UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO
DEPARTAMENTO DE COMPUTAÇÃO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA APLICADA

LUIS FELIPE CAVALCANTI DE AMORIM

Does autonomy influence Software Engineer turnover in distributed teams?

Recife

2023

LUIS FELIPE CAVALCANTI DE AMORIM

Does autonomy influence Software Engineer turnover in distributed teams?

Dissertação apresentada ao Programa de Informática Aplicada da Universidade Federal Rural de Pernambuco, como requisito parcial para obtenção do título de Mestre em Informática Aplicada

Área de Concentração: SOFTWARE ENGINEERING

Orientador (a): Prof. Dr. MARCELO LUIZ MONTEIRO MARINHO

Coorientador (a): Prof. Dr. IVALDIR HONÓRIO DE FARIAS JUNIOR

Recife

2023

Dados Internacionais de Catalogação na Publicação
Universidade Federal Rural de Pernambuco
Sistema Integrado de Bibliotecas
Gerada automaticamente, mediante os dados fornecidos pelo(a) autor(a)

A524d Amorim, Luis Felipe Cavalcanti de
Does autonomy influence Software Engineer turnover in distributed teams? / Luis Felipe Cavalcanti de Amorim. - 2023.
121 f. : il.

Orientador: Marcelo Luiz Monteiro Marinho.
Coorientador: Ivaldir Honorio de Farias Junior.
Inclui referências e apêndice(s).

Dissertação (Mestrado) - Universidade Federal Rural de Pernambuco, Programa de Pós-Graduação em Informática Aplicada, Recife, 2023.

1. Autonomy. 2. Turnover. 3. Distributed Software Development. I. Marinho, Marcelo Luiz Monteiro, orient. II. Junior, Ivaldir Honorio de Farias, coorient. III. Título

CDD 004

Luis Felipe Cavalcanti de Amorim

Does autonomy influence Software Engineer turnover in distributed teams?

Dissertation submitted to the Applied Informatics post-graduation program (PPGIA), Federal Rural University of Pernambuco (UFRPE), as part of requirements to obtain the Master of Science degree in Applied Informatics. The dissertation was approved by unanimous decision in a public session on August 30th, 2023.

EXAMINATION BOARD

_____

Dr. MARCELO LUIZ MONTEIRO MARINHO (Advisor)
Universidade Federal Rural de Pernambuco - UFRPE

_____

Dra. TACIANA PONTUAL DA ROCHA FALCAO
Universidade Federal Rural de Pernambuco - UFRPE

_____

Dr. HERMANO PERRELLI DE MOURA
Universidade Federal de Pernambuco - UFPE

I dedicate this work to God who provided me with a healthy life and with energy to face all challenges, to me for believing in myself and not giving up and to my family, friends and professors who have helped me along this path to achieve my goals.

ACKNOWLEDGEMENTS

I'd like to thank God for allowing me to live in this world and for providing me with the blessing of consciousness to make studies like this.

I'd like to thank my family, especially my parents, Virgínia and Alberes, for providing me with all the basic education and knowledge needed to chase my career goals.

I'd like to thank my girlfriend Marina for supporting me throughout this whole process with a word of caring and motivation.

I'd like to thank all my friends from graduation and from life who always cheered me up when I needed the most.

I'd like to thank my professors from UFRPE who lectured me in this course, especially my advisor and friend Marcelo Marinho, who has been my partner in research since 2019 and who has always helped me with insights and contributions within this study, and professor Ivaldir who also provided me with great support at initial stages.

I'd like to thank all the participants who took some time to answer our research questions helping me to come up with this work.

Lastly, I'd like to thank me for believing in me and for never quitting and for overcoming all challenges presented in life.

RESUMO

Nos últimos tempos, as equipes distribuídas tornaram-se cada vez mais predominantes nas empresas de software. No entanto, as organizações de Desenvolvimento Global de Software (GSD) frequentemente encontram desafios relacionados à alta rotatividade de desenvolvedores. Por outro lado, outras pesquisas indicam que a autonomia e seus fatores associados poderiam atenuar ou prevenir tal rotatividade. Este estudo explora a relação entre autonomia e rotatividade nas equipes GSD. Para atingir esse objetivo, realizamos uma pesquisa baseada em uma Revisão Sistemática da Literatura (SLR) e, em seguida, um survey abrangente e uma pesquisa que envolveu 181 engenheiros de software de diversas localidades globais. Nossas descobertas lançam luz sobre os fatores críticos de autonomia que afetam significativamente a rotatividade em projetos GSD, incluindo reconhecimento, comunicação, colaboração, confiança e equilíbrio de tarefas. Ao oferecer uma compreensão abrangente desses elementos relacionados à autonomia, nossa pesquisa fornece às empresas de software informações valiosas para abordar o problema persistente da rotatividade em projetos GSD. Por meio de uma compreensão mais profunda do papel da autonomia na redução da rotatividade, as empresas podem implementar estratégias direcionadas para aprimorar a dinâmica da equipe, promover uma melhor comunicação e melhorar a colaboração em ambientes de desenvolvimento de software distribuído. O conhecimento adquirido com este estudo será fundamental na criação de um ambiente de trabalho positivo e produtivo para equipes distribuídas, levando a melhores resultados de projeto e sucesso organizacional geral.

Palavras-chaves: Autonomia. Rotatividade. Desenvolvimento Global de Software.

ABSTRACT

In recent times, distributed teams have become increasingly prevalent in software companies. Nevertheless, Global Software Development (GSD) organizations often encounter challenges concerning high developer turnover. Conversely, other research indicates that autonomy and its associated factors could mitigate or prevent such turnover. This study explores the relationship between autonomy and turnover within GSD teams. To achieve this objective, we conducted a survey based on a comprehensive Systematic Literature Review (SLR) and a survey that involved 181 software engineers from diverse global locations. Our findings shed light on the critical autonomy factors that significantly impact turnover in GSD projects, including recognition, communication, collaboration, trust, and task balance. By offering a comprehensive understanding of these autonomy-related elements, our research provides software companies and organizations with valuable insights to address the persistent turnover issue in GSD projects. Through a deeper comprehension of the role of autonomy in mitigating turnover, companies can implement targeted strategies to enhance team dynamics, foster better communication, and improve collaboration in distributed software development settings. The knowledge gained from this study will prove instrumental in creating a positive and productive work environment for distributed teams, ultimately leading to improved project outcomes and overall organizational success.

Keywords: Autonomy. Turnover. Distributed Software Development.

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

1.1 CONTEXT

Software development has become a pivotal driver of innovation and business growth in the modern era of globalized economies and interconnected markets. Organizations increasingly adopt a global software development (GSD) model to harness diverse talent, optimize costs, and achieve faster time-to-market. GSD allows companies to leverage expertise from different geographical locations, fostering collaboration among distributed teams. However, managing GSD teams poses unique challenges, including the dynamic interplay between autonomy, turnover, and project success (CONCHÚIR et al., 2009).

This section contextualizes the relationship between autonomy, turnover, and GSD by drawing upon relevant academic articles and research findings. We will explore the concepts of autonomy and turnover in the context of software development teams working across national boundaries and delve into their implications for project outcomes and team performance.

Autonomy refers to the decision-making freedom and independence granted to software development teams operating in the GSD model (DECI; RYAN, 2012). As teams are dispersed across various locations, they often work in different time zones and cultural settings. This can lead to variations in management practices, development methodologies, and communication styles. Several studies have highlighted the significance of autonomy in GSD teams (CONCHÚIR et al., 2009; NOLL et al., 2017; HYNNINEN; PIRI; NIINIMÄKI, 2010).

