

Technological Communication Tools in Use

The Shape of Knowledge Shared within Software Development Teams

Danieli Pinto¹, Karoline Garcia² and Nelson Tenório³

¹Knowledge Management Masters' Degree Program, UniCesumar, Av. Guedner n.1610, Maringá-PR, Brazil

²Math, Technology, and Agrarian Center (CETA), UniCesumar, Maringá-PR, Brazil

³Knowledge Management Masters' Degree Program, UniCesumar, Maringá-PR, Brazil

Keywords: Technological, Communication Tools, Software Development Teams.

Abstract: Knowledge is a strong competitive source for organizations, giving sustainability, and generating long-term success to them. Communication is the basis of the software project since it provides a way to share knowledge among development team members. Technological tool adoption is one of the strategies to improve communication within software development teams and provide the knowledge sharing among its members. Thus, this paper aims to investigate the shape of the knowledge shared by software development teams through technological communication tools (tech-comm-tools). We collected data through a structured questionnaire and face-to-face interviews. The findings point out the team knowledge is shared by mean texts, photos, and videos according to four categories, such as notification, issues, support to clients, and socialization.

1 INTRODUCTION

Knowledge is something that can be processed and stored in the human mind. It is originating from facts, procedures, concepts, interpretations, ideas, observations and judgments (Wiig, 1997). According to Ipe (2003), knowledge arises through the interaction between individuals at various levels of the organization. However, is only possible to take advantage of the knowledge when it is shared.

Sharing knowledge means to communicating something (Hendriks, 1999) through information, skills, or experience to other individuals (Bukowitz & Willians, 1999), i.e., to make knowledge available to other individuals within the organization, thus creating a link between organization and individual. When this knowledge reaches the organizational levels, it is converted into economic and competitive value for whole the organization (Ipe, 2003). Also, the author suggests that knowledge sharing is not giving up possession of what is known, but rather the act of making knowledge available to others, enabling the recipient to transform knowledge so that it is understood, absorbed, and useful.

Sveiby (1998) highlights that Knowledge Management only makes sense whether people's knowledge can be shared. The author also emphasizes

that a satisfying organizational performance is linked to people's efficiency to create new knowledge, share, and use it for a continuous improvement of the organization and the individuals involved.

There are unknown standards to share knowledge. This sharing can vary among both individuals and companies. They adopt the one that is most effective and best suited to the requirements of the intended recipient, depend on the type of task and knowledge transfer.

Communication plays a significant role within development teams (Hummel, Rosenkranz, & Holten, 2013; Yague, Garbajosa, Diaz, & Gonzalez, 2016) once the software development process is knowledge intensive (Bradshaw, Pulakanam, & Cragg, 2015).

Knowledge is a tacit-asset spreading dynamically within software development teams which evolves technology, changes in software development processes, and organizational culture (Aurum, Daneshgar, & Ward, 2008). To be useful, the individual knowledge must be shared, i.e., socialized. This process requires interaction among team members through face-to-face communication or using a technological tool.

Communication can be established using text messages, photos, and videos through technological tools (Panahi, Watson, & Partridge, 2016). Although

stored into a tool, knowledge is still composed of tacit elements which depend on the individual mental model to be understood and internalized by them.

According to Tenório *et al.* (2017), development team members use technological communication tools (tech-comm-tools) to share project issues among them. Those contents are related to the development of techniques, process, bugs, tools, and so forth. It can happen even when face-to-face interactions are available. This communication can be established easily through personal smartphones, computers, or tablets belonging to development team members and used by them during working hours.

Considering this background, we highlight following research question: *What is the knowledge shape shared by software development team through technological communication tools (tech-comm-tools)?*

Thus, this paper uses an explanatory sequential mixed method (ESMM) approach. It aims to deepen findings presented by Tenório *et al.* (2017) using interviews with software development experts. This research becomes relevant because it can encourage researchers and companies to build a robust knowledge database based on knowledge-shared through tech-comm-tools.

