An Initial Understanding of Task Interdependence in Software Engineering: A Case Study

George Marsicano
University of Brasilia/FGA
Brasilia, Brazil
Center for informatics - UFPE
Recife, PE, Brazil
georgemarsicano@unb.br

Victor L. de Oliveira
Center for informatics - UFPE
Recife, PE, Brazil
vlo2@cin.ufpe.br

Leila M. R. de S. Mariz
Center for informatics - UFPE
Recife, PE, Brazil
lmrsm@cin.ufpe.br

Fabio Q. B. da Silva
Center for informatics - UFPE
Recife, PE, Brazil
fabio@cin.ufpe.br

ABSTRACT

Context: Task interdependence is one of the social characteristics of work design, which has been related by some authors to the level of interaction between team members and their results. In recent years, more research aiming to investigate the interactions between people and teamwork in Software Engineering (SE) has been conducted. However, few of these initiatives have been associated with work design, especially as related to task interdependence in SE. **Goal:** To investigate the perception of the individuals in a software development team concerning task interdependence and their individual impact on teamwork dynamics. **Method:** We investigated a development team from a Brazilian software development company. For data collection, interviews were conducted and qualitative coding techniques were used to analyze and synthesize our findings. In addition, we have the support of an analytical framework built at the commencement of our research. **Results:** Task interdependence increases the need for information sharing and synchronization of tasks, it also favors the creation of an environment conducive to redundancy of knowledge and mutual help, and it is moderated by interpersonal relationships, a sense of belonging, and individual competencies and skills, favoring the generation of better results in software development teams. **Conclusion:** Task interdependence is an important practice and an essential and impacting factor in teamwork dynamics which can enhance the performance of software development teams.

CCS CONCEPTS

• **Software and its engineering → Programming teams**;

KEYWORDS

task interdependence, work design, software engineering, human factors, qualitative research

1 INTRODUCTION

Work design theory focuses on the experience of workers and individual work practices [10]. Other authors have also begun to emphasize the social interaction aspects of workers in their investigations [14]. One of the social characteristics of work design is task interdependence, which some authors have related to teamwork and their performance [4][7][15][23].

Katz-Navon and Erez [13] affirm that the level of interaction among the members of a team is determined, among other factors, by task interdependence. Thus, this factor affects the nature of teamwork processes, shaping relationships between different roles, as well as requiring coordination among a team's members.

In Software Engineering (SE), activities are most often performed by teams, where human factors are vital to their effectiveness [24]. In recent years, teamwork, as well as interaction among team members, has been an increasingly relevant topic in SE, drawing attention to the importance of people and their interactions as part of the success of software construction [20].

As far as research on work design in SE is concerned, the literature is at an early stage [1][5][9][18], especially when it comes to empirical studies on task interdependence.

In order to contribute to the evolution of studies on work design in SE, this research aims to investigate the impact of task interdependence in SE. Thus, this work is primarily guided by the following question: *What are the individual perceptions of the members of a software development team concerning task interdependence and its impacts on teamwork dynamics?*

George Marsicano, Victor L. de Oliveira, Leila M. R. de S. Mariz, and Fabio Q. B. da Silva

In this study, we intend to investigate task interdependence related to the team of developers. That is, testers, designers, and the project leader are considered external to the team of developers (programmers). To address our research question, a case study was conducted in a Brazilian company, with the objective of collecting data on the experience of the developers about task interdependence. This paper presents empirical evidence on the impact of task interdependence on team dynamics and proposes an initial model of task interdependence in SE.

It's important to report that this work is a pilot study and the results herein are preliminary. Further studies will be performed to reinforce and promote adjusts to our initial results. This is the first step toward a better understanding of task interdependence in software development teams with the objective of constructing a task interdependence theory for software engineering.

2 BACKGROUND

Interdependence can be defined as a general sense that team members must depend on each other at work [23]. Therefore, task interdependence refers to a situation where the process and the result of a task affect the process and the result of other tasks [17]. One feature of interdependence is reciprocity, meaning that people are mutually interdependent. Task interdependence involves the design of work in the group or the degree to which group members interact and depend on one another to accomplish work [14]. Kiggundu [14] differentiates task interdependence between initiated and received. First, Initiated Interdependence occurs when the job of one individual affects that of another (the result of "A's" job is input to the beginning of "B's" job). Second, Received Interdependence occurs when an individual's job is affected by the job of others (the commencement of "A's" job depends on the result of "B's" job). Furthermore, according to Kiggundu [14] task interdependence can also be conceptualized as a multidimensional concept with three subdimensions: scope, resources, and criticality.