Findings from studies conducted by Fitzgerald *et al.* indicate that high levels of team autonomy in GSD can foster creativity, initiative, and ownership, enabling teams to adapt quickly to local challenges and opportunities (FITZGERALD et al., 2017). However, excessively decentralized decision-making can result in inconsistent approaches, misalignment with organizational goals, and difficulty coordinating efforts across teams (HERBSLEB; MOCKUS, 2003). Therefore, striking the right balance between autonomy and centralized control is crucial for successful GSD projects.

On the other hand, turnover or the rate at which team members join or leave a GSD project, presents a significant challenge in global software development. The distributed nature of teams can amplify the effects of turnover due to communication and coordination challenges. Employee turnover can be disruptive, leading to knowledge loss, decreased team cohesion, and increased project risk (UZOKA et al., 2011; MASSONI et al., 2019; GOPALAKRISHNAN; HALGIN;

BORGATTI, 2013).

According to Dikert *et al.* (DIKERT; PAASIVAARA; LASSENIUS, 2016), the high turnover rates in GSD teams can result in delayed project timelines, decreased productivity, and reduced software quality. Dikert *et al.* (DIKERT; PAASIVAARA; LASSENIUS, 2016) further suggests that establishing mechanisms to mitigate turnover-related issues, such as knowledge transfer protocols and enhanced onboarding processes, is essential for project success.

The relationship between autonomy, turnover, and GSD is complex and multifaceted. Autonomy can serve as a mitigating factor against the adverse effects of turnover, as empowered teams may adapt more effectively to personnel changes. On the other hand, high autonomy combined with frequent turnover can lead to inconsistent practices and fragmented knowledge, affecting overall project stability (SMITE et al., 2019).

Other researches suggest that fostering a culture of autonomy while maintaining consistent knowledge-sharing mechanisms can bolster team resilience in the face of turnover (NOLL et al., 2017; BEECHAM et al., 2008). Effective communication and knowledge management strategies become pivotal in maintaining team cohesion and project continuity (LUNDENE; MOHAGHEGHI, 2018; UZOKA et al., 2011).

Furthermore, the relationship between autonomy, turnover, and global software development is a critical area of investigation. Empirical evidence shows that striking the right balance between team autonomy and centralized control is essential for effectively leveraging GSD's benefits. Moreover, addressing the challenges posed by employee turnover through proactive knowledge management and communication strategies can significantly impact project outcomes and overall team performance. By contextualizing these factors within the domain of global software development, organizations can develop informed strategies to navigate the complexities of distributed development teams and enhance project success.

1.2 MOTIVATION

In recent years, the landscape of software development has undergone a significant transformation due to the globalization of the industry. Yet, the COVID-19 pandemic has considerably changed and fastened the way of working inside software teams worldwide and has also opened the possibilities of remote work (MARINHO et al., 2021), making software engineers able to work for whom and from anywhere they want. Therefore, Global software development (GSD) has emerged as a prominent practice, enabling organizations to tap into a diverse talent pool,

collaborate across borders, and deliver products and services to a global customer base. While GSD offers numerous advantages, it also introduces unique challenges, with employee turnover being one of the most critical issues faced by organizations operating in this context.

Turnover has been an aspect of software teams investigated in previous researches (FERREIRA; SILVA; VALENTE, 2020; ANG; SLAUGHTER, 2004; LACITY; IYER; RUDRAMUNIYAIAH, 2009; GOPALAKRISHNAN; HALGIN; BORGATTI, 2013; MASSONI et al., 2019; UZOKA et al., 2011) with some studies stating it was a bad thing to happen focusing on critical impacts for organizations (LIN; ROBLES; SEREBRENIK, 2017; SEDANO; RALPH; PÉRAIRE, 2017) and others saying the opposite (MELAND; WAAGE; SEIN, 2005) as it fosters innovative ideas and new people to come in.

Employee turnover is a multifaceted phenomenon encompassing voluntary and involuntary employee departures from an organization. In the context of GSD, turnover can result from many factors, such as cultural differences, communication barriers, and geographical distance. The cost and impact of turnover on an organization include increased recruitment expenses, loss of valuable knowledge, disrupted project continuity, and diminished team morale. Consequently, understanding and mitigating turnover has become a priority for managers and researchers alike.

On the other hand, software engineering has also witnessed extensive research addressing various aspects related to individuals and teams throughout the software development life cycle. Motivation, (BEECHAM, 2014; NOLL et al., 2017; HACKMAN; OLDHAM, 1976; BEECHAM; NOLL, 2015), happiness (MARINHO et al., 2021; GRAZIOTIN et al., 2018; AMORIM; MARINHO; SAMPAIO, 2020), communication (FAGERHOLM et al., 2015), satisfaction (FERRATT; ENNS; PRASAD, 2001; BELLINI et al., 2019) have received significant attention in the literature. Among these aspects, autonomy has emerged as a widely studied factor due to its links with motivation (CAMPION, 1988; NOLL et al., 2017; DECI; RYAN, 2012), decision-making, and freedom of choice in the workplace (BASS et al., 2018; MARINHO et al., 2021).

Amidst the complexities of GSD, autonomy has emerged as a potentially influential factor that can impact employee turnover. Autonomy refers to the degree of independence and decision-making authority granted to employees in their work processes. While autonomy has long been recognized as a crucial aspect of job satisfaction and organizational commitment in traditional collocated settings, its role and significance within the context of GSD remain relatively unexplored (CONCHÚIR et al., 2009).

Although turnover and autonomy have been widely addressed in previous studies, each one

of them were mainly linked to other factors within software teams, while their inner relationship and its impact on software development teams were not adequately addressed yet. Also, the distributed context shows itself as essential and up-to-date (MARINHO et al., 2021) to have this analysis executed. Yet, to ensure the success of global software development projects, managers are urged to choose a technically competent team, ensure effective communication and continuous customer engagement throughout the project, encourage autonomy and manage a low staff turnover in teams (VITHANA; ASIRVATHAM; JOHAR, 2018).

By delving into this topic, we aim to address the following key research objectives stated in Section 1.3. Based on that we define our research question which will drive our research during this work: *"Does the software engineer's autonomy impact turnover in GSD teams?"*

1.3 OBJECTIVES

In this study, our objective is to understand if there is a connection between software engineer's autonomy and turnover and, if so, how autonomy and turnover relationship takes place within Global Software Development teams, considering the perceived impacts of this relationship.

In order to achieve the overall objective of this work, the following specific objectives are defined:

- To provide a comprehensive overview of the state-of-the-art and related factors in autonomy and turnover in the GSD context;
- To analyse real-life scenario by capturing autonomy and turnover relationship's perception among software engineers within GSD teams;
- To identify and highlight good practices related to autonomy and turnover within GSD teams based on outcomes generated from this research.

To fulfill our first objective of this study, we conducted a Systematic Literature Review (SLR). The SLR gathered studies and results from the literature regarding autonomy, turnover, and GSD to provide a strong basis support for further analysis. Subsequently, we conducted a *survey* to better explore and expand the understanding regarding autonomy-turnover dynamics by collecting perceptions from real software engineers.

Therefore, by exploring the interplay between autonomy and turnover within distributed software teams, our study aims to contribute to a better understanding of these complex dynamics. The findings have practical implications for organizations, offering insights into the perceived autonomy factors and turnover motivators that influence software engineers in distributed settings. Ultimately, this research aims to provide valuable knowledge that can assist software companies in addressing the challenges associated with turnover in distributed software teams.

Our document is organized as follows: in Chapter 2, we introduce the background and related work. Chapter 3 describes the method applied to SLR design and execution as well as the applied survey. Chapter 4 displays the gathered results from SLR and survey participants. Then, Chapter 5 presents discussions about results and literature. Chapter 6 highlights our conclusions and possible further investigations and in Appendix A we have our Survey protocol available.