The remainder of this paper is structured as follows. In section 2 is presented the related work. Section 3 describes the research method. In section 4 are shown the findings regarding tech-comm-tools. Finally, in section 5, is shown the conclusion drawn from this paper answering our research question.

2 RELATED WORKS

Software development involves complex tasks with numerous interacting elements. The knowledge within the software development teams is driven by developers' expertise to produce a finished product (Ghobadi & Mathiassen, 2016). In this context, the communication plays an important role in the software development because it is capable of to orchestrate the expertise of the team (or individual) to perform project tasks (Cooper & Sommer, 2016; Yague *et al.*, 2016). However, the communication in software development projects is tacit, informal, and require especial attention for social interaction, *i.e.*, a face-to-face interaction where good quality of it is related to tacit knowledge. Thus, both knowledge tacit and explicit may be common to a group or divided over individuals involving a complex social process that shape the knowledge sharing (Ryan & O'Connor, 2013).

In development teams, knowledge sharing is an important practice for performing daily-project-tasks. That offers a creative way to improve competitiveness and sustainability of the software development enterprises generating explicit knowledge. Santos *et al.* (2015), suggest that the knowledge exchange is achieved by practices adopted for socializing individual expertise within development teams. Licorish and MacDonell (2014) used content analysis within a software developer team to identify knowledge sharing behavior, coming from shared expertise, during textual interaction among developers. For instance, some software development method encourages team members to exchange knowledge among and across them using different ways. According to Larusdottir *et al.* (2017), software development teams are focused on development speed and communication within them.

Ghobadi and Mathiassen (2016), studied communication barriers within software development teams pointing out that knowledge sharing is related to communication quality. For those authors, overcoming communication barriers is a way to spread out knowledge within software development teams increasing their productivity. Also, communication problems generate rework and wrong estimates within software projects. That explains why the key to increasing the productivity and decrease misunderstanding is the improvement of team communication (Cooper & Sommer, 2016). Thus, raises up new parallel communication channels such as forums or tweets as an alternative to face-to-face communication.

Conforto and Amaral (2016) present a framework capable of facilitating the communication among development team members. Majumbar *et al.* (2013) identified benefits and emerging patterns in communication using the social software within software development companies according to manager's perceptions. Rola *et al.* (2016) validated a model for an office space arrangement to support software development teams in carrying out project tasks aiming to improve, among others, communication feature. That model institutionalizes the communication arranging it into physical cells. For instance, face-to-face communication during 'off-task' moments occur into specific social space, such as social kitchen and chill out rooms. In contrast, phone communication occurs into the conference and/or relaxation rooms. The formalized spaces presented by the authors provide knowledge sharing among software development team members even though occurring unstructured information exchange.

Software project information such as

documentation, plans, and models, can be exchanged using tech-comm-tools (Hummel *et al.*, 2013).

According to Daim *et al.* (2012), a tech-comm-tools has two main aspects in a software development project: i) communication planning; ii) communication channels. The first one regarding good practices involving information and communication needs of the project stakeholders. The second one regarding connections between stakeholders in a project. Further, communication has an important role to organize and store organizational knowledge to be used for tracking purposes and decision making (Daim *et al.*, 2012). Boh and Wong (2013) propose the “four-quadrant framework” to identify unofficial and official communication mechanisms. Framework quadrants 1 and 4 describe an expertise shared through unofficial channels, social activities, and informal document exchanges. On the other hand, quadrants 2 and 3 describe a formal knowledge sharing through communities of practices, dialogue sessions, cross-training, repositories, and intranet.

Tech-comm-tools provide knowledge sharing within and across organizations. The knowledge became explicit since it is shared and stored in different formats, such as texts, photos, and videos. Investigate and identify the shape of knowledge sharing is relevant to create a knowledge database to be used by organizations. Next section presents the research method followed by findings and conclusions.

3 RESEARCH METHOD

This research is an explanatory sequential mixed method (ESMM) approach. A mixed-method research combines qualitative and quantitative elements to explore and investigate a problem. This provides more liberty to researcher to approach the problem (Creswell, 2013). According to the author, a ESMM start by a quantitative research and it is complemented by a qualitative research.

