can adversely affect the quality of the products. In order to answer it, the question was asked, "Rushing out can affect the quality". The responses are:

TABLE 7: RUSHING OUT EFFECT THE QUALITY

Legends		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	SA	12	23.1	23.1	23.1
	A	19	36.5	36.5	59.6
	N	10	19.2	19.2	78.8
	D	8	15.4	15.4	94.2
	SD	3	5.8	5.8	100.0
	Total	52	100.0	100.0	

The results are explained in the following bar graph.

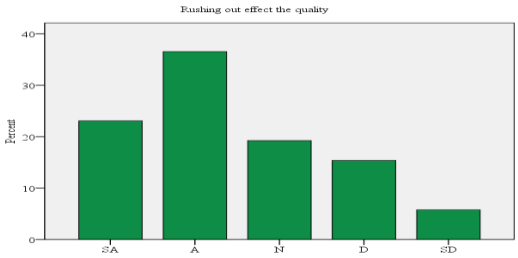


FIGURE 6: RUSHING OUT EFFECT THE QUALITY

The respondents argued that rushing out can affect the quality of the product. It can also affect the productivity of the development. As, teams' evaluation is based on performance, and performance is directly related to productivity. So, with the aim of increasing productivity, rushing out negatively affect the performance of the development teams. If the product quality is not up to the bench mark, this has a bad impression on teams' performance. Concurrently, this has some bad effects on teams' motivations, and it is well known fact that obtaining a high-quality product needs team' motivation. Furthermore, in the rush times of the small intervals this affect can be seen more clearly.

7. Rushing out effect testing

Rushing out has direct effect on the quality, and quality is related with the product testing. The more and more tests can ensure the quality of the products. As from the previous question, it was cleared that rushing out has direct effects on the quality and the whole argument is made. Here, the causes of the adverse effects are explained. For this purpose, the question was asked that "Rushing out effects testing". The following responses are obtained:

TABLE 8: RUSHING OUT EFFECT TESTING

Legends		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	SA	12	23.1	23.1	23.1
	A	17	32.7	32.7	55.8
	N	14	26.9	26.9	82.7
	D	7	13.5	13.5	96.2
	SD	2	3.8	3.8	100.0
	Total	52	100.0	100.0	

The results are as follows.

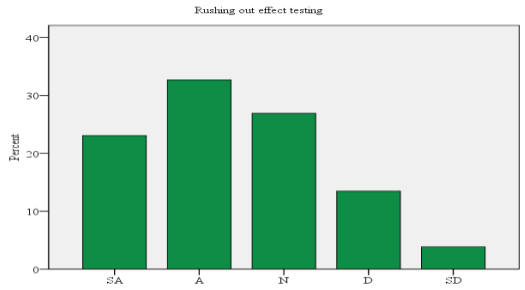


FIGURE 7: RUSHING OUT AFFECT TESTING

According to most of the respondents, it is true that rushing out can affects testing. This is an argument which can be used to strengthen the stance that rushing out affects the quality of the products because of the direct relationship of the testing with quality.

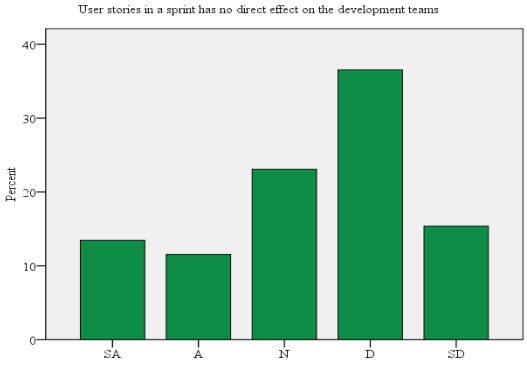
8. User stories in a sprint has no direct effect on the development teams

User stories are considered to be the functional requirements stories of the customers. User stories tell us what to develop. User stories are need to be analyze in order to extract requirements. The contents of the user stories are customer dependent, which means what the customers have written, must be first understood and then implement. In some scenarios, customer put such stories that are hard to be analyzed. Brainstorming can be used to simplify the objectives of the user story which is further a headache for the development teams. Furthermore, sometimes in a single sprint, multiple user stories can be put down. These user stories may be heterogeneous in nature in order to implement an objective of the user stories. In order to verify these arguments, the question was asked that "User stories in a sprint has no direct effect on the development teams". The responses are.

TABLE 9: USER STORIES IN A SPRINT HAS NO DIRECT EFFECT ON THE DEVELOPMENT TEAMS

Legends		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	SA	7	13.5	13.5	13.5
	A	6	11.5	11.5	25.0
	N	12	23.1	23.1	48.1
	D	19	36.5	36.5	84.6
	SD	8	15.4	15.4	100.0
	Total	52	100.0	100.0	

These responses are shown in the following bar graph.



According to these responses, most of the respondents disagreed that sprint has no direct effect on the development teams. Which means that most respondents are agreed to the argument that user stories have direct effect on the development teams. The reason of this is explained earlier. Suggestion will be made later on in the result section.