2 BACKGROUND

This chapter introduces the key concepts necessary for comprehending the subject matter addressed in this work. Additionally, we will delve into pertinent references that connect the core ideas presented in each piece and highlight any existing gaps.

2.1 AUTONOMY

Autonomy is known as one of the most studied job characteristics, and it plays a central role in motivation's work design (CAMPION, 1988). In an early stage, autonomy was seen as a synonym of how much freedom and independence a team member had while executing a project or work task (HACKMAN; OLDHAM, 1976). While modern software companies strive to increase team autonomy to enable them to successfully operate the piece of software they develop and deploy, efficient ways to orchestrate the work of multiple autonomous teams working in parallel are still poorly understood (ŠMITE et al., 2023).

After that, research (WALL; JACKSON; DAVIDS, 1992; MORGESON; HUMPHREY, 2006; BASS et al., 2018; MARINHO et al., 2021) regarding this theme made explicit that the concept of autonomy in workplaces was mentioned together with some characteristics such as freedom, independence, decision making, choices of working routine and methods to task execution.

Autonomy is strongly related to the Self-Determination theory stated by Deci and Ryan (2012), which aims to explain the factors that could mediate motivation. In this context, this theory indicate competency, identification with work, and autonomy as three psychological necessities innate to a person to feel satisfied in a workplace. Furthermore, Deci and Ryan (2012) pointed out that the higher the autonomy one feels, the more motivated one is.

Yet, autonomy has been identified as an important motivational factor for software engineers who work in distributed models. Those professionals exhibit greater tolerance for motivational changes because they are naturally exposed to hours of antisocial work and travel necessities (IVARSSON; GORSCHEK, 2011).

Also, according to Šmite et al. (2023) autonomy does not mean anarchy, or unlimited permissiveness. Instead, autonomous teams are expected to take responsibility for their work and coordinate, communicate and align their actions with others, comply with a few enabling constraints, take many decisions independently without management control or due to collec-

tive efforts that bypass formal boundary structures. Yet, autonomy is also related to effective sharing of the code base, achieving alignment, networking and knowledge sharing.

Yet, team autonomy has a range of implications and is challenged by a number of factors, such as knowledge complexity and decision-making, learning, large-scale problems, product and technical dependencies, the use of platforms, virtual collaboration and diversity which makes alignment with stakeholders and external teams particularly necessary in multi-team environments with many technical dependencies. Therefore, teams in complex environments increasingly need to regulate and manage their work in cooperation with internal and external partners and systems (RAVN et al., 2022).

Additionally, according to Noll et al. (2017) autonomy at work was recognized as a central factor in work design, leading to many positive outcomes including motivation and satisfaction while professional demotivation is considered a predictive factor towards turnover, which could be differently expressed in terms of distributed projects (CRUZES; DYBA, 2011).

Nevertheless, autonomy could be defined from different perspectives, such as individual autonomy, internal and external autonomy, where external autonomy is defined as the influence of management and other individuals outside the team on the team's activities. Internal autonomy refers to the degree to which all team members jointly share decision authority, while individual autonomy refers to the freedom and discretion an individual has in carrying out assigned tasks (MOE et al., 2021).

2.2 TURNOVER

Team members turnover negatively impacts software development. Even though many authors consider turnover a natural part of an organization, high levels usually negatively impact processes efficiency (BASS et al., 2018).

Turnover may be classified as external when team members leave the organization or internal when they remain in the company but change their previous work. This process can happen voluntarily when the employee voluntarily decides to abandon the company and their role or involuntarily, which happens when the organization decides to terminate its relationship with the employee (CHATZIPETROU; ŠMITE; SOLINGEN, 2018).

Furthermore, turnover rates can be calculated by dividing the number of employees who left the organization in a determined time by the total amount of employees during this same period. In daily organizational practice, a need for more detailed data arises to pinpoint the

specific causes behind turnover. This lack of understanding is recognized as a significant barrier to organizations as such data are highly relevant for strategy creation that might be applied in an efficient way to deal with turnover and its consequences properly (CHATZIPETROU; ŠMITE; SOLINGEN, 2018).

Previous research demonstrates that perceived job autonomy was somehow associated with turnover, but only in cases involving employees with high levels of perceived support as the greater the organizational support received, the lower the intention to leave the company. In companies where employees had a higher degree of organizational commitment, their intention to leave the company decreased to an even greater extent (SUÁREZ-ALBANCHEZ et al., 2022). However, these results must be integrated with new ways of working, such as the Global Software Development (GSD) (DYSVIK; KUVAAS, 2013), which has become a common practice worldwide, intensified by financial, structural and global factors (BASS et al., 2018).

Among several characteristics related to GSD is the organization's potential to attract qualified professionals, even being in a local context with professionals' scarcity. This incredible capacity to attract and hire specialized professionals positively impacts productivity and services, and products' quality (DYSVIK; KUVAAS, 2013).

GSD projects might have turnover levels raised because of physical distance, while turnover is mentioned as one of the five most significant risks for GSD projects by Ebert, Kuhrmann and Prikladnicki (2016). Yet, previous short research related to the relevance of motivational factors to prevent this phenomenon in distributed projects (DECI; RYAN, 2012).

2.3 GLOBAL SOFTWARE DEVELOPMENT - GSD

A global project is a group of people distributed in different locations that work united in a single project for an extended period. This kind of software project in which the human resources involved are spread by distance, regionally, nationally or globally, is defined as Global Software Development (GSD) (HERBSLEB; MOITRA, 2001; SULAYMAN et al., 2012; MARINHO; NOLL; BEECHAM, 2018; MARINHO et al., 2019).

Furthermore, GSD can be classified according to two factors: the distance among work teams, which presents itself as Onshore (teams located in the same country) and Offshore (teams located in different countries). Besides, on the other hand, the control relationship the central organization has over the remote teams presents itself as Outsourcing (hiring of a third-party company) or Insourcing (creation of a company remote unit) (RICHARDSON et al.,

2010).

Research indicates that offshore outsourcing service providers could reduce staff member turnover by improving work-life balance and adopting more family-friendly employment policies, such as increased autonomy. Further, outsourcing service providers could reward innovation more effectively and structure contracts to enable software product ownership to improve staff retention (BASS et al., 2018; MARINHO; NOLL; BEECHAM, 2018).

According to Herbsleb (2023), GSD has improved along time and so did the tools used by teams in such conditions. The awareness of cultural differences is widespread, and developments such as the foregrounding of open source have all enhanced the ability to work across geographic divides. However, better ways of incorporating ethics into GSD team's development practices should be investigated. Also, paying far more attention to harmful unintended consequences should also be considered. Moreover, it is necessary to carry out in-depth research into the aspects of working in distributed teams that may have a negative impact on developers' levels of motivation and stress, along with what could be beneficial in order to improve levels of motivation and decrease levels of stress (SUÁREZ; VIZCAÍNO, 2023).

2.4 RELATED WORK

Recently, Gustavsson, Berntzen and Stray (2022) stated that in cases where autonomous teams need to coordinate toward a common goal they must sacrifice some level of autonomy. Therefore, they conducted 28 interviews and 17 on-site visits as part of a multiple case study to explore how team autonomy changed in three agile software development organizations that implemented the Scaled Agile Framework (SAFe). As results, identified positive changes to team autonomy included getting a better overview, making better long-term decisions, giving and receiving help, and signaling limitations. However, the identified negative impacts on team autonomy were limited feature choice and enforced refinement. Yet, a lack of a system for team support and reduced external autonomy (which is defined by Hoegl and Parboteeah (2006) as the influence of management and other individuals (outside the team) on the team's activities) were found to have a crucial impact on that. Their work is aligned with Mikalsen et al. (2019) modern socio-technical theory study that states the effective functioning of autonomous teams is challenged by the need to coordinate and align work with multiple experts and other units in complex organizations leading to a result that a production structure with many dependencies causes challenges and a misaligned control structure is time-consuming and reduces team

autonomy (MIKALSEN et al., 2019).