In the quantitative research, we adopted the questionnaire suggested by Pinto *et al.* (2016). This questionnaire was answered by fifty-one different software development experts, among them developers, testers, leaders, and managers, from ten medium and small-sized Brazilian software development companies.

The questionnaire contained nineteen questions regarding knowledge management, processes, practices, and communication tools. For this paper, we considered only six questions regarding tech-

comm-tools. They presented in Appendix A of this document. The questionnaire's objective was to investigate how often the tech-comm-tools are used by software development team member. The questionnaire was rating in five points of the Likert scale, such as (5) Always, (4) Frequently, (3) Sometimes, (2) Rarely, and (1) Never (Likert, 1932). The response time to each questionnaire lasted on average fifteen minutes.

We arranged questionnaires' raw data in MS-Excel and calculated the percentage of the Likert scale answers (Creswell, 2013).

In the qualitative phase, we interviewed seventeen different software development experts. All of them responding the questionnaire in the quantitative phase. They were selected randomly by availability. The purpose for this phase was to deepen the findings resulting from the application of the questionnaire. The interview protocol was based in the quantitative phase results. Table 1 shows the interviewees profile. All interviews were face-to-face and recorded after obtaining consent from the participants. The interviewee's identities were hidden and preserved, and each interview took around forty minutes. After that, we transcribed and analyzed the interviews through six steps suggested by (Creswell, 2013).

Two researchers performed all phases. At the end of data analysis, a technical report with research results (phase 4) was written and sent to the participants.

Table 1: Interviewee profile.

Role	# Interviewee	Tech-comm-tools used by
Developer	7	email, Skype, Slack, Whatsapp
Business Analyst	3	
Tester	2	email, Skype, Slack
Project Manager	5	email, Skype, Whatsapp

4 FINDINGS

The questionnaire results evidenced the frequency with which tech-comm-tools are used within software development teams. Figure 1 presents the percentage of utilization for each type of tech-comm-tool. The data shows Email, Slack, Skype, and WhatsApp are

the tech-comm-tools more used. Conversely, Forums, Blogs, FbIM, and SMS do not happen often.

In the following sections, we present the interviews' results that aided us in shaping the knowledge sharing through tech-comm-tolls use by software development team members.

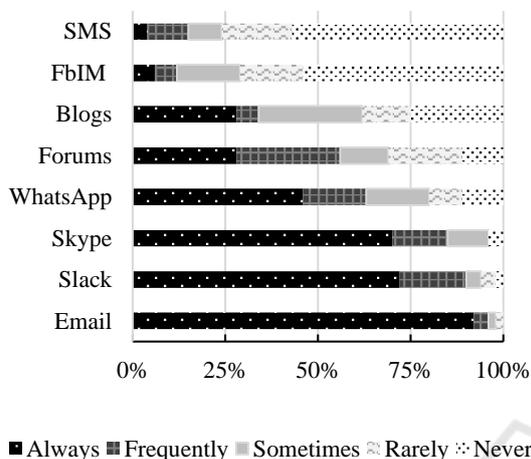


Figure 1: Percentage of tech-comm-tools use.

4.1 Emails

The email is the most used tech-comm-tool, as shown in Table 2. Also, the email is a way to formalize decision taken with clients regarding new functionalities, date to deliver requests, contract values, and so on. Thus, the email is used in formal ways. Interviewees 4 and 12 describe email use in this context-

INT4: *"We always try to formalize the conversations with our clients via email, sometimes they call on Skype or phone reporting system errors and asking for solutions, but this needs to be registered somewhere, so I send an email to register it."* (Interview INT4, date 10/03/2017).

INT12: *"We following talking by Skype and at the end, we say: 'Right, we must formalize it.'. Thus, we write an email, to register the conversation and send it to people involved in the decision. That is necessary because, in the future, can rise some divergences among us and the email can prove the decision that was taken."* (Interview INT12, date 06/04/2017).

