Communication Patterns of Kanban Teams and their Impact on Iteration Performance and Quality

Saad Shafiq, Irum Inayat
SEAL, National University of Computer and Emerging Sciences, Islamabad, Pakistan
saad4is@hotmail.com, irum.inayat@nu.edu.pk

Muhammad Abbas
RISE SICS, Research Institutes of Sweden, Västerås, Sweden
muhammad.abbas@ri.se

Abstract: Software development industry is growing rapidly and so are the time and budget constraints getting stringent. After Scrum, the widely adopted agile method, agile practitioners are now shifting towards Kanban due to its effective communication facilitation, transparency and limited work in progress traits. Since, the industry is in transition from scrum to Kanban therefore we don't find many empirical studies yielding results of adopting Kanban. Therefore, in this study we aim to explore more on Kanban teams. Mainly, we aim to find the impact of Kanban team's communication patterns on their iteration performance and quality. The findings revealed that the centralization communication patterns have negative impact on iteration performance and quality of a project. However, small world communication pattern has positive impact on iteration performance and quality of a project.

I. Introduction

The adoption of Kanban process in software industry is emerging due to its improved visibility, continuous communication, and limited work in progress traits[1][2]. The rising interest of industry towards adopting Kanban method, and its communication pro nature demands the exploration of the collaboration and communication patterns of Kanban teams in different industrial settings [3]. Unfortunately, just a handful of studies have yet been conducted to provide empirical evidence on Kanban teams [4]. For instance, collaboration and communication of Kanban teams [4], impact of adopting Kanban method on software project work [3], and communication overhead measurement within Scrum, Kanban and XP teams [5]. On the other hand, literature shows examples of empirical studies in which communication patterns of traditional and agile teams have been studied (e.g.[1],[6],[7]) and that communication patterns of agile teams may have significant effect on their performance and quality [7]–[9]. However, communication overhead (ratio between communication and development effort) studied in Kanban teams showed that Kanban teams preferred group discussions and had reduced communication as compared to XP teams with more one-to-one communication [5]. This opens up interesting avenues to explore communication among Kanban teams [4]. Literature reveals that using Kanban method has strong impact on teams' communication, yet in the first few iterations. As the teams gets to know each other the message board loses its significance in facilitating the communication among teams. Hence, it shows that interesting patterns may be derived by studying Kanban teams' communication. This motivated us to conduct a study to explore the communication patterns of Kanban teams and to connect it with product quality and iteration performance.

The rest of the paper is organized as follows. Section 2 describes the background and related work. Methodology is discussed in Section 3. Results and Discussion are mentioned in Section 4. Threats to validity and conclusion are described in Section 5 and 6.

II. Background and Related Work

Communication is defined as the information exchange among the team members. Communication pattern is the repetitive communication behavior followed by the team members. Communication patterns can be well represented through communication networks with team members as nodes and existing communication between team members as edges (non-existent communication means no edge between two nodes). Direction of communication is shown through directed edges (if A communicated with B direction is from A to B and from B to A if otherwise). These communication networks help to identify the communication patterns of the team members. For instance, a node (representing team lead) originates most of the edges (communication) to other nodes (team members), it shows that team lead is the most central member throughout the communication networks. Communication patterns of agile teams have been studied in literature (e.g. [7]–[9],[10],[11]). The communication patterns studied commonly include hierarchal, central, and small world patterns. Hierarchal communication patterns are defined as the communication that originates by a team member or a small group of team members to others without it flowing back to them [12].. Centralised patterns are those in which certain member sends and receives most of the communication to/from the rest of the team members. Small world network patterns are defined as the network in which a person is surrounded by

dense cluster of highly connected people and it actually calculates the minimum number of hops required to pass information from one node to another non-neighbor node [11].

Cataldo et. al. studied hierarchal and small world communication patterns of a globally distributed team working on IBM Rational Team concert and studied their effect on iteration performance (number of tasks planned versus completed per iteration) and quality (number of reported bugs) [9]. Results showed improvement in iteration performance and product quality when teams followed hierarchical and small world communication patterns respectively [11]. However, a significant additional improvement in performance and productivity was noticed when team leaders played central role (i.e. shared and received most of the information) in communication networks [13]. Likewise, a framework was proposed to study communication among Scrum teams in [7] and empirical investigation corroborated the results of Cataldo and Ehrlich [9] for small world network patterns. Furthermore, the results showed improvement in performance with increase in in-degree centralization of teams' communication pattern [6].