Outside of SE, some studies suggest that employees react more positively to initiated interdependence than to received interdependence since the first increases the level of cooperation, mutual help and collaborative behavior and the second tends to reduce the level of autonomy, negatively impacting motivation and satisfaction [14][22]. Adams [2] believes that this can be explained by the notion of reciprocity in social exchange and by reciprocity norms. In this context, individuals who are perceived to be more dependent on their peers (received interdependence) will be more motivated to be reciprocal, facilitating the work of others (initiated interdependence).

Referring to the benefits of task interdependence in teamwork, Johnson and Johnson [12] argue that this factor helps, among other things, improve learning and interpersonal relationships between team members [12][22]. De Dreu [6] complements this in indicating that when team members perceive the interdependence of cooperative outcomes they cope better with different opinions, learn more, and perform their tasks more efficiently. Additionally, Kiggundu [14] shows that members who have a sense of responsibility for the work of others achieve better individual results.

In SE, task interdependence is a subject which has not yet been extensively explored. Rehman, Mahmood, Salleh and Amin's [18]

research shows that job interdependence is high in SE profession since developing software requires involvement from multiple specialists, such as the software architect, designer, coder, and tester. In the same way, Ganesh's [9] results indicate that task interdependence creates a need among team members for interacting, coordinating and sharing information, which is essential in software development.

Among recent studies, Acuña, Gómez and Juristo's [1] suggests that interdependence is positively related to satisfaction and that the promotion of these factors by managers will reduce the team's possibility of producing low-quality software. Da Silva, et al. [5] also indicates that task interdependence is related to role conflict. Moreover, the initiated interdependence is negatively related to the interchangeability of roles, indicating that if the individual performs a task that others depend on, he realizes his role as one more difficult for others to perform [5].

Other researchers in SE have investigated the interaction between interdependence and other constructs such as autonomy [11], knowledge sharing [18], conflict resolution and coordination strategy [3], but some results are inconclusive and disconnected. In this study, we propose an initial model that demonstrates the relation between task interdependence and other constructs related to the software development teams' work.

3 METHOD

Our objective in this article is to understand how individual members of a software development team (developers) perceive task interdependence and its impacts on teamwork dynamics. To this end, we performed a qualitative study, with semi-structured interviews based on several studies[8][16][19][21].

3.1 Sample Selection and Analysis Unit Definition

The sample selection strategy adopted was intentional, based on the assumption that the researcher chooses the most appropriate sample to learn about the phenomenon investigated [19].

The criteria used for the sample selection was: (i) there will be no restrictions on the characteristics of the team; (ii) individuals performing the role and primary activities of building (developers) software; (iii) individuals from the same software development team; (iv) individuals linked to the same software development project and physically located in the same work environment.

As for the unit of analysis, checks were be made on the perception of individuals within a particular team.

3.2 Data Collection

Data collection was performed using semi-structured interviews [19]. Four developers from the same team were interviewed. In total, approximately 118 minutes of individual interviews were performed. The interview script was constructed based on initiated and received task interdependence, and the multidimensional concept of task interdependence (scope, resources, and criticality) as presented by Kiggundu [14]. The full interview script can be found in Appendix A.

As the objective of this work was to investigate the perception of the individuals in a software development team concerning task

interdependence and its impacts on teamwork dynamics, the concept of task interdependence was not presented to the interviewees prior to the interviews, in order to not influence results.

All the interviews were recorded and transcribed *ipsi literis*. It should be mentioned that in all interviews, in addition to the interviewee and an interviewer, a scribe was present. The scribe was responsible for recording and taking notes of relevant points of the interview.

3.3 Research Steps

According to Sjøberg, et al. [21] the first step is to identify and define its constructs. To this end, the qualitative analysis of the data was based on the principles of grounded theory for open, axial and selective coding [16]. The research steps were supported by the MAXQDA Plus 12 tool.

Due to the participation of many researchers in the open coding process [16], we built an analytical framework [8], aiming at aligning the perspective of all and obtaining a consensus on the transcribed data. This framework was subsequently used to perform the coding of the other interviews. It should be noted that, whenever necessary, adjustments and/or insertion of new codes and items into the framework were established.

Thus, the open coding of transcripts was performed with the selection of text segments relevant to the research and names (codes). The codes were generated using an iterative approach, for each interview, and constantly compared to each other, both within the same interview and between interviews, to identify similarities and differences. From this, the codes were grouped into categories (axial coding [16]) representing the constructs of teamwork related to the tasks' interdependence. The codes, constructs and their descriptions constitute the analytical framework of the research. The constructs are presented in Section 4.4.