However, Jr and You (2018) studied the autonomy relationship towards satisfaction within virtual teams where the goal was how to determine an easier increase of satisfaction levels through shared leadership, individual trust, and autonomy. The study was conducted with 163 individuals in 44 virtual teams, and results indicate shared leadership facilitating satisfaction through the promotion of trust and autonomy being related to higher levels of satisfaction which is a strong predictor of virtual performance among team members.

Noll et al. (2017) observed and interviewed fifteen participants from two distributed software development teams in order to highlight the relationship between autonomy and motivation where participants from the outsourcing firm reported issues such as anti-social working hours and persistent overtime that could negatively affect motivation, while the other participants reported positive experiences in areas such as work-life balance, technical challenges, and team connectedness. They also found that there was a significant difference in the motivation levels of experienced team members, which were lower than less experienced members. Furthermore, highly motivated developers are more likely to remain in their current jobs, while lack of motivation may result in attrition and turnover (BEECHAM et al., 2008; VERNER et al., 2014). However, autonomy was also perceived to be linked to stress if the necessary competence is missing while lack of connectedness can lead to a feeling of isolation (RYAN; DECI, 2000).

Nevertheless, Sethi, King and Quick (2004) study has already addressed this stress and turnover relationship within the work environment by interviewing and questioning software engineers to assess which were the main stressors and their outcomes. Their results show stress is a key factor impacting staff turnover and that it must be addressed properly in software teams.

Yet, Agarwal and Ferratt (2000) executed a study with 130 IT professionals to develop and investigate an exploratory model of those professionals joining and staying/leaving behaviors. In their work, turnover or retention would be defined by a set of career motives such as "Preferred Length of Employment", "Career stage", and "Career anchor". Then, three hypotheses were raised based on each motive, with the first one being "The preferred length of employment of IT professionals is negatively related to their turnover intentions.", the second one "An IT professional's career stage moderates the relationship between the preferred length of employment and turnover intentions." and the third one "An IT professional's career anchor moderates the relationship between the preferred length of employment and turnover

intentions." Their results confirm the mentioned hypothesis and it also illustrates the importance of investigating and discussing software engineers turnover and retention predictors and consequences.

Back then, organizations seeking lower costs and access to skilled resources began to experiment with remotely located software development facilities. This change profoundly impacted not only on marketing and distribution but also on the way products are conceived, designed, constructed, tested, and delivered to customers. The number of organizations distributing their software development processes worldwide kept increasing over the years which has made software development teams a multi-site, multicultural and globally distributed environment. More recently, attention has turned toward trying to understand the factors that enable multinationals and virtual corporations to operate successfully across geographic and cultural boundaries. Based on these factors, lessons learned from case studies regarding the adoption of GSD teams collocated and dispersed in two software development units from multinational organizations located in Brazil were presented by Prikladnicki, Audy and Evaristo (2003).

Later, Bass et al. (2018) conducted a mixed-method approach case study with 62 software engineers from three global companies to address causal factors linked to turnover and retention in both in-house offshoring and offshore outsourcing companies. Among several appointed factors practitioners cited employment policies, work-life balance, workplace innovation, product quality, alignment of offshore work hours with onshore, long working hours, and adverse impact on health as those affecting staff retention. In-house Offshore outsourcing service providers could improve development team member retention by improving work-life balance and adopting family-friendly employment policies. Moreover, they were also able to find an established connection between turnover and motivation as high levels of motivation can have a positive effect on staff retention which corroborates Hall et al. (2008) findings. Finally, they stated employment policies and employer commitment to work-life balance are designed to nurture and retain good quality staff members for internal product development, while outsourcing service providers could reward innovation more effectively and structure contracts to enable software product ownership to improve staff retention.

Chatzipetrou, Šmite and Solingen (2018) stated turnover in GSD is an extremely important issue to be discussed and focused their research on the relationship between software engineer's experience levels and staff retention throughout a case study within an offshore company. Their results were driven by two main questions: "Does the experience of an employee in the company influence their decision to stay or leave their job?" and "Is there a threshold, which determines

the likelihood of the employees leaving their job?" and they indicate a positive influence related to the first question while the threshold for an employee to leave their job are likely 2 years with 90% of participants leaving the company still within the first year.

Hynninen, Piri and Niinimäki (2010) conducted three case studies (each one regarding a distributed project) among 12 globally distributed ones. In their hypothesis, voluntary turnover at off-site seemed to be related to for example the levels of work autonomy, challenges provided by the work, and job security at off-site. Therefore, its purpose was to address the following questions: 1) How was the level of voluntary turnover in the studied projects described? 2) How did the projects differ in regard to the suggested antecedents of affective commitment? The study was conducted through interviews and the results revealed a high turnover rate in off-site settings due to low commitment towards the project. They also observed a significant relationship between the level of commitment and the project's structure and practices. Enhancing off-site commitment and reducing voluntary turnover can be achieved by structuring the project in a more inclusive manner, enabling off-site team members to have more influence over their work. Additionally, fostering closer collaboration between on-site and off-site employees was found to be beneficial.

Massoni et al. (2019) has addressed other characteristics related to turnover such as satisfaction and work exhaustion. In their quantitative study, 76 Brazilian software engineers who have left their jobs voluntarily were presented with a survey to assess the reasons for that. As result, low to moderate autonomy and satisfaction were identified by the participants as reasons for leaving.

Gopalakrishnan, Halgin and Borgatti (2013) conducted a long 5-year study to analyze complete project affiliation data linking 728 geographically distributed employees at a multinational high technology firm across five years to test whether employees who were leaving the project would influence those who remained to follow the same path. During the study execution, 183 left the company and their findings suggest that project affiliation networks in such settings occasion social comparisons among employees and serve as conduits for the diffusion of their career mobility decisions. Yet, they discuss and claim the management of voluntary turnover in knowledge-intensive distributed organizations.

Finally, our work differs from the ones presented above as it was conducted through a Systematic Literature Review plus a survey focused on understanding the relationship and impact between autonomy and turnover by collecting different perceptions from 181 software engineers from distributed teams and their members dispersed into different dimension levels

such as software engineers working in teams distributed collocated (using virtual tools but in the same geographical city or region), within the same country (team members dispersed through several cities within the same country), within the same continent (team members dispersed through countries within the same continent) and globally (team members from different countries and continents working together).

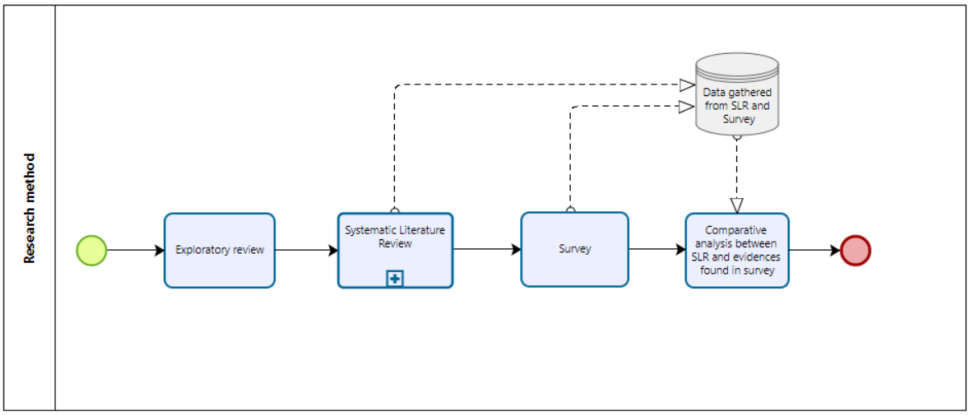
2.5 CLOSING REMARKS

In this chapter has laid the foundation for understanding the key elements crucial to the subject matter of this work. We've explored the concept of autonomy, its significance in motivation and work design, and its relevance in software engineering, particularly in distributed projects. The discussion also encompassed the critical factor of internal and external turnover, its consequences on software development, and its specific implications within the realm of Global Software Development (GSD). Additionally, we've delved into related studies, which have provided valuable insights into the factors influencing turnover, autonomy, and their interconnections. With this background, we are now prepared to explore the specific challenges and potential solutions within the landscape of software engineering, autonomy, and turnover.