According to interviewee 8, emails are used for clients to give their project's feedback to managers and development team-

INT8: *"Last week our leader sent us an email reporting that one of our clients was very satisfied*

with the project. He said that in the beginning, the project was terrible (laughs), but the current releases were very good and with quality." (Interview INT8, date 17/03/17).

The interviews revealed that emails are used to formalize subjects discussed with clients by mean tech-comm-tools or face-to-face interactions. In addition, the individual sends formal messages to company's departments regarding personal bureaucratic and administrative issues.

4.2 Slack & Skype

Following email, the interviewees highlight the use of Slack and Skype. Interviewee 5 describes a discussion of the possibility to adopt a tool to analyse and measure code quality by mean Slack use-

INT5: *"We discussed new rules and tools to measure code quality by means Slack. We were arguing and analyzing what kind of rules and tools were relevant to be adopted. Our result satisfied whole group members and was useful to the company, of course"* (Interview INT5, date 10/03/2017).

Nonetheless, Skype is used mainly for meetings with clients (by video or call) and text messages exchange among team members, as describes interviewee 5-

INT5: *"(...) when we identify some problem or divergence requirement, he [home-office member] is online on the Skype. Then we contact him by text or call asking him to figure it out. He answers us and solves quickly our request"* (Interview INT5, date 10/03/2017).

Interviewees emphasize that both tech-comm-tools provide groups creation (channels in Slack) and specific topics arrange discussions according to their needs-

INT11: *"Slack permit us to create channels. We create one channel to improve our project management communication. Thus, we have some specific channels to notify homologation team and code quality control team regarding issues, such as bugs or a client request. The code quality control team, for instance, exchange knowledge about code standards defined by the group and advises when standards are deviating from the group definition"* (Interview INT11, date 06/04/2017).

INT12: *"We have the main group in the Skype where participate all team members. However,*

those people also participate in other small groups. Thus, we have groups to discuss specific topics. We create groups by topics according to new needs. We have development groups - divided by teams-, homologation group, analysts group, support group, and so on. The communication occurs more intense within subgroups or small groups.” (Interview INT12, date 06/04/2017).

Furthermore, according to interviewee 12 Slack and Skype are used simultaneously by software development teams-

INT12: “We usually use Slack to solve some doubt regarding our task. If it is a technical issue, since I’m a developer, we text it in our developer’s group, such as ‘I have a problem here, do you guys can help me?’. People see it there; they start to discuss sometimes the discussion end into Skype in private, right? Thus, I talk with the person that is helping me until we get to a point where the problem is solved and the conversation ends up” (Interview INT12, date 06/04/2017).

Considering knowledge sharing in Slack and Skype, the interviewees 5 and 10 reveal subjects regarding coding, bugs, standards, task assignments, and task monitoring-

INT5: “We do a lot of things such as, solve coding issues, define new rules for development, new standards. We create a new group in the Skype or new thread in Slack, invite the people, you know? And everyone gives their suggestions about the theme of the discussion” (Interview INT5, date 10/03/2017).

INT10: “We have a ‘general’ project group in which all team members participate. Thus, we send bugs messages, and everyone can read it, you know? We try to help members to solve it. That makes our work easier because if someone has already gone through this, it can solve it faster. I always share solutions in our group. Even if they are situations that I face it, such as bugs, I share so that the people know how to figure it out whether they face it. We are using chat a lot between the team members, even if we were close to people. We used the chat a lot. Sometimes when you’re doing something, I’ve found something interesting, or something to do tasks in a way that’s better that I see... So, I share it.” (Interview INT10, date 30/03/2017).

Interviewee 6 describes an experience sharing a photo to show doubt regarding his code and ask for help in the Skype group-

INT6: “I already took some pictures from my

screen to ask help about my code (or even a configuration). If I have doubts, for example, I do that. One picture can say better than text a message. It is fast to share and intuitive to understand. I prefer it!” (Interview INT6, date 16/03/2017).