9. Team leaders estimate everything correct that can avoid stress later on

In agile method, team leaders have a great role/contribution in the development process. Team leader can also be rolled as "Scrum Master" in Scrum, guides the team throughout the development team. Team leaders initiate the project and evaluate the work progress. Before putting the project to the workplace, the team leaders estimate time and budget. This is very critical situation because any uncertainty at this stage can cause great affect later on. Team leaders some time becomes greedy which cause stress for the development teams in the later stages. This is quite obvious and most of the time practitioners have this objection. In order to make an argument regarding this issue, a simple question is asked, "Team leaders estimate everything correct that can avoid stress later on". The responses are:

TABLE 10: TEAM LEADERS ESTIMATE EVERYTHING CORRECT THAT CAN AVOID STRESS LATER ON

Legends		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	SA	3	5.8	5.8	5.8
	A	9	17.3	17.3	23.1
	N	15	28.8	28.8	51.9
	D	15	28.8	28.8	80.8
	SD	10	19.2	19.2	100.0
	Total	52	100.0	100.0	

These responses are shown in the following bar graph.

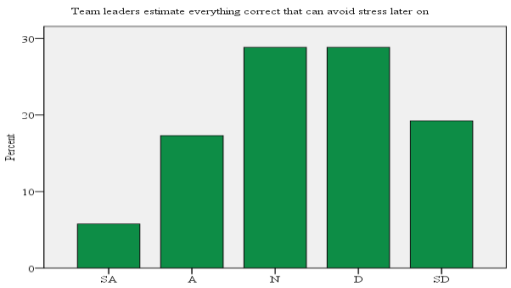


FIGURE 9: TEAM LEADERS ESTIMATE EVERYTHING CORRECT THAT CAN AVOID STRESS LATER ON

The responses show that 23.1%respondents are agreed that team leaders estimate everything correctly, which can avoid stress in the subsequent stages. 28.8% are neutral this regard and 28.8% disagreed with this argument that team leaders' estimation is correct. 19.2% strongly disagreed it. So, it is concluded that most of the respondents are agreed that somewhere the team leaders' estimation can lead the team to stress.

10. Documentation of code can increase the work load of teams

Due to rushing out the work, the development teams face many challenges. One of these challenges is the documentation of code. As the development teams don't have time much time because of team leaders are rushing to finish work as soon as possible. Developers don't have much time to document code. Often the teams need extra time for code documentation. If they don't have much time, they often did documentation later on which is treated to be an extra work load. In order to verify this argument, the question was asked, "Documentation of code can increase workload of teams". The responses are:

TABLE 11: DOCUMENTATION OF CODE CAN INCREASE THE WORK LOAD OF TEAMS

Legends		Frequency	Percent	Valid Percent	Cumulative Percent
Valid	SA	12	23.1	23.1	23.1
	A	17	32.7	32.7	55.8
	N	13	25.0	25.0	80.8
	D	7	13.5	13.5	94.2
	SD	3	5.8	5.8	100.0
	Total	52	100.0	100.0	

The responses are shown in the following bar graph.

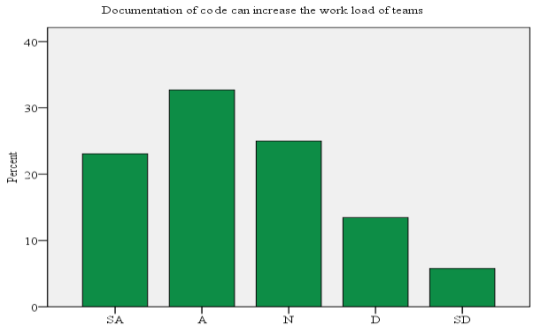


FIGURE 10: DOCUMENTATION OF CODE CAN INCREASE THE WORK LOAD OF TEAMS

23.1% respondents are strongly agreed with the argument made before. 32.7% respondents agreed with the agreed and 25% are neutral this regard. This is a strong evidence, which shows that documentation of code in the later stages creates stress and is treated to be as extra work load.

V. RESULTS

The analysis of data shows some conclusive evidences from Pakistani software industry. The analysis shows that in the following situations the development teams feel more stressful. A set of guidelines is also presented at the end of section in which the study tries to solve these pressure situations.

Attending everyday meetings is stressful

- Generating everyday reports is stressful
- Small intervals of time for each release during the development process create stress
- Release in small intervals can increase productivity
- Scrum rushing out the work
- Rushing out effect quality
- Rushing out effect testing
- User stories in a sprint has direct effect on the development teams
- It is not true that team leaders estimate everything correct that can avoid stress later on
- Documentation of code can increase the work load of teams

The following suggestions are been made in order to resolve these issues, which can help to avoid pressure which is the main cause of pressure.

- ✓ It is the responsibility of the team leaders to constantly observe the team's social and psychological aspects
- ✓ Product owners have the responsibility to indicate that the teams are under pressure
- ✓ Project managers must rely on the teams while estimating time. He/she needs to overestimate, so the teams will have time to complete their jobs rightly.
- ✓ There must be a story points for the backlog items. This will help the development teams to select that how much work they can do in the upcoming sprint.
- ✓ Team leaders need to trust the teams while estimating their work items for a sprint.