3 RESEARCH METHOD

This section will unveil the methodology process used in this entire research. Figure 1 shows our research's general process.

Figure 1 – Research Method



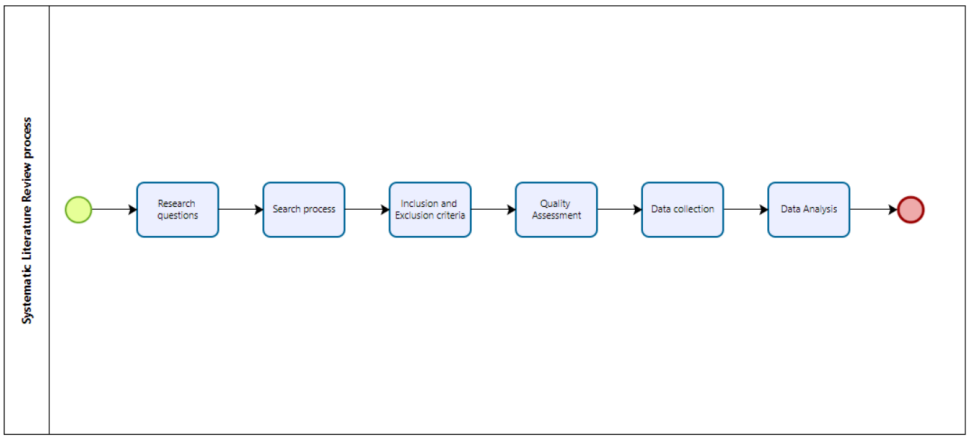
Source: Author

Moreover, a more comprehensive explanation of the steps taken, including the phases referenced in Figure 1, such as the Systematic Literature Review (SLR) and its associated questions, answers, and research methodology. Additionally, the details of the survey that was created and administered to 181 participants, including the design process, will be provided. These results will be discussed in comparison to the SLR findings in the upcoming chapter 5.

3.1 SYSTEMATIC LITERATURE REVIEW

We conducted a Systematic Literature Review (SLR) to answer the research question: *What is the relationship between autonomy and turnover in distributed software projects?* We followed Kitchenham and Charters' guidelines (KITCHENHAM; CHARTERS, 2007), and it had its execution based on a research protocol (See appendix 9) that contains our research questions, inclusion and exclusion criteria for gathered studies, search string, chosen digital libraries, and a quality evaluation guideline. Figure 2 presents the steps and the next subsections detail each step.

Figure 2 – Systematic Literature Review process based on Kitchenham



Source: Author

3.1.1 Research questions

We sought to answer the following research question: *RQ1: How is autonomy related to turnover in GSD teams?* and to this, we elaborated the following research questions:

1. RQ1.1 - What factors contribute to autonomy in GSD teams?
2. RQ1.2 - What factors contribute to turnover in GSD teams?
3. RQ1.3 - What factors mitigate turnover in GSD teams?
4. RQ1.4 - What impact does autonomy have on turnover within GSD teams?

3.1.2 Document selection

We used the boolean search string shown bellow to ensure that we captured a wide variety of papers.

("global software engineering" OR "global software development" OR "distributed software engineering" OR "distributed software development" OR GSE OR GSD OR "distributed teams" OR "global team" OR "dispersed team" OR "spread team" OR "virtual team" OR offshore OR outsource OR DSD OR DSE) AND (turnover OR "turnover intention" OR departure OR "rate of replacement" OR "employee retention") AND (autonomy OR "self-government" OR independence OR "self-rule" OR freedom OR "self-sufficiency" OR "job control" OR "schedule control" OR "self-management" OR isolation OR "job autonomy")

3.1.2.1 Inclusion/Exclusion criteria

The following criteria guided the selection of papers that helped us address the research questions. We *included*:

- (i) Complete, peer-reviewed, published papers;
- (ii) Papers directly related to the research questions;
- (iii) The study is available via the university library services accessible to the authors during the time of the research.

We *excluded*:

- (i) Texts not published in English;
- (ii) Technical content, e.g: editorials, tutorials, key-note speech, white papers, thesis, dissertations, technical reports, books;
- (iii) Studies not related to Software Engineering.

Our search string returned 1752 studies from 5 engines, as presented in Table 1. We identified 95 duplicate papers and no replicates. After excluding duplicate results from the dataset, we identified papers for inclusion in the initial selection (phase 1). Of these papers, 85 were passed on to phase 2, in which 25 were eliminated and 60 (<https://bit.ly/3wWUV0Q>) were finally passed on to the data extraction and data synthesis phase (see Table 1).

Engine	Selection	Phase 1	Phase 2
ACM	759	40	33
Scopus	148	27	20
Wiley	175	5	3
Springer	353	3	2
IEEE	317	10	2
Total	**1752**	**85**	**60**

Table 1 – Papers by engine.

3.1.3 Study Quality

The quality of the data extraction phase significantly impacts the overall integrity of the systematic literature review. Transparent and well-documented extraction processes enhance the review's transparency and reproducibility, allowing other researchers to assess the rigor of the methodology employed. To ensure the accuracy of extracted data, cross-checking by multiple reviewers, as well as resolving discrepancies through consensus or third-party arbitration, is a recommended practice.

By systematically and rigorously collecting relevant information from selected studies, researchers can build a comprehensive and evidence-based understanding of the research field, ultimately contributing to the advancement of knowledge and informed decision-making.

Our study assessed papers quality criteria based on Dyba, Dingsoyr and Hanssen (2007) principles and good practices regarding empirical research process in software engineering. We answered the following questions for each selected study using 1 for *Yes*, 0.5 for *Partially* and 0 for *No*. I. Is there a clear definition of the study objectives? II. Is there a clear definition of the justifications of the study? III. Is there a theoretical background about the topics of the study? IV. Is there a clear definition of the research question (RQ) and/or the hypothesis of the study? V. Is there an adequate description of the context in which the research was carried out? VI. Are used and described appropriate data collection methods? VII. Is there an adequate description of the sample used and the methods for identifying and recruiting the sample? VIII. Is there an adequate description of the methods used to analyze data and appropriate methods for ensuring the data analysis were grounded in the data? IX. Is provided by the study clearly answer or justification about RQ / hypothesis? X. Is provided by the study clearly stated findings with credible results? XI. Is provided by the study justified conclusions? XII. Is provided by the study discussion about validity threats?

3.1.4 Data Extraction and Analysis

The data extraction phase consists on a pivotal component of the Systematic Literature Review process, contributing significantly to the rigor and credibility of the review's findings. This phase involves the meticulous and systematic collection of relevant information from selected studies, enabling researchers to synthesize and analyze the cumulative knowledge in a specific field or research question. In this subsection, we assess essential aspects of the data extraction phase within the executed Systematic Literature Review, exploring its purpose, methods, challenges, and implications. Therefore, the data extraction phase within a systematic literature review is a crucial step that lays the groundwork for subsequent analyses and synthesis.

A spreadsheet was used to record the extracted data. Two researchers performed the data extraction independently to reduce the bias of the data extraction. Before the formal data extraction process, two researchers discussed the definitions of the data, in pairs, items to be extracted to ensure that both researchers had a common understanding. After we completed the data extraction, a discussion was held to resolve conflicts for reaching a consensus on the data extraction results.