The interviews point out Slack as a tech-comm-tools toward to discuss specific technical topics arranged by channels. Conversely, the Skype’s use to go further Slack use because Skype permits meetings by videos and calls, media messages (photos and videos), and screen sharing. Furthermore, Skype is an enabler tech-comm-tools between local and home-office teams for the most projects.

4.3 WhatsApp

The most famous tech-comm-tools is WhatsApp. It is available through personal devices mainly smartphones and has more than 1 billion users around the world. In spite of some companies forbid WhatsApp use during work hours, we observed employees working with the personal smartphones beside them.

That behavior confirms the reason why Table 2 shows WhatsApp in fourth place, as following interview 1 statement-

INT1: “We have Whatsapp group, and we use it. We don’t use it to work, but sometimes somebody text asking something related to work. Also, we text some work’s notices.” (Interview INT1, date 10/03/2017).

The interviewee 2 describes his experience with WhatsApp. He uses it to ask help in development group regarding a component in software development tool (Eclipse IDE)-

INT2: “Once, I’d a doubt about the use of a component in Eclipse IDE. So, I texted my problem in our WhatsApp group. Past five minutes, the solution was sent in the group.” (Interview INT2, date 10/03/2017).

We observed a technical support situation by mean WhatsApp, as describes the interviewee 3-

INT3: “Before yesterday, while I was lying down on my bed, one of our clients texted me in my Whatsapp 11:30 P.M. asking me that something was ‘broken’ in the system and he lost all system database. Thus, I turned on my laptop, and I answered that all his database still was there (...). I still told him to calm down because I identified the issue and the next morning I’ll prioritize it.” (Interview INT3, date 16/03/2017).

As observed in Skype and Slack use, the WhatsApp is arranged by groups. However, the majority interviewees described participating in a single WhatsApp workgroup to discuss several topics both work and funny things (e.g. videos, pictures, gifs). The project managers reported that as healthy behavior because it provides relaxing time and improves interaction among its members. WhatsApp groups also are used to schedule happy hours, parties, and barbecues. Thus, we notice the WhatsApp use just in personal smartphones, while Skype and Slack are used in company's laptops.

4.4 Forums, Blogs, FbIM, SMS

According to interviewees' narratives, there is a low-use of Forums and Blogs. The reason is why the long response time in that tools. Furthermore, they highlight feel more "comfortable" trying to take out doubts first by mean groups in instant messaging tools, such as Skype and WhatsApp. Forums and Blogs are used to take out doubts regarding development tools, process, and practices. Also, we observed enterprise's policies to encourage team members to participate in Forums and Blogs during work hours.

Software development teams do not frequently use FbIM and SMS. The interviewees reported that the Facebook's use is forbidden because "distract and disturb" the workplace. Also, SMS has a cost, and it is not cheap. According to interviewees, it is used in the emergency case to talk to a client or team member that is not online at the moment in the instant messaging tool.

4.5 The Pitfalls of Tech-comm-Tools

Tech-comm-tools presented some pitfalls as interviewees 5 and 6 describe-

INT5: *"The communication can fail. I mean, you think that people understood the message, but they didn't. Thus, you cannot check this out, and it becomes a mess."* (Interview INT5, date 10/03/2017).

INT6: *"When you are texting a message, you can lose some important information, such as physical posture of the person who you are texting. If you look into his/her eyes, you can observe if this person is nervous, calm, and so on. Communication tools is a hard way to identify that and create empathy among people (...) It maybe requires control and people awareness because the focus of the communication must be on work*

and not on personal issues" (Interview INT6, date 16/03/2017).

Therefore, considering software development context, we observed three pitfalls: i) it does not create empathy among people; ii) it can cause communication misunderstanding; iii) it can cause deviate work-focus because personal messages might be exchanged.

5 RESULTS AND DISCUSSIONS

Tech-comm-tools is an alternative tool that provides a fast communication with low-cost. Furthermore, it also allows their users to store and retrieve information that can become knowledge to be used to them during the project performance, as describes interviewee 7 –

INT7: *"Nowadays, digital communication is much better than traditional one, such as face-to-face. The reason is that it can avoid future issues, for instance, somebody says: 'I didn't know about that!' (...). Then, you can show the subject: 'I told you on this date/hour'"* (Interview INT7, date 16/03/2017).