Based on Merriam and Tisdell [16], the second and third steps are related to selective coding. The second step was to specify the propositions [21]. In this sense, from the identification of relationships between the constructs, statements that supported these relationships were identified and extracted from the interviews.

The third step was to present logical and explanatory justifications for each established proposition [21]. Such explanations have a greater level of detail than propositions aiming to explain the phenomenon studied. Each proposition had one or more explanations associated with it, depending on the need. The results of the second and third steps are presented in Section 4.5.

Finally, we compared the findings of this work with data presented in the literature. These comparison are presented in Section 5.1.

3.4 Ethics

In order to follow research ethics regulations, all participants signed an informed consent form complying to the Brazilian Resolution 466/12 - CNS-MS from the National Health Council that regulates research with human subjects. The company signed an Authorization Term, and the researchers signed a Non-Disclosure Agreement.

Table 1: Characteristics of research participants

ID	Role	Age (Years)	Years in IT	Years in Company	Project Months
D1	Developer	27	7	4.5	9
D2	Developer	25	3	2	6
D3	Developer	28	10	3.8	9
D4	Developer	29	6.5	4	4

4 RESULTS

In this section, we first describe the context of the participant organization. We identified two perspectives on task interdependence, presented in Section 4.2. Following this, antecedents and constructs of Task Interdependence (TI) are defined, according to the coding process. Finally, the relations between constructs are described and the model of TI is presented.

4.1 Context Description

The study was conducted in Brazil, in a company created in 2005 through an agreement signed between an Informatics Center of a Higher Education Federal Institution and a multinational corporation in the mobile device and general electronics industry. This company has approximately 80 employees.

The participant team in this research is distributed, with one part employed in the company headquarters and another segment in another city. It is comprised of ten employs, including developers, testers, designers, and the project leader. Given this, and according to the sample selection criteria previously established (Section 3), only the developers (development team) physically located in the company headquarters participated. Altogether four professionals, were invited and were participated alongside with the researchers.

Following the ethical criteria presented in Section 3.4, altogether four professionals were invited by the researchers, and voluntarily participated in the research.

During the interviews, participants reported their experiences with teamwork as well as their perceptions concerning task interdependence in the context of the current project. It is worth mentioning that this development team uses the agile methodology Scrum as the main approach to work, and develops solutions for mobile and web devices.

Table 1 presents some characteristics of the research participants.

4.2 Perception of Task Interdependence

Two types of point of view were perceived among team members regarding task interdependence: internal and external. Internal task interdependence refers to the interdependence among members of the development team itself where participants recognize initiated and received interdependence, from all to all. On the other hand, external interdependence refers to the need for a relationship between the development team and the client, test team, usability team, and project leader. In this type of interdependence, our findings point towards different relationships (initiated and received). These relationships are described below.

4.3 Task Interdependence Antecedents

In order gain a better understanding, some constructs which influences task interdependence were identified during research. That is, depending on their change, task interdependence became stronger, weaker, or non-existent. These constructs are:

Process —The use of a job methodology based on agile practices, in a certain case, can negatively influence the level of task interdependence, since, in the context of agile software development teams, tasks must be broken into small parts so that they can be performed from start to finish independently. On the other hand, constant communication and encouragement of the use of multidisciplinary and self-management teams can be positive factors that facilitate the use of task interdependence.

Team Structure —The team structure (specialist, multi-functional, size, distribution, etc.) is another antecedent that can influence task interdependence since it can change the hierarchical relations among its members, as well as the need to maintain synchronism between tasks.

Task Structure —According to the nature of tasks, their structuring and division can cause more or less task interdependence in a project. Complex large tasks, for example, can generate the need to involve multiple people working in a complementary (interdependent) way. On the other hand, simple tasks can lead to team members to work independently.

Project Moment —During project implementation, the level of interdependence between development team tasks can be changed. Structuring tasks such as setting an architecture, for example, can increase the level of interdependence among team members.

Project Leader Intervention —The project leader, considered as the focal point for the resolution of issues (conflicts) associated with task interdependence interferes in the planning, structuring and distribution of tasks among team members. In this sense, the strategy adopted by the project leader can strengthen or weaken the level of task interdependence in the team.

4.4 Associated Constructs to the Software Development Team's Work affected by Task Interdependence

During our investigations and in structuring our research in alignment with Sjøberg, et al. [21], we have identified constructs related to the work of software development teams which are affected in some way by task interdependence. In this section quotes from the interviews illustrating the scenarios identified are presented. Each developer will be identified by the ID used in Table 1.