Similarly, Ehrlich et. al. explored the effect of the position of an individual in a communication network on her performance by studying the development of IBM Rational Team Concert project. Performance measures used in the study were number of work items completed by an individual. The results showed that individuals embedded in a cluster (intra-team communication network) performed better (i.e. solved more bugs) as compared to the central members within the whole project's communication network (inter-team communication networks) [14].

Summarizing above, communication seems a promising candidate to study agile team's productivity and product quality. However, it is so far a less travelled road and needs to be explored further especially for methods like Kanban. Therefore, it is an interesting idea to investigate on "How Kanban teams' communication patterns effects its productivity and quality".

III. Research Methodology

To get a better understanding of Kanban teams' communication and to explore its impact iteration performance and quality, we formulated following hypothesis based on the literature findings.

Avg. Degree Centrality-Performance

H_0: Centralization communication pattern has negative effect on iteration Performance

Avg. Degree Centrality-Quality

H_0: Centralization communication pattern has positive effect on iteration Quality

Small-World-Performance

H_0: Small-World Communication Pattern has negative effect on iteration Performance

Small-World-Quality

H_0: Small-World Communication Pattern has negative effect on iteration Quality

The hypotheses explain the relationship between small-world network measure, centralization value against the iteration performance and quality. These variables are defined as:

1) Clustering Coefficient

Clustering Coefficient is the measure of degree to which nodes tend to cluster or group together.

2) Path Length

The longest or shortest path (distance) between any two nodes in the network is known as path length.

3) Small World

Small world is a network that exhibits a significantly high clustering coefficient and relatively shorter average path length.

4) Degree Centralization

Degree centrality is the measure that counts how many neighbors a node has.

The research methodology followed for this research comprised of three phases, explained below.

A. Phase 1 - Data collection and data pre-processing:

Multiple cases were selected based on following criteria: (a) Teams strictly following Kanban (b) Focus on building new product (c) the project life span less than 10 iterations, (d) the iteration size less than 3 weeks, and (e) the team has to be more than 15 members, (f) language constraint-the communication medium of the teams has to be English. To reduce the external validity threat, we took three projects in our case study.

Open Source project management platforms like Tree.taigo.io, Waffle.io, GitHub, Bitbucket and Sourceforge follow Scrum and Kanban methodologies. These platforms also allow the developers to coordinate implicitly by following each other and by subscribing to the issues of a particular software project [9]. In order to identify communication, we took into account the comments of those Kanban teams on user stories, tasks, and issues. After the selection of three projects, REST API provided by the repository was used to extract the communication data (i.e. comments made) on three items (i) user stories, (ii) tasks, and (iii) issues. We also collected other relevant information about the projects (e.g. no. of user stories, no. of tasks, no. of issues, no. of iterations, start/end date of iterations) and history of the project (e.g. no. of milestones in terms of story points defined/achieved). The data acquired from the repository in JSON format were then manually analyzed after pre-processing. The pre-processing involved mapping the comments made

by each team member participated in the dyadic communication (between two or more persons) and assignment of specific weights to each pair based on the frequency of communication. A matrix was constructed based on the number of comments per team member. If a member comments on another member's created user story, task or issue, a weight of 1 is assigned to it. Afterwards the data were converted into nodes tables and edges tables. Nodes represent the team roles and edges represent the directed communication between them. After analyzing the data manually, excel sheets were prepared to record the dyadic communication that took place between team members on each user story, task, and issue.

B. Phase 2 Construction of communication networks:

The excel sheets were then imported to a social network analysis tool named Gephi (www.gephi.org) to construct communication networks for each iteration separately for user stories, issues, and tasks. Then, we applied social network analysis measures (SNA) (i.e. degree centrality, density, clustering coefficient and path length) to understand the characteristics of the communication networks. For instance, we used clustering coefficient, in-degree centralization and density of networks. These measures eventually helped us in determining the hierarchical and small world communication patterns evolving in these teams [6][15].