In this sense, our findings point out that the knowledge sharing in software development teams through tech-comm-tools can be shaped in Notification, Issues, Support clients, Socialization, and their subjects.

Notification is the knowledge shared by team members to inform the group about some subject matter. We observed following subjects in this category:

- Standards: group members draw attention to keep or define new standards for the code, names of files, documents, and requests;
- Delivery reminders: group members make aware of delivery and important dates of client requests;
- Change of plans: warn group members about the change of plans, such as a request to be delivered.

Issues are the knowledge sharing by team members to ask help about following subjects:

- System errors: group members report errors occurred during system execution. These errors are investigated by owner of code, i.e., who developed it;
- Doubts: they report doubts about installation and tools use, business rules, and standards

adopted. Some pictures are sent in the group to facilitate the comprehension.

Support to clients knowledge sharing happens when a group member is contacted individually by a client that has its contact. Thus, we observe following subjects:

- Training: to learn a new functionality to the client (user). That occurs sharing videos;
- Bug report: the client (user) send an image to his contact asking and “showing” a misunderstood message. That occurs sharing photos.

Usually, a *Support Bug report* can generate an *Issue System error*. It can happen when the individual contacted by a client unknown a solution to *Support Bug report*. Thus, the individual share the bug in a group asking for help.

Socialization is a peculiar category because it involves various subjects not related to the previous categories. Also, those groups are not available to clients. Thus, socialization is not a formal way to knowledge sharing, *i.e.*, it does not happen through official tech-comm-tools, such as email, Slack, or Skype. Instead of that, it happens by personal smartphone. We notice following subjects:

- Exchange of experience and viewpoints: group members freely expose their opinions about policies, programming languages, processes, databases, poems, texts, and so on. Also, they exchange previous work and life experiences;
- Information disseminating: group members recommend books, movies, links, tools, foods, drinks, and places.
- Funny things: they send funny photos, videos, audio, gifs, and jokes.

Table 2: The shape of knowledge sharing.

Category	Subject	Tech-comm-tools
Notification	Standards; Delivery reminders; Change of plans	email, Slack, Skype, WhatsApp
Issues	System errors; Doubts	Slack, Skype, WhatsApp
Support to clients	Training; Bug report	Skype, WhatsApp
Socialization	Exchange of experience and viewpoints; Information disseminating; Funny things	Whatsapp

The shape of knowledge sharing is presented and summarized in Table 2. The tech-comm-tools used to notify team members is email, Slack, Skype, and WhatsApp. Slack, Skype, and WhatsApp are used to discuss technical issues with both clients and team members. Client support is performed through Skype and WhatsApp. Finally, the socialization is performed just through WhatsApp.

6 CONCLUSIONS

The objective of this paper investigated the shape of the knowledge shared by software development teams through tech-comm-tools use. The research was performed in two phases. The first phase was based on a questionnaire and the second phase was based on seventeen interviews with software development experts from ten different companies.

We observed that email, Slack, Skype, and WhatsApp as the most tech-comm-tools. We found out that email, Slack, Skype, Blogs, and Forums is considered official tech-comm-tools for the most companies. Conversely, WhatsApp, FbIM, SMS are considered unofficial tech-comm-tools.

The interviews revealed that the tech-comm-tools provide an easy communication among and across team members. Furthermore, it promotes, knowledge sharing, storage, and retrieval. However, the interviews also revealed some tech-comm-tools pitfalls, such as: i) they do not create empathy among people; ii) they can generate communication misunderstanding; iii) they can deviate work-focus from team members.

The main contribution here is the shape of knowledge shared by software development team members. Thus, we notice that the knowledge is divided into four main categories and different subjects according to tech-comm-tools related to them.

Identifying the shape of the knowledge shared within software development team through tech-comm-tools is relevant to create an integrated organizational knowledge database.