We opted for a mixed approach based on quasi schemes (standard) where the research type of facet was selected as described by (WIERINGA et al., 2006). The rigour and relevance model used was the one proposed by (IVARSSON; GORSCHEK, 2011) in order to establish a qualified point of view regarding SLR's found paper's relevance.

Based on (IVARSSON; GORSCHEK, 2011) we assessed the rigour and relevance studies. The Rigor consists of combining three concepts: context, study design and discussed validity. They received a score where 0 indicated weakly, 0.5 medium, and 1 strong. The most of the papers were classified as strong on the three concepts mentioned before (25).

Furthermore, we calculated the relevance amount in order to analyze papers relevance which consisted of combining four classifications: Relevance subject, context, scale and method and each one of them were scored with 0 (weak), 0.5 (medium) and 1 (strong). As a result, most papers (38) were classified with a score of 4, which meant strong relevance. To measure the maturity of the themes, a scheme proposed by (EBERT; KUHRMANN; PRIKLADNICKI, 2016) was carried out, where the results found in research facets and contribution facets over time are crossed.

Yet, according to Strauss A Corbin (2008), the data analysis procedure involves coding

phases. During those phases it is suggested to identify codes and categories for the collected quotes from papers. Then an open coding phase takes place where all collected material is transcribed, analysed and keywords are selected. Moreover, identified concepts are grouped in categories and theirs characteristics and dimensions are identified (COLEMAN GERRY O'CONNOR, 2007).

Therefore, after executing our Systematic Literature review, a thorough reading was done to extract as many quotes that somehow answered our questions as possible. All data were categorized in a digital sheet, using Google Sheets tool, which contained the following fields: study's sequence ID, study's quotes ID, related research questions, quote's field itself with each extracted quote and yet the reason why each one of them was chosen. In the end, it was possible to identify 161 quotes that answered the proposed research questions. They were all exported to MaxQDA 2020 software (version 20.2.1) for proper codification.

In order to codify and to attribute themes for some groups of quotes, we followed Cruzes and Dyba (CRUZES; DYBA, 2011). The initial papers' codification was executed by building a list of codes with 388 categorizations. Then, codes were grouped into themes that resulted in 5 main groups: organizational culture, Autonomy and Global Software Development (GSD) relationship, Autonomy and Turnover relationship, Autonomy dimensions in GSD, and Factors related to Autonomy in the event of Turnover.

3.2 SURVEY

The basic idea of survey methodology is to collect information from a group of people by sampling individuals from a large population. Examples of surveys are found in daily life in several situations, such as election polls, market surveys, etc. and there is a large amount of literature on the general methodology (LINAKER et al., 2015). According to Linaker et al. (2015) a survey should have the following phases in order to be well conducted:

1. Defining research goals
2. Identifying target audience
3. Designing sampling plan
4. Designing survey instrument
5. Evaluating survey instruments

6. Analyzing survey data
7. Drawing conclusions
8. Documenting and reporting

Therefore, this survey based its assumptions and questions on results gathered by previously executed SLR and aimed the validation and assess the perception of turnover, autonomy, and their relationship within GSD teams.

3.2.1 Study Design

In this study, we used a web survey hosted on Google Forms tool to gather data from participants. This survey was divided into 4 subsections (See Appendix 7). First, three basic questions were displayed aiming at an evaluation of a certain level of participants and their global teams's autonomy. All of them were stated using the 5-point Likert scale where 1 means 'Strongly disagree' and 5 means "Totally agree'. In the second section, there was a goal to identify turnover-related results. Therefore, for this section, it was possible to have questions more focused on turnover intentions and what could possibly improve inside GSD teams as well as questions focused on impacts generated by turnover in GSD teams. Furthermore, this section was designed with open questions and classification ones. In the third section, there were open questions and affirmative statements displayed using the 5-point Likert scale such as in the first section. The aim of those questions and statements evaluated by participants was to check the relationship between autonomy and turnover/turnover intentions for further analysis. Ultimately, the study's demography perspective was addressed with questions intended to collect participant's gender, age, work experience, education, role, team distribution, and company size.

3.2.2 Setting

This research aimed at participants who work in global software development teams to gather data from them about turnover and autonomy inside their companies. Focusing on the research's quality, readability, and validity, the developed survey went through a pilot session with three software engineers who helped on addressing failures and small issues on statements

and questions before it takes place. Moreover, some web platforms were used to have the survey sent to participants, such as Linkedin, Twitter, Whatsapp, e-mail, Instagram. It was applied during a short period (from August 20th to September 30th) due to deadlines approach and even with the short period it was possible to have 181 participants (already excluding the 3 ones from pilot phase).

3.2.3 Participants

Out of the 181 participants mentioned earlier, the majority were comprised of software developers, testers, scrum masters, project managers, team leaders, technology leaders, security analysts, UX/UI designers, software architects, data scientists, and so on. Each of them had at least one team member working in a distributed environment.

3.2.4 Data Analysis Procedure

Qualitative content analysis (QICA) is an inductive research approach based on naturalist/humanist research orientation in which research questions guide data gathering and analysis while potential themes and other issues may arise through data reading as it is possible to generate broader generalizations and theories from specific observations aiming the capture of meanings, emphasis and themes of messages, also understanding the organizations and process on how they are presented. It searches for multiple interpretations by considering the diversity represented by aspects such as ideological positions, critiques or the diverse use of the texts examined (WHITE; MARSH, 2006). It is also known to follow a subjective approach combined into categories that support interpretations, diversity of ideas, and perspectives (WHITE; MARSH, 2006).

Therefore, for the data analysis procedure, the author chose a qualitative approach where all collected data was extensively analyzed generating perceptions stored in a spreadsheet which was later on translated into charts, tables, and quotes constructing parallelisms by engaging in triangulation to support arguments and conclusions for this study (WESTER, 2005).

3.3 CLOSING REMARKS

This Chapter discussed the Research Method employed for the research study. We provided a comprehensive explanation of the methodology employed in the research, encompassing both the systematic literature review and the survey. This methodological approach aimed to address the research questions effectively and provide valuable insights into the relationship between autonomy and turnover in GDS teams. Primarily focused on the "Systematic Literature Review" (SLR) and the subsequent steps. The research questions were outlined, categorized into sub-questions, and clearly defined. The document selection process was described, mentioning the search string and inclusion/exclusion criteria. The data extraction and analysis process was explained, emphasizing the use of coding to categorize and analyze quotes extracted from the literature. The survey section was introduced, explaining its purpose and design. The survey aimed to validate and assess perceptions related to autonomy and turnover in GSD teams. The qualitative content analysis approach was chosen to analyze the collected survey data, facilitating the exploration of themes, perspectives, and meanings within the responses. All results gathered from SLR and survey questions will be properly analysed and presented in Chapter 4.

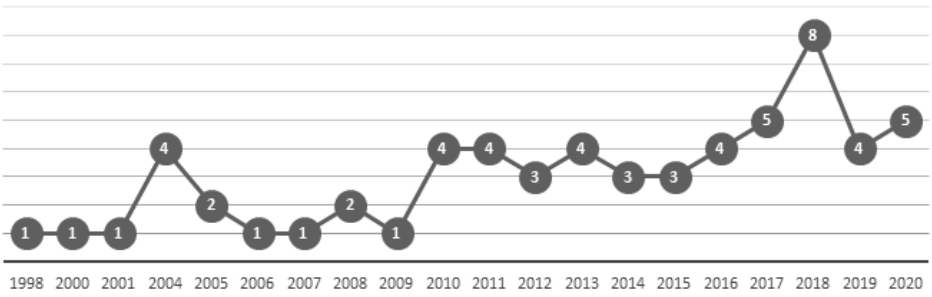
4 RESULTS

In this chapter, we present the analysis of the results obtained from the comprehensive study focused on exploring the intricate interplay between autonomy, turnover, and Global Software Development (GSD) within the context of modern organizational structures. The chapter begins with an overview of the research questions addressed and the methodology employed. Subsequently, the findings are organized and discussed, shedding light on the implications for both academia and industry.