The first construct is **Information Sharing**, linked to the creation, registration, and access to the necessary information for software development, as well as constant communication among those involved. The participant's report indicates that task interdependence increases the need for sharing information among those involved in performing the tasks. The results point to two information sharing perspectives: external and internal. With external sharing, the development team needs information from the customer, the test team, the usability team, and the project leader. In sharing information with external agents the development team work is done through tools, documents, and face-to-face conversations. From the internal perspective, information sharing is performed among the members of the development team, again using tools, documents, and face-to-face conversations. On this construct our findings indicate that the development team has an interdependence bond initiated and received in relation to its agents (external and internal).

> *"(...) the requirements are in the tool. If we have a question, we know we can discuss it in a meeting(...). Because without the requirements I cannot accomplish my task."* (D1)

> *"(...) we basically do an alignment by talking about the input, output, what I'm going to generate as output and how you can take advantage of this output as input to your task."* (D3)

The second construct is **Mutual Help**, which is related to the internal work dynamics of the development team. In this context, each team member seeks to help others, either to realize a task or to ask a question. The goal is to make the whole team grow and evolve together, both in the technical context and in relationships. The team's mutual help environment is reported to be quite informal. Therefore, task interdependence benefits the creation of a helping environment among team members.

> *"We try to help, see what can help to make this activity go fast and in the right way (...). The idea is to make the team walk steadily and not stand still because someone had a problem."* (D2)

The third construct is **Task Synchronization**, which refers to the maintenance of sequential task accomplishment. The challenge of Task Synchronization is reported by development team members as being an object of concern among those involved in the sense of not generating delays and/or blocking tasks. Thus, task interdependence increases the need of synchronization between them. Task Synchronization relates to both the activities between development team members (internal) and their activities with external agents (test team). In this sense, our findings point to a link, both of initiated and received interdependence, in relation to its agents (external and internal).

> *"If everyone is synchronized and the progress of my activity helps in the progress of the other, and when the other generate his activity, I can progress in mine."* (D1)

> *"The delay in my task may delay a schedule, because we have the scheduled start date for testing (...). Because of this, we need to keep our activities in sync (...). We are dependent."* (D4)

The fourth construct is the **Knowledge Redundancy**, referring to the strategy adopted by the team, in what concerns promote knowledge sharing among its members, i.e., team members should have the same base knowledge in order to make it possible to allocate any task to any developer. This is a construct linked to the internal team dynamics.

> *"(...) the idea is that everyone can do everything when necessary, if someone is sick, or if someone cannot come for some reason."* (D4)

"As we have the idea that everybody can do everything everywhere [in the software], so that there is no centralization of knowledge(...)" (D2)

"So, I have to think about how I'm going to write the code, because afterwards I have to explain to the team how it works. And also, because I can't be the only one with this knowledge" (D3)

Interpersonal Relationships among team members, point to people with close, such as friendship and partnership. When talking about interpersonal relationships with external agents (client), the team reports point in another direction, with more distant relations.

"(...) we have a very good integration, I know the team well and I consider myself as a friend outside the work environment, it is sometimes difficult to know how to separate this ... I really like my team and we have a good relationship (...). I think this helps a lot in the day-to-day when we need to interact and exchange information." (D2)

Acting in tandem with to Interpersonal Relations, the **Sense of Belonging** emerges as an individual construct, from where each research participant perceive their relationship with the team, feeling part of and believing in the value of teamwork. This construct is linked to internal team dynamics. In this research, it was possible to verify a strong sense of belonging among team members.

"I feel totally part of the team (...). We have total freedom of opinion, of suggestion, of criticism. I can receive positive and negative feedbacks. I'm fully integrated in this group. I have no limitations, I have a lot of freedom." (D4)

The **Competence and Skills** of development team members moderate acts of mutual help and knowledge redundancy. This construct is attributed to the internal team dynamics.

"Then, suddenly a bug can appear that I know my colleague on the side has more experience with(...), we do not have a practice to pass a bug, but ask for support so that we can fix the bug as fast as possible." (D3)

Another important construct identified is the **Guarantee of Norms and Development Standards** associated with the relationship between information sharing and outcomes generated by the development team. This construct is linked to the internal team dynamics.

"I try to modularize as much as possible, I'm tired of refactoring because I made the code and it can be replicated in three other classes ... I try to ensure the application of development standards to make life easier for all of the team." (D1)

From the perception of task interdependence in the development team, our research identifies some **Feelings** reported by the participants, which can moderate the relationship between the results obtained by the team and the phenomenon of task interdependence itself. Such feelings are generated, in part, by the discovery of "unexpected" interdependence. This was not identified during planning and yet does impact the dynamics of teamwork.