- ✓ Sprints length must be kept same throughout the project, i.e. 1 to 4 weeks
- ✓ There should be something for the recreation of the team. This will make them relax for the next day work.

CONCLUSION

A survey by Pakistani software industry SMEs provides some exclusive facts. It is concluded that most respondents have argued that in the industry there are some pressures situation that contribute to stress the development teams. These stresses come from both the management and the standards implemented during agile implementation. In order to prevent such situation, practitioners in the industry pursue multiple activities. This study proposes a number of points for Pakistani small and medium-sized enterprises to take account of the sociological and psychological dimensions of the development teams.

It is proposed that in future it will be possible to conduct a full case study, to basically observe behaviors, record these behaviors and then report it. At the time of the interview, we hope to concentrate on individual agile methods.

ACKNOWLEDGMENT

We dedicate our work to the beloved papers and teachers for supporting us throughout our academic career. We are thankful to the respondents who respond to our survey. Without whom, we were unable to carry out this research.

REFERENCES

[1] S Shahzad, A Keerio, and S Nazir, "Training for Agile Transformation at Universities: A Case Study Analysis," Sindh University Research Journal(Science Series) SURJ Vol 50 No 4 (2018).

[2] Y. A. Solangi, Z. A. Solangi, A. Maitlo, S. Chandio, and A. Shah, "Agile Methods Acceptance Model: Understanding Factors for Implementing Agile Methods in Pakistan," Sindh University Research Journal (Science Series) SURJ, Vol 51 No 3 (2019).

[3] W Aslam and F. Ijaz, "Identification of Dependencies in Task Allocation during Distributed Agile Software Development," Sindh University Research Journal (Science Series) SURJ, Vol 51 No 1 (2019).

[4] S. Shafiq, Y. Hafeez, S. Ali, N. Iqbal, and M. Jamal, "Towards Scrum Based Agile Framework for Global Software Development Teams," Mehran University Research Journal of Engineering & Technology, Volume. 38, No. 4, pp. 979–998 (2019).

[5] K. Saeed, Y. Hafeez, S. Ali, M. U. Shahid, and N. Iqbal, "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, pp. 341–350 (2019).

[6] A. S. Hashmi, Y. Hafeez, M. Jamal, and S. Ali, "Role of Situational Agile Distributed Model to Support Modern Software Development Teams," Mehran University Research Journal of Engineering & Technology, Volume. 38, No. 3, pp. 655–666 (2019).

[7] M. A. Ali, "Survey on the State of Agile Practices implementation in Pakistan," International Journal of Information and Communication Technology Research, Volume. 2, No. 2223–4985, (2012).

[8] B. Maqbool, F. Ur Rehman, M. Abbas, and S. Rehman, "Implementation of scrum in Pakistan's IT industry," 2nd International Conference on Management Engineering, Software Engineering and Service Sciences, At Wuhan, China, pp. 139–146, (2018).

[9] S. Siddiq, "Implementation Issues of Agile Methodologies in Pakistan Software Industry," International Journal of Natural and Engineering Sciences, Volume. 8, No. 3, pp. 43–47, (2014).

[10] J. V Rao and K. Chandraiah, "Occupational stress, mental health and coping among information technology professionals," Indian journal of occupational and environmental medicine, Volume. 16, No. 1, pp. 22–26, (2012).

[11] S. T. Acuña, A. De antonio, X. Ferré, l. Maté, and m. López, "The Software Process: Modeling, Evaluation and Improvement," Handbook of Software Engineering and Knowledge Engineering, Volume. 0, No. 0, pp. 193–237, (2001).

[12] M. STOICA, M. MIRCEA, and B. GHILIC-MICU, "Software Development: Agile vs. Traditional," Informatica Economica, Volume. 17, No. 4, pp. 64–76, (2013).

[13] E. Altameem, "Impact of Agile Methodology on Software Development," International Journal Computer Technology and Electronics Engineering, Volume. 8, No. 2, pp. 2246, (2015).

[14] M. Ibrahim, M. J. Khan, and A. Salam, "Comparative analysis of scrum and XP in Pakistani software industry," Journal of Software Engineering & Intelligent Systems, Volume. 2, No. 3, pp. 199–215, (2017).

[15] P. Hohl et al., "Back to the future: origins and directions of the 'Agile Manifesto' – views of the originators," Journal of Software Engineering Research and Development, Volume. 6, No. 15, pp. 1–27, (2018).

[16] R. Hoda, J. Noble, and S. Marshall, "Self-Organizing Roles on Agile Software Development Teams," IEEE Transactionon Software Engineering, Volume. 39, No. 3, pp. 422–444, (2013).

[17] A. A. Mohallel and J. M. Bass, "Agile software development practices in Egypt SMEs: A grounded theory investigation," IFIP Advances in Information and Communication Technology, Volume. 551, pp. 355–365, (2019).

[18] A. M. M. Hamed and H. Abushama, "Popular agile approaches in software development: Review and analysis," International Conference on Computer, Electrical and Electronics Engineering: "Research Makes a Difference", ICCEEE, pp. 160–166, (2013).