4.1 SYSTEMATIC LITERATURE REVIEW RESULTS

We executed an overview of all papers found. The first year the theme was found was in 1998, and it reached its apex in 2018 with eight papers found. It was noticed that after 2010, autonomy related to Turnover in GSD projects had a constant recurrence on literature with an average of 4 papers per year as seen in Figure 3.

Figure 3 – Papers distribution by year

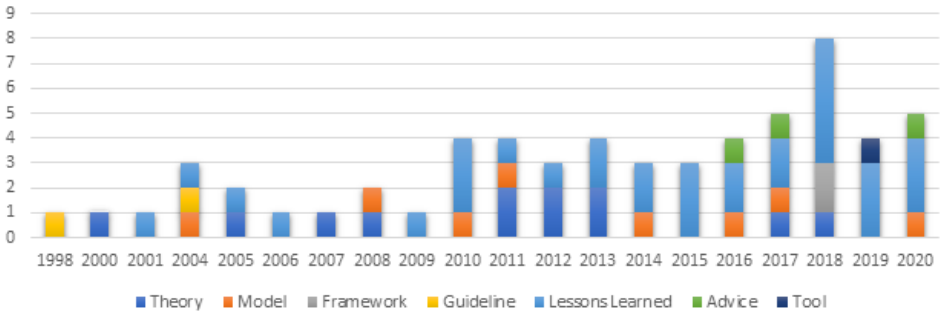


Source: Author

The research type facets according to Wieringa et al. (2006) is presented by year in Figure 4 where most of the analyzed papers were classified as Lessons Learned(32), Theory (12) and Model (8). Also, it was possible to observe that since 2010 there was a theory and Lessons Learned tendency seen. It is also relevant to state the increase of Framework facets types since 2016.

The distribution of contribution type facets of the reviewed studies derived from Petersen et al. (2008) is presented in Figure 5 by year. Most papers were classified as Evaluation (32) followed by papers of Experience (17) and Solution Proposal (5) in the context of research

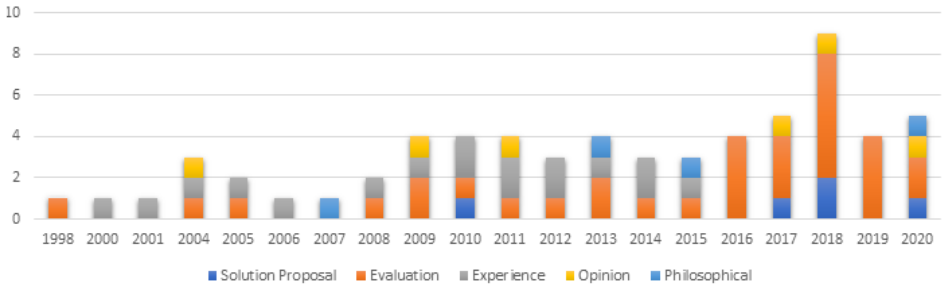
Figure 4 – Research type facets over time



Source: Author

type facets classification regarding contribution. It was noticeable an increase of Evaluation type tendency over time through found papers contribution analysis.

Figure 5 – Contribution type facets over time



Source: Author

Based on (IVARSSON; GORSCHEK, 2011) we assessed the rigour and relevance studies. The Rigor consists of combining three concepts: context, study design and discussed validity. They received a score where 0 indicated weakly, 0.5 medium, and 1 strong. The most of the papers were classified as strong on the three concepts mentioned before (25).

Furthermore, we calculated the relevance amount in order to analyze papers relevance which consisted of combining four classifications: Relevance subject, context, scale and method and each one of them were scored with 0 (weak), 0.5 (medium) and 1 (strong). As a result, most papers (38) were classified with a score of 4, which meant strong relevance. Both rigor and relevance analysis are illustrated in Figure 6.

In order to analyze the different contributions' maturity published about autonomy impact on turnover in distributed projects, Figure 7 provides a systematic map to illustrate maturity achieved by the gathered papers. The map shows a focus on collecting lessons learned to guide the knowledge about the theme. Lessons learned (33) represented the most significant part

Figure 6 – Papers Rigor and Relevance

RELEVANCE						
0,5			1			
1	2					
1,5			1	2		
2			1		2	
2,5				1		
3		1	2		1	3
3,5		1			2	2
4	1	3	2	8	2	20
RIGOR	0,5	1	1,5	2	2,5	3

Source: Author

among results and, from them, 19 corresponded to Evaluation papers. This map also shows a kind of maturity in terms of theory based on experiences and evaluation models.

Figure 7 – Systematic Map

Experience Paper	2	7	0	0	8	0	0
Evaluation Paper	6	3	1	2	19	1	0
Philosophical Paper	0	1	0	0	3	0	0
Solution Proposion	0	1	3	0	1	0	0
Opinion Paper	0	0	0	0	2	0	0
-	Model	Theory	Framework	Guideline	Lessons Learned	Advice	Tool

Source: Author

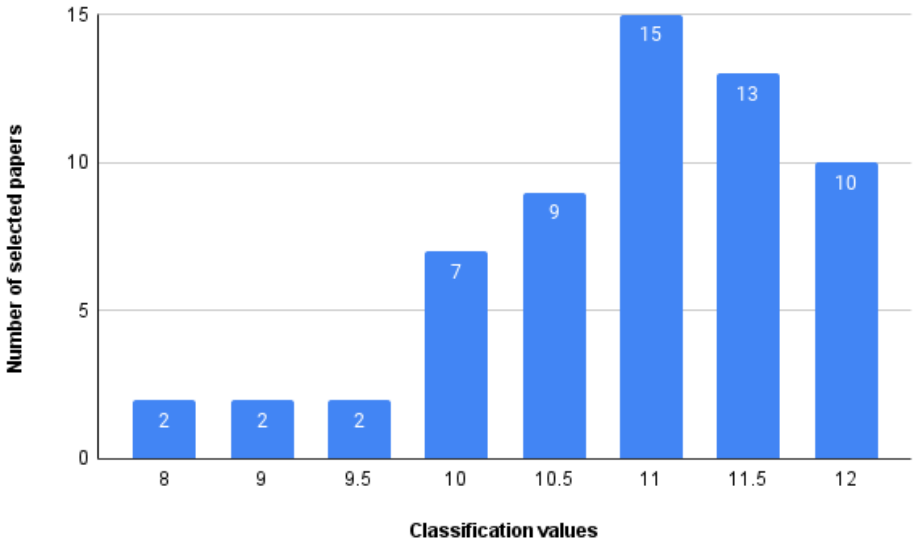
Regarding our quality assessment questions presented in subsection 3.1.3 we were able to classify our selected papers based on Dyba, Dingsoyr and Hanssen (2007) principles. Most of the classified studies(54) were considered as highly qualified based on sum of criteria values (from 10 to 12) while only 6 of them were classified between 8 and 10 (See Figure 8).

4.1.1 RQ1.1 - What factors contribute to autonomy in GSD teams?

During the RSL's categorization phase, it was possible to identify more than a simple relationship between autonomy and turnover. A really interesting category found was the one related to factors that might predict a certain level of autonomy from software engineers (SEs) inside a distributed software development team.

Autonomy is usually not a problem when working remotely; a prerequisite for remote working is the ability to work independently. However, individuals can be undermined if head

Figure 8 – Quality assessment classification



Source: Author

office is heavy-handed, and interferes with communication, say with on-site customers, or if their work is monitored too stringently. For developers working under the customer's spotlight, autonomy can be problematic (BEECHAM, 2014).