"If someone else's work was halted by my mistake, I become sadder, and wanting to solve this as soon as possible." (D3)

Finally, the last construct refers to the team **Performance**, i.e. the results generated. From the reports made by the research participants, the development team recognizes that internally, the relationships can be characterized as interdependence initiated and received among all its members.

"Everyone interferes in everybody's work, because that's how I said it's a small team." (D2)

"(...) the task interdependence can happen from anyone to anyone. And, everyone impacts on the result." (D4)

4.5 Relation between the Constructs and Task Interdependence

Using the reported findings associated with the software development team constructs affected by task interdependence, we returned to our data and identified four key relationships. According to the methodological structure [21], the propositions and the explanations of the relations between the constructs are presented bellow. The relationships between the constructs are represented in Figure 7.

Proposition 1 (P1): the need of increasing **Information Sharing** mediated by **Interpersonal Relationships**, and **Task Synchronization** positively impact software development team performance.

Participants report that both the lack of information (business requirements, interface document, proofs of concept) needed to complete tasks, as well as the mismatch between them, generate direct impacts on the results obtained by the team. In this regard, it is necessary that the interaction among those involved (internal and external to the team) is extended, in order to make the flow of information constant. To do so, the reports point to an exercise in information conversations, keeping up the synchronization between the various activities of the project.

"We talk to see if the task is aligned with what is expected, if it is clear what should be done and when it will be done. Otherwise, if there is a mismatch, our result will be compromised." (D2)

"The more our tasks depend on each other, the more we need to be aware of their execution sequencing. (...). And, we need to talk all the time in order to align information about the project." (D3)

"Good relationships between team members help keep our constants conversations and all information aligned." (D1)

Proposition 2 (P2): the creation of a **Mutual Help** and **Knowledge Redundancy** mediated by individual **Competence and Skills**, **Interpersonal Relationships** and the **Sense of Team Belonging** positively impact the **Performance** of the software development team.

The report of good interpersonal relations between the members of the software development team associated with the strong sense
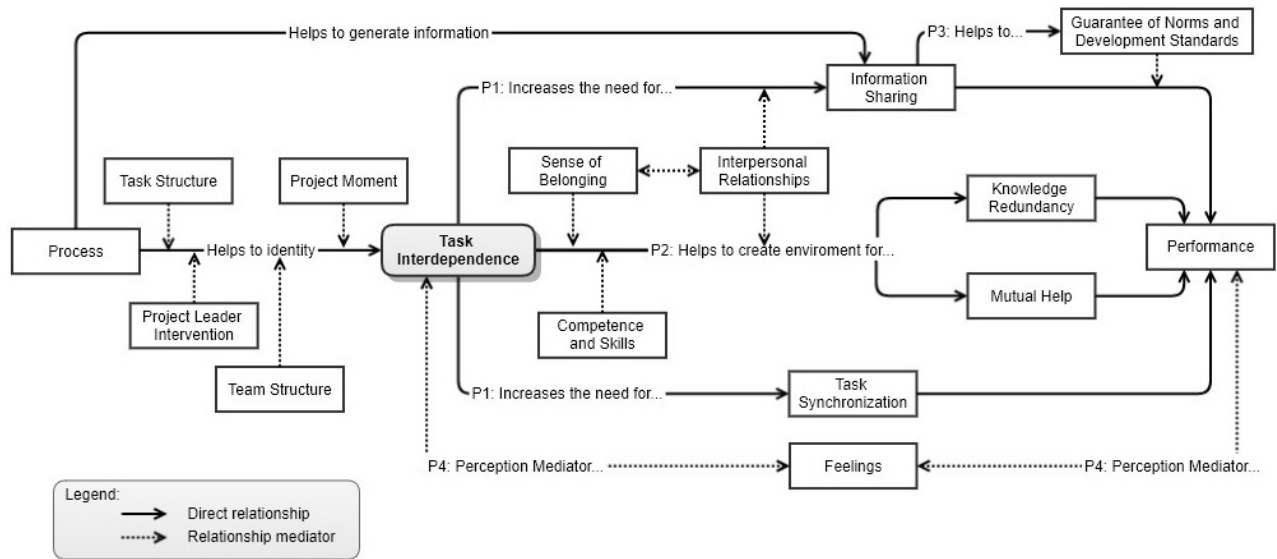
Figure 1: Initial Understanding of Task Interdependence in Software Development Teams.

of team belonging, as well as the diversity of knowledge and skills (seniority and mastery of technologies) facilitate the team's actions and and individual's willingness to give and receive help from each other, as well as share their knowledge, envision collective growth and improve team results.