The next step of this research is to investigate the shape of knowledge shared through tech-comm-tools within software development team in public sector. We intend to compare the findings to improve our actual results. Also, we suggest to investigate others tech-comm-tools, such as Evernote, Wunderlist, and Trello also revealed in our interviews data.

ACKNOWLEDGEMENTS

Our special thanks to *Instituto Cesumar de Ciência, Tecnologia e Inovação (ICETI) – Maringá – Paraná – Brasil*. We also thanks to *Programa de Suporte à Pós-Graduação de Instituições de Ensino Particulares (PROSUP) da CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior)* that provide master's scholarship; to *Fundação Nacional de Desenvolvimento do Ensino Superior Particular (Funadesp)*, that provide an undergraduate scholarship; to Software by Maringá (SbM) entity that aided us to contact the companies; and finally to all companies and their workers, that gently accepted to participate as a volunteer in this research.

REFERENCES

- Aurum, A., Daneshgar, F., & Ward, J. (2008). Investigating Knowledge Management practices in software development organisations—An Australian experience. *Information and Software Technology, 50*(6), 511-533.
- Boh, W. F., & Wong, S. S. (2013). Organizational Climate and Perceived Manager Effectiveness: Influencing Perceived Usefulness of Knowledge Sharing Mechanisms. *Journal of the Association for Information Systems, 14*(3), 122-152.
- Bradshaw, A., Pulakanam, V., & Cragg, P. (2015). Knowledge Sharing in IT Consultant and SME Interactions. *Australasian Journal of Information Systems, 19*, S197-S217.
- Bukowitz, W. R., & Willians, R. L. (1999). *The Knowledge Management field book*. London: FT Management.
- Conforto, E. C., & Amaral, D. C. (2016). Agile project management and stage-gate model—A hybrid framework for technology-based companies. *Journal of Engineering and Technology Management, 40*, 1-14.
- Cooper, R. G., & Sommer, A. F. (2016). From Experience: The Agile-Stage-Gate Hybrid Model: A Promising New Approach and a New Research Opportunity. *Journal of Product Innovation Management, 33*(5), 513-526.
- Creswell, J. W. (2013). *Research design: Qualitative, quantitative, and mixed methods approaches*: Sage publications.
- Daim, T. U., Ha, A., Reutiman, S., Hughes, B., Pathak, U., Bynum, W., & Bhatla, A. (2012). Exploring the communication breakdown in global virtual teams. *International Journal of Project Management, 30*(2), 199-212.
- Ghobadi, S., & Mathiassen, L. (2016). Perceived barriers to effective knowledge sharing in agile software teams. *Information Systems Journal, 26*(2), 95-125.
- Hendriks, P. (1999). Why share knowledge? The influence of ICT on the motivation for knowledge sharing. *Knowledge and Process Management, 6*(2), 91-100.
- Hummel, M., Rosenkranz, C., & Holten, R. (2013). The Role of Communication in Agile Systems Development An Analysis of the State of the Art. *Business & Information Systems Engineering, 5*(5), 338-350.
- Ipe, M. (2003). Knowledge Sharing in Organizations: A Conceptual Framework. *Human Resource Development Review, 2*(4), 337-359.
- Larusdottir, M., Gulliksen, J., & Cajander, Å. (2017). A license to kill – Improving UCSD in Agile development. *Journal of Systems and Software, 123*, 214-222.
- Licorish, S. A., & MacDonell, S. G. (2014). Understanding the attitudes, knowledge sharing behaviors and task performance of core developers: A longitudinal study. *Information and Software Technology, 56*(12), 1578-1596.
- Likert, R. A. (1932). Technique for the measurement of attitudes. *Archives of Psychology, 140*, 44-53.
- Majumdar, A., Krishna, S., & Bjorn, P. (2013). Managers' perceptions of social software use in the workplace: identifying the benefits of social software and emerging patterns of its use.
- Panahi, S., Watson, J., & Partridge, H. (2016). Information encountering on social media and tacit knowledge sharing. *Journal of Information Science, 42*(4), 539-550.
- Pinto, D., Bortolozzi, F., Menegassi, C. H. M., Pugino, P. M. F., & Tenório Jr., N. (2016). Design das etapas a serem seguidas em um instrumento para a coleta de dados para organizações do setor de TI. In VI Congreso Internacional de Conocimiento e Innovación. Bogotá.
- Rola, P., Kuchta, D., & Kopczyk, D. (2016). Conceptual model of working space for Agile (Scrum) project team. *Journal of Systems and Software, 118*, 49-63.
- Ryan, S., & O'Connor, R. V. (2013). Acquiring and sharing tacit knowledge in software development teams: An empirical study. *Information and Software Technology, 55*(9), 1614-1624.
- Santos, V., Goldman, A., & de Souza, C. R. B. (2015). Fostering effective inter-team knowledge sharing in agile software development. *Empirical Software Engineering, 20*(4), 1006-1051.
- Sveiby, K. (1998). *Intellectual capital: thinking ahead*. Australian: CPA.
- Tenório, N., Pinto, D., Vidotti, A. F., Oliveira, M. S. d., Urbano, G. C., & Bortolozzi, F. (2017). Tool Based on Knowledge Management Process: An Interview Protocol to Gather Functional Requirements from Software Industry Experts. *MATTER: International Journal of Science and Technology, 3*(1), 45-54.
- Wiig, K. M. (1997). Knowledge Management: An Introduction and Perspective. *Journal of Knowledge Management, 1*(1), 6-14.
- Yague, A., Garbajosa, J., Diaz, J., & Gonzalez, E. (2016). An exploratory study in communication in Agile Global Software Development. *Computer Standards & Interfaces, 48*, 184-197.