C. Phase 3 - Impact analysis:

To find the impact of the communication patterns of Kanban teams we considered two variables i.e. iteration performance and quality. Iteration Performance refers to the rate at which the team is progressing towards completion. It is defined as the number of tasks planned for the iteration but remained incomplete in the end [11]. In order to understand the impact of communication patterns of Kanban teams on iteration performance, we have chosen to assess a factor highly relevant to the performance measure defined in literature i.e., number of achieved story points per iteration.

Iteration Quality is defined as the number of reported bugs per iteration [11]. To measure the quality, we have chosen to assess number of weighted issues (defects) per iteration. The weight is defined as the severity level of the issue raised in the development of iteration. Severity level includes Major, Minor, Important and Critical levels. This concerns with how the team is fixing the defects occurred based on severity during the development of iteration.

After obtaining the hierarchical and small world values, we applied Linear Regression in order to identify the impact of these values on iteration performance and quality. In order to measure Iteration Performance, our Independent variables in this test were the communication patterns (Values) obtained from GEPHI and dependent variable was the Number of story points closed based on total defined story points per iteration. In order to measure Iteration Quality, our Independent variable was communication pattern values associated to issues only and our dependent variable was number of issues fixed per iteration.

D. Projects description

The summary of the selected projects is presented in table 1.

Table 1 Details of case studies

Parameters	Case 1	Case 2	Case 3
Team Size (members)	19	39	24
Duration	11 months	9 months	4 months
Iteration Size	2-3 weeks	2-3 weeks	2 weeks
No. of Iterations	9	8	6
No. of User story	123	151	89
No. of Tasks	142	151	150
No. of Issues	27	90	29
No. of User Story_Comments	45	739	19
No. of Task_Comments	93	303	83
No. of Issue_Comments	4	295	19

All the subjects under discussion employed Kanban as the development methodology. Requirements are expressed in the form of user stories, sub divided into tasks, and issues associated with those user stories and tasks. Roles involved in the projects include Product Owner, Frontend Dev, Backend Dev, UX Designer, Tester, Content Writer and Quality Assurance staff.

IV. Results and Discussion

A. Impact of Communication Patterns on Iteration Performance

The R^2 Value 0.386 of our model indicates that 38% of the variation in our dependent variables is predicted by our independent variables which shows that we can continue interpreting the results as the study suggests that R^2 Values are usually small in case of human-behavior oriented studies [16].

The results of the regression analysis are shown in Table 2 with the β coefficient and significance values. The β coefficient values are negative for Case 2 and Case 3 which indicates that Avg. Degree Centralization communication pattern has negative effect on the iteration's performance except in Case 1 which has a positive effect. In case of small world communication pattern, the β coefficient values are positive in Case 1 and Case 3 and negative for Case 2.s. The β coefficient values for Case 2 and Case 3 are -0.129 and -0.012 which shows that average degree centralization has slightly negative impact on iteration's performance. However, the B coefficient value for Case 1 is 0.012 which is slightly positive. In case of Small

World Communication pattern, the β coefficient values are positive for Case 1 and Case 3 whereas strongly negative for Case 2 which shows that Small World communication pattern has significant negative impact on iteration's performance for Case 2 and slightly positive impact in Case 1 and Case 3. Overall, the results support our hypothesis i.e., "Centralization Communication Pattern has negative impact on iteration's performance" and "Small World Communication Pattern has negative impact on iteration's performance".

Table 2 β coefficient and significance values for performance

Communication Pattern	Case 1 (β, Sig.)	Case 2 (β, Sig.)	Case 3 (β, Sig.)
Avg. Degree Centralization	0.012, 0.155	-0.1296, 0.008	-0.073, 0.031
Small World	0.092878, 0.043	-4.9830, 0.031	2.2796, 0.047

B. Impact of Communication Patterns on Iteration Quality

Our results showed that iteration quality was negatively affected by Avg. Degree Centralization Communication Pattern for Case 1 and Case 3. However, Small World communication pattern had positive effect for Case 2 and negative effect for Case 3 on iteration's quality (see Table 3). The β coefficient values are skewed negatively in case of avg. degree centralization which shows negative impact of Avg. Degree Centralization on iteration's quality. In case of Small World communication patterns, the β coefficient values are positive for Cases 1 and 2 but negative for the Case 3 which shows positive impact of Small World communication patterns on iteration quality. The results support our hypothesis i.e., "Centralization Communication Pattern has negative impact on iteration's quality" and "Small World communication pattern has positive impact on iteration's quality".