> "When we first detect the dependency between tasks, we enter into integration mode to when one ends before, the other goes there and tries to help." (D4)

> "The results generated by the members of my team directly impact in my result and vice versa. If someone do something and don't share the knowledge to the rest of the team this can create a further problem." (D3)

> "I feel that as a team member it is important that I help others so that the result is collective rather than individual." (D2)

Proposition 3 (P3): Information Sharing helps to guarantee the use of **Norms and Development Standards** positively impacting the **Performance** of the software development team.

The participant's reports show the need for sharing information and attention to development standards, in order to guarantee good results and to minimize the impact between the tasks of different developers.

> "(...) the most implicit part of the thing, is how the person responsible for the task I'm depending on, has performed his task. Obviously, we discuss during the task. So, we usually know what the person is doing, and if you do something very out of the standards, we go there and give him some advices to get back on track, or vice versa with me." (D1)

> "If not done in a correct and simple way it will directly affect the outcome of others. It can generate bugs, problems of understanding, and then people may not be able to do their work because they can not understanding the code. Your task is totally connected with that of others." (D4)

Proposition 4 (P4): the perception of **Task Interdependence** can be mediated by the **Feelings** generated, from its impact on the **Performance** of the software development team.

From their experiences with task interdependence and the obtained results, team members report their feelings, which tend to direct their perceptions and actions to different paths, impacting the way they treat or will address task interdependence.

> "It happens when, sometimes has a task that impacts everyone, for example, an environment/architecture task. If it's not ready, no one can do it. You are under pressure to finish it." (D2)

> "I do not like when there is any interdependence between the tasks, especially when it interrupts the work of the team, or when it hampers our ability to comply with what was promised in the sprint." (D1)

5 DISCUSSION

5.1 Literature Enfolding

Here, the results of this research are compared with data from the literature presented in Section 1 and Section 2, in order to identify similarities and differences.

Similar to that presented by Katz-Navon and Erez [13], our research points out that the level of interaction among team members is affected by task interdependence. In addition, our research also

indicates the need for an increase in the interaction of the development team with external agents.

Another similarity with Katz-Navon and Erez [13] is concerning the relationship between the team's performance and its results based on the presence of a high level of interdependence, i.e. team results are associated with a high degree of task interdependence, which influences the interaction between team members and the sense of team belonging. On the other hand, the low level of interdependence favors individual performance [13]. In our study, the findings were similar.

Starting from the claims of Taggar and Haines [22] and Adams [2], our findings on the care and concern that the team has to ensure the application of norms and development standards in order to facilitate the work of all can be related to what these authors term the norm of reciprocity. This norm is reflected in the significance between received and initiated task interdependence, summarized as *"The more I receive, the more I give"*. Similarly, our results indicate a reciprocal movement among team members facilitating and ensuring the application of norms and standards. We have identified a different relationship from that presented by Johnson and Johnson [12], who interpreted the interpersonal relationship as one of the benefits of interdependence. Our findings pointed to interpersonal relationships as mediating processes of interaction (information sharing, knowledge redundancy, and mutual help), i.e. the interpersonal relationship would make it easier for the team to deal with interdependence, as opposed to interdependence improving interpersonal relationships among team members. A factor that does not arise explicitly in this paper, but which may be generating benefits to the interpersonal relationships among team members is trust which, according to De Jong, et al. [7] favors making members more available to give and to receive help from colleagues.

5.2 Validity, Reliability, and Limitations

The strategies used to guarantee the internal validity of this research were based on multiple researchers [16] supported by an analytical framework [8], and peer review [16]. Multiple researchers (three) performed the open coding and construction of the analytic framework, which was later used by the first author of this study as a support for axial and selective coding. One expert and another researcher (also authors of this study) helped to validate the results. Finally, two other authors performed the peer review process.

Regarding reliability, the problem investigated by this research is associated with human behavior, which is not static and cannot be isolated [16]. The strategy used to improve the consistency of the results was open coding performed by three researchers, with the revision of one expert.

External validity refers to the possibility of applying results to other situations [16]. Our result is transferable, instead of generalizable. In other words, it is possible to learn from them and see to what extent they may be transferable to other situations. To enhance transferability, we seek to provide a clear description of the research method, the context in which the research was conducted and its results.

Finally, we identified the fact that the case study was performed with only one team and four developers as a limitation. Due to the small number of participants, it can be argued that the propositions presented are potentially explanations in the context of the organization studied. We recognize this limitation, but it does not invalidate the results of our research, firstly as it is the construction of an initial perspective, and secondly because these results are aligned with those of several previous studies.