APPENDIX

This appendix presents part of the questionnaire applied within software development companies and full interview protocol regarding tech-comm-tools use.

A. QUESTIONNAIRE

Question: How often do you use technological communication tools to knowledge sharing with your software development team members?

1. Email:

(5) Always, (4) Frequently, (3) Sometimes, (2) Rarely, and (1) Never

2. Blogs:

(5) Always, (4) Frequently, (3) Sometimes, (2) Rarely, and (1) Never

3. WhatsApp

(5) Always, (4) Frequently, (3) Sometimes, (2) Rarely, and (1) Never

4. Skype

(5) Always, (4) Frequently, (3) Sometimes, (2) Rarely, and (1) Never

5. SMS

(5) Always, (4) Frequently, (3) Sometimes, (2) Rarely, and (1) Never

6. Facebook Messenger

(5) Always, (4) Frequently, (3) Sometimes, (2) Rarely, and (1) Never

7. Forums

(5) Always, (4) Frequently, (3) Sometimes, (2) Rarely, and (1) Never

8. Slack

(5) Always, (4) Frequently, (3) Sometimes, (2) Rarely, and (1) Never

9. Other: _____

(5) Always, (4) Frequently, (3) Sometimes, (2) Rarely, and (1) Never

B. INTERVIEW PROTOCOL

Background questions:

1. Gender: _____ Age: _____
2. How long do you work here?
3. What's your position? [Manager, Developer, Tester, Analyst, and so on]
4. How much experience do you have in this position?

Specific questions:

1. What technological communication tool do you use? [Skype, Slack, WhatsApp, email, and so on]
2. Why do you use those tools? For what?
3. Do you participate in any group? [If Yes] What kind of group(s)? Workgroups?
4. How often do you access the group and send any

message?

5. Can you use your smartphone during working hours? [If Yes] How often do you use it?

6. Does the conversation between you and your colleagues happen only during working hours? [If Yes] What kind of tech-comm-tools do you use to do that?

7. What do you prefer? a.face-to-face interaction; b.tech-comm-tools? Why? Which of them is more effective?

8. Does it possible replace face-to-face interaction by tech-comm-tools?

9. What tech-comm-tools do you prefer? Which of them can you use during working hours?

10. Are there any tech-comm-tools that you cannot using during working hours? [If Yes] Which? What are the consequences if you were caught using it?