Table 3 β coefficient and significance values for quality

Communication Pattern	Case 1 (β, Sig.)	Case 2 (β, Sig.)	Case 3 (β, Sig.)
Avg. Degree Centralization	-2.666667	0.945, 0.00	-0.763, 0.134
Small World	0.092878, 0.798	2.028, 0.140	-5.29, 0.026

To get a better understanding of the results are shown in table 4.

Table 4 Results of Regression Analysis

Hypothesis		Case 1	Case 2	Case 3
Centralization	Performance	Positive (Rejected)	Negative (Not Rejected)	Negative (Not Rejected)
	Quality	Negative (Not Rejected)	Positive (Rejected)	Negative (Not Rejected)
Small World	Performance	Positive (Rejected)	Negative (Not Rejected)	Positive (Rejected)
	Quality	Positive (Rejected)	Positive (Rejected)	Negative (Not Rejected)

As you can see centralization communication patterns has positive impact on performance for case 1 and negative for case 2 and 3 whereas it has positive impact on quality for case 2 and negative for case 1 and 3. Similarly, small world communication pattern has negative impact on performance for case 2 while positive for case 1 and 3 whereas it has negative impact on quality for case 3 while positive for case 1 and 2.

This research draws attention to the fact that studying communication of software development teams may lead to interesting results regarding their performance and product quality. In this regard Kanban teams with slightly larger teams and transparent workflow seemed an interesting case to be explored. In addition, it provides a good opportunity to question the findings of existing studies conducted on Scrum teams to find similarities and differences in the communication behaviors of both teams.

Communication among open-source teams is maintained through online web-portals. Hence, it gives a systematic way to gather communication data rather than collected self-reported questionnaire or interview-based data. The results from this study reconfirm the importance of inter-team communication among software development teams on user stories, tasks, and issues. Our results promote the role of centralized and small world communication patterns among geographically dispersed teams following computer-mediated communication in case of open source projects.

The results of our study partially corroborate with the results of previous studies for small world communication patterns stating that small world communication pattern has positive effect on iteration performance and negative in effect on iteration quality. This shows that small world styled communication among software development teams leads to more erroneous products. The reason may be close interaction among some clusters only. On the contrary, when teams follow small world structures for their information exchange performance improves. This might be due to the quick completion of planned tasks. However, the quality suffers as a consequence.

Our results for centralized communication patterns contradict with the results of previous studies results (e.g.[9],[11],[12]) saying that centralization communication patterns has negative impact on performance and quality. However, in our study centralization communication patterns showed negative effect on performance and quality for two out of three cases. This explains the communication patterns of Kanban teams are slightly different that the ones exhibited by scrum teams.

V. THREATS TO VALIDITY

The study has several limitations. First, we have used open-source project management repository to pick our three subject cases with variable duration and small team size. Therefore, the results cannot be fully generalized to all sorts of projects following Kanban method. Second, we have not

considered any other control factors such as team member's experience, individual task completion rate, throughput etc. Finally, we have analyzed the communication data retrieved from the online repository manually. This might have led to misinterpretation. Although, we tried to overcome this threat by double checking the analysis done by first and second author.

VI. Conclusion

To summarize, this study explores the communication patterns of Kanban teams to find their impact on performance and quality. The results showed negative impact of centralized communication patterns of Kanban teams on performance as well as on quality. For small world communication patterns, Kanban teams tend to have positive impact on their performance as well as quality. The results of this study partially corroborate the literature findings describing the communication patterns of Scrum teams for centralization's negative impact on quality and small world's positive impact on quality. However, the effect size is considerably low in terms of quality for both of the communication patterns.