6 CONCLUSION AND FUTURE WORK

In this research, we presented our preliminary results to understand the perception of software developers concerning task interdependence and its impacts on the dynamics of teamwork. To this end, our study was performed using a team of developers from a Brazilian software development company with about 80 employees.

Our results indicate that task interdependence affects team dynamics, promoting the increase of cooperation and interaction among its members and favoring information sharing, redundancy of knowledge and mutual help. It is important to emphasize that the participants demonstrated good levels of interpersonal relationship and a high sense of team belonging, which reinforces the identified evidence regarding the satisfaction of team members in obtaining collective growth, as opposed to just individual growth.

Finally, the main conclusion of this research is that task interdependence is an important practice and an essential and impacting factor in teamwork dynamics. However, it is necessary for managers and professionals to pay attention to their antecedents and moderators, which can generate negative impacts if neglected, as well as potentialize results when well accompanied. This is the first step towards a better understanding of task interdependence in software development teams.

In the future, it is suggested other case studies using the same work methodology to be carried out in order to reinforce and promote adjusts in our preliminary results. In addition, longitudinal studies to investigate task interdependence and its impacts on the team dynamics over time will also be conducted. Thus, with better and more consistent data we will continue to build a theory on task interdependence for software engineering.

A INTERVIEW SCRIPT

The interview script was developed based on the concepts of initiated and received task interdependence, and on the multidimensional concept of task interdependence (scope, resources, and criticality), both presented by Kiggundu [14]. The interview script was used for all interviews conducted with the members of the development team.

Introduction

- Interviewers should introduce themselves
- Thank the participants
- Ensure confidentiality
- Ask the interviewees to go as far as possible in the questions presented
- Request permission to make recordings

Context of the Participant

- How long have you been working in this company?
- How long have you been on this project?
- Is it okay if I ask how old you are?
- What is your area of training?

- How much experience do you have?
- What is your role in the organization?
- How do you get the job done?

Team

- Do you feel part of a team?
- What role do you play within your team?
- Describe the roles of the other members of your team.
- Tell me a little bit about the division of tasks into the team.

Task Interdependence

- Is it common for your team's tasks to depend on the involvement of more than one person to complete? (*scope*)

Initiated Task Interdependence

- Are there tasks that cannot be performed before you complete yours? (*scope*) How do you feel about this?
- What information do you generate so others can do their jobs? How do you generate this information? (*resource*)
- How do the results generated by your tasks interfere with the performance of the rest of the team? (*criticality*)
- Could you tell which people are directly affected by your work?

Received Task Interdependence

- Are there tasks that you cannot perform before other people complete your assignment (s)? (*scope*) How do you feel about this?
- What information do you need to develop your tasks? How do you get it? (*resource*)
- Do the results generated by other members of your team interfere with your performance? (*criticality*)
- Could you tell which people interfere (have relationship, generate information, etc.) directly in your work? Ask for names of other members.

Closing

- Do you have any questions for me?
- Is there anything else you would like to discuss that was not addressed by the previous questions?

ACKNOWLEDGMENTS

REFERENCES

[1] Silvia T. Acuña, Marta Gómez, and Natalia Juristo. 2009. How do personality, team processes and task characteristics relate to job satisfaction and software quality? *Information and Software Technology* 51, 3 (2009), 627 – 639. https://doi.org/10.1016/j.infsof.2008.08.006

[2] J. Stacy Adams. 1965. Inequity In Social Exchange. Advances in Experimental Social Psychology, Vol. 2. Academic Press, 267 – 299. https://doi.org/10.1016/S0065-2601(08)60108-2

[3] Hayward Patrick Andres. 1996. *The Effects of Task Interdependence, Goal Conflict, and Coordination Strategy on Software Project Success: An Empirical Investigation.* Ph.D. Dissertation. Tallahassee, FL, USA. UMI Order No. GAX96-25655.

[4] Janis A. Cannon-Bowers and Eduardo Salas. 2001. Reflections on shared cognition. *Journal of Organizational Behavior* 22, 2 (2001), 195–202. https://doi.org/10.1002/job.82

[5] Fabio Q. B. da Silva, César França, Cleyton V. C. de Magalhães, and Ronnie E. S. Santos. 2016. Preliminary Findings About the Nature of Work in Software Engineering: An Exploratory Survey. In *Proceedings of the 10th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement (ESEM '16)*. ACM, New York, NY, USA, Article 10, 6 pages. https://doi.org/10.1145/2961111.2962625

[6] Carsten De Dreu. 2007. Cooperative Outcome Interdependence, Task Reflexivity, and Team Effectiveness: A Motivated Information Processing Perspective. 92 (06 2007), 628–38.

[7] Simon B. de Jong, Gerben S. Van der Vegt, and Eric Molleman. 2007. The relationships among asymmetry in task dependence, perceived helping behavior, and trust. *Journal of Applied Psychology* 92, 6 (11 2007), 1625–1637. https://doi.org/10.1037/0021-9010.92.6.1625

[8] Nicola K. Gale, Gemma Heath, Elaine Cameron, Sabina Rashid, and Sabi Redwood. 2013. Using the framework method for the analysis of qualitative data in multidisciplinary health research. *BMC Medical Research Methodology* 13, 1 (18 Sep 2013), 117. https://doi.org/10.1186/1471-2288-13-117

[9] M.P. Ganesh. 2011. Climate in Software Development Teams: Role of Task Interference and Procedural Justice. *Conference on Interdisciplinary Business Research* (2011). https://doi.org/10.2139/ssrn.1869469

[10] J. R Hackman and G. R. Oldham. 1980. *Work redesign.* Addison-Wesley, Reading, MA.

[11] Brian D. Janz, Jason A. Colquitt, and Raymond A. Noe. 1997. Knowledge worker team effectiveness: The role of autonomy, interdependence, team development, and contextual support variables. *Personnel Psychology* 50, 4 (1997), 877–904. https://doi.org/10.1111/j.1744-6570.1997.tb01486.x

[12] David Johnson and Roger T. Johnson. 1989. Cooperation and Competition: Theory and Research. (01 1989).

[13] Tal Y. Katz-Navon and Miriam Erez. 2005. When Collective- and Self-Efficacy Affect Team Performance: The Role of Task Interdependence. *Small Group Research* 36, 4 (2005), 437–465. https://doi.org/10.1177/1046496405275233 arXiv:https://doi.org/10.1177/1046496405275233

[14] Moses N. Kiggundu. 1983. Task interdependence and job design: Test of a theory. *Organizational Behavior and Human Performance* 31, 2 (1983), 145 – 172. https://doi.org/10.1016/0030-5073(83)90118-6

[15] Jared Ledoux. 2009. *Team Task Interdependence Perceptions: Toward an Integrative Model of Teamwork.* Master's thesis. Louisiana State University.

[16] Sharan B. Merriam and Elizabeth J. Tisdell. 2016. *Qualitative research: A guide to design and implementation.* John Wiley and Sons.

[17] Eric Molleman, Aukje Nauta, and Karen A. Jehn. 2004. Person-Job Fit Applied to Teamwork: A Multilevel Approach. *Small Group Research* 35, 5 (2004), 515–539. https://doi.org/10.1177/1046496404264361 arXiv:https://doi.org/10.1177/1046496404264361

[18] M. Rehman, A. K. Mahmood, R. Salleh, and A. Amin. 2014. Work design characteristics and knowledge sharing behavior among Software Engineers. In *2014 International Conference on Computer and Information Sciences (ICCOINS)*. 1–5. https://doi.org/10.1109/ICCOINS.2014.6868447

[19] Per Runeson and Martin Höst. 2008. Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering* 14, 2 (19 Dec 2008), 131. https://doi.org/10.1007/s10664-008-9102-8

[20] Ken Schwaber and Mike Beedle. 2001. *Agile Software Development with Scrum* (1st ed.). Prentice Hall PTR, Upper Saddle River, NJ, USA.

[21] Dag I. K. Sjøberg, Tore Dybå, Bente C. D. Anda, and Jo E. Hannay. 2008. *Building Theories in Software Engineering.* Springer London, London, 312–336. https://doi.org/10.1007/978-1-84800-044-5_12

[22] Simon Taggar and Victor Y. Haines. 2006. I need you, you need me: a model of initiated task interdependence. *Journal of Managerial Psychology* 21, 3 (2006), 211–230. https://doi.org/10.1108/02683940610659560 arXiv:https://doi.org/10.1108/02683940610659560

[23] R. Wageman. 1995. Interdependence and group effectiveness. *Administrative Science Quarterly* 40, 1 (1995), 145–180. https://doi.org/10.2307/2393703

[24] T. Dingsøyr and T. Dybå. 2012. Team effectiveness in software development: Human and cooperative aspects in team effectiveness models and priorities for future studies. In *2012 5th International Workshop on Co-operative and Human Aspects of Software Engineering (CHASE)*. 27–29. https://doi.org/10.1109/CHASE.2012.6223016