In future, we plan to expand the study by collaborating with the open project repositories and automating the study through a research framework as accessing a public project with considerable significant iterations is difficult to find. As Kanban is being adopted in many known industries and becoming a trend especially in software development industry, we aim to study the other control factors such as team related factors, communication related factors, technical experience, task priorities etc. that might affect the performance and quality of a project's iteration.

VII. Acknowledgement

This work has been supported by and received funding from the XIVT project (https://itea3.org/project/xivt.html) and TESTOMAT project (https://itea3.org/project/testomatproject.html).

References

[1]A. Deshpande, L. Barroca, H. Sharp, and P. Gregory, "Remote Working and Collaboration in Agile Teams," *Int. Conf. Inf. Syst.*, 2016.

[2]M. O. Ahmad, P. Kuvaja, M. Oivo, and J. Markkula, "Transition of software maintenance teams from scrum to Kanban," *Proc. Annu. Hawaii Int. Conf. Syst. Sci.*, vol. 2016–March, no. i, pp. 5427–5436, 2016.

[3]M. Ikonen, E. Pirinen, F. Fagerholm, P. Kettunen, and P. Abrahamsson, "On the impact of Kanban on software project work: An empirical case study investigation," *Proc. - 2011 16th IEEE Int. Conf. Eng. Complex Comput. Syst. ICECCS 2011*, pp. 305–314, 2011.

[4]N. Oza, F. Fagerholm, and J. Munch, "How does Kanban impact communication and collaboration in software engineering teams?," *2013 6th Int. Work. Coop. Hum. Asp. Softw. Eng. CHASE 2013 - Proc.*, vol. 68, pp. 125–128, 2013.

[5]D. Taibi *et al.*, "Comparing Communication Effort within the Scrum , Scrum with Kanban , XP , and Banana Development Processes Comparing Communication E ff ort within the Scrum , Scrum with Kanban , XP , and Banana Development Processes," no. May, pp. 1–6, 2017.

[6]I. Inayat and S. S. Salim, "A framework to study requirements-driven collaboration among agile teams: Findings from two case studies," *Comput. Human Behav.*, vol. 51, pp. 1367–1379, 2015.

[7]D. Damian, S. Marczak, and I. Kwan, "Collaboration Patterns and the Impact of Distance on Awareness in Requirements-Centred Social Networks," *15th IEEE Int. Requir. Eng. Conf. (RE 2007)*, pp. 59–68, Oct. 2007.

[8]T. La Fond, D. Roberts, J. Neville, J. Tyler, and S. Connaughton, "The Impact of Communication Structure and Interpersonal Dependencies on Distributed Teams," 2012.

[9]M. Cataldo and K. Ehrlich, "The Impact of the Structure of Communication Patterns in Global Software Development : An Empirical Analysis of a Project Using Agile Methods," *Communication*, no. May, p. 17, 2011.

[10]M. Cataldo and K. Ehrlich, "The impact of communication structure on new product development outcomes," *Proc. 2012 ACM Annu. Conf. Hum. Factors Comput. Syst. - CHI '12*, p. 3081, 2012.

[11]M. Cataldo and K. Ehrlich, "The Impact of the Structure of Communication Patterns in Global Software Development : An Empirical Analysis of a Project Using Agile Methods," *Communication*, no. May, p. 17, 2011.

[12]P. Hinds and C. McGrath, "Structures that work: Social structure, work structure and coordination ease in geographically distributed teams," *20th Anniv. ACM Conf. Comput. Support. Coop. Work. CSCW 2006, Novemb. 4, 2006 - Novemb. 8, 2006*, pp. 343–352, 2006.

[13]K. . Ehrlich and M. . Cataldo, "The communication patterns of technical leaders: Impact on product development team performance," *Proc. ACM Conf. Comput. Support. Coop. Work. CSCW*, pp. 733–744, 2014.

[14]K. Ehrlich and M. Cataldo, "All-for-one and one-for-all? A multi-level analysis of communication patterns and individual performance in geographically distributed software development," *Proc. ACM Conf. Comput. Support. Coop. Work CSCW*, pp. 945–954, 2012.

[15]T. Wolf, "Predicting Build Failures using Social Network Analysis," *Icse'09*, pp. 1–10, 2009.

[16]R. A. Gordon, *Regression analysis for the social sciences*. Routledge,2015.