Furthermore, this *high level of monitoring needs to be decreased* as individuals driven by a desire for autonomy derive satisfaction by disassociating themselves with employment and setting up their own business (MGAYA et al., 2009).

Yet, among all identified factors that may lead to a software engineer's (SE) autonomy, it is possible to cite *redundancy of skills* since it affects the team's capability to adapt to changing situations and *cross-functionality* as team members are able to get less control of the scheduling and implementation of their own tasks and to share more activities which consequently fosters them to share information so everyone has the knowledge to influence decisions. These factors also light up the *need for a shared purpose* and *mutual trust among team members* (JR; YOU, 2018; LUNDENE; MOHAGHEGHI, 2018; BEECHAM, 2014) which are other essential points on generating this desired autonomy as well as a *good management support* in order to create the right environment for the teams. (LUNDENE; MOHAGHEGHI, 2018; BASS et al., 2018; AGARWAL; FERRATT, 2000)

Once this management support is facilitated taking place in an effective way, not only the software engineer's autonomy is encouraged but it also supports its relationship and dependency on connectedness among peers through collaboration, a culture of learning and valuing

individuals in the organization as well as fostering an environment of trust and flexibility in business processes (TAFTI; MITHAS; KRISHNAN, 2007).

It's also valuable to mention that this explicit attention provided by leaders and managers to the needs of software engineers at different stages of their careers can help decrease the incidence of high turnover (FERRATT; ENNS; PRASAD, 2001) as the recognition that comes from the leaders and stakeholders for the members of their respective teams is an essential factor linked to happiness and consequent satisfaction (AMORIM; MARINHO; SAMPAIO, 2020).

Table 2 – Autonomy's contributing factors by references

Related Factors	References	Occurrences
Work Regime	(BASS et al., 2018), (NOLL et al., 2017), (UZOKA et al., 2011), (FOERDERER et al., 2016), (JOHRI; TEO, 2018).	5
Management/Leadership competence and support	(BASS et al., 2018),(AGARWAL; FERRATT, 2000) .	4
Career Satisfaction	(BASS et al., 2018), (SMITH; SPEIGHT, 2006), (GARRISON et al., 2010), (UZOKA et al., 2011).	4
Mutual trust among team members	(LUNDENE; MOHAGHEGHI, 2018),(JR; YOU, 2018),(BEECHAM, 2014).	3
Job opportunities	(UZOKA et al., 2011),(JOHRI; TEO, 2018), (MGAYA et al., 2009).	3
Work Routine	(SETOR; JOSEPH, 2019), (LUNDENE; MOHAGHEGHI, 2018), (MONTEIRO et al., 2011)	3
Decrease monitoring levels	(BEECHAM, 2014), (MGAYA et al., 2009).	2
Cross-functionality	(LUNDENE; MOHAGHEGHI, 2018)	1

4.1.2 RQ1.2 - What factors contribute to turnover in GSD teams?

Certain organizations may not deliberately focus on retaining their IT workers, such as those with a strict task-focused organizational mindset (BELLINI et al., 2019). However, the absence of ballancing collaboration and individualism regarding the assigned tasks for developers presents itself as a key factor for increasing the chances of software engineers leaving the project, i.e. finding the balance between writing new code and maintenance tasks or even between dealing with documentations and having hands-on coding might increase the odds for an SE staying longer in the project or organization (LIN; ROBLES; SEREBRENIK, 2017).

Stress was pointed as an important factor as it is becoming increasingly clear that steps must be taken to address the problem of high stress because of its effect on employee productivity and turnover, stress among information system (IS) professionals is long recognized as a key factor affecting IS productivity and turnover and leading to substantial associated costs (SETHI; KING; QUICK, 2004). However, previous research cited the effects of exposure

Table 3 – Turnover motivators related factors by references

Related Factors	References	Occurrences
Lack of satisfaction	(BASS et al., 2018),(SMITH; SPEIGHT, 2006), (GARRISON et al., 2010), (UZOKA et al., 2011), (LIN; ROBLES; SEREBRENIK, 2017), (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013), (MONTEIRO et al., 2011), (ADYA; COTTON, 2012), (FOERDERER et al., 2016), (SUMNER; YAGER; FRANKE, 2005), (GALLIVAN, 2004), (MGAYA et al., 2009), (YENER; ARSLAN; KILINÇ, 2020), (JR; YOU, 2018), (REMUS et al., 2016), (CAMPBELL et al., 2013), (LACITY; IYER; RUDRAMUNIYAIAH, 2009), (STAPLES; WEBSTER, 2008) .	18
Autonomy	(JAIN; SUMAN, 2018), (MOURMANT; NIEDERMAN; KALIKA, 2013), (LUNDENE; MOHAGHEGHI, 2018), (MGAYA et al., 2009), (SETOR; JOSEPH, 2019), (JR; YOU, 2018), (FOERDERER et al., 2016), (MASSONI et al., 2019)	8
Lack of motivation	(BASS et al., 2018), (NOLL et al., 2017), (SMITH; SPEIGHT, 2006),(MONTEIRO et al., 2011), (MOQUIN; RIEMENSCHNEIDER, 2013), (STAPLES; WEBSTER, 2008), (BEECHAM; NOLL, 2015), (BEECHAM, 2014)	8
Lack of supervisor and management support	(NOLL et al., 2017), (UZOKA et al., 2011), (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013), (LUNDENE; MOHAGHEGHI, 2018), (QUAN; CHA, 2010), (HYNNINEN; PIRI; NIINIMÄKI, 2010)	6
Stress	(BASS et al., 2018), (UZOKA et al., 2011), (SETHI; KING; QUICK, 2004),(YENER; ARSLAN; KILINÇ, 2020),(HYNNINEN; PIRI; NIINIMÄKI, 2010).	5
Payment	(BASS et al., 2018), (UZOKA et al., 2011), (BAO et al., 2017), (HYNNINEN; PIRI; NIINIMÄKI, 2010).	4
Mismatch between expectations and reality	(SMITH; SPEIGHT, 2006), (BAO et al., 2017), (ZHOU et al., 2014)	3
Lack of connection with co-workers	(BASS et al., 2018), (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013), (TAFTI; MITHAS; KRISHNAN, 2007)	3
Lack of collaboration among the team	(BASS et al., 2018), (LIN; ROBLES; SEREBRENIK, 2017), (TAFTI; MITHAS; KRISHNAN, 2007)	3
Poor communication among the team	(BASS et al., 2018), (ZHOU et al., 2014), (BEECHAM, 2014)	3
Long working hours	(BASS et al., 2018),(BAO et al., 2017)	2

to stress from work-related technology on turnover retention, productivity, commitment, job satisfaction, and technology-supported performance (YENER; ARSLAN; KILINÇ, 2020) and it is also suggested to decrease organizational commitment (HYNNINEN; PIRI; NIINIMÄKI, 2010).

Mismatch between expectations and reality were also found as an anchor linked to software engineers' possible turnover as the work environment, or specifically this simple mismatch between an individual's personality, their desires and aspirations, and what the job really provides that drives this takeoff (SMITH; SPEIGHT, 2006). Moreover, emotion-rule dissonance, like surface acting, can lead to negative outcomes for the actor, as the continued need to display emotions that are inconsistent with experienced emotions can deplete mental resources and lead to emotional exhaustion, psychological strain, and employee turnover (RUTNER et al., 2015). To the extent that a career anchor represents an individual disposition, it would not be surprising to find that individuals who are in an employment situation where the inducements

offered are inconsistent with their career anchor will tend to withdraw from the situation and exhibit negative behaviors such as absenteeism and turnover (AGARWAL; FERRATT, 2000).

Long working hours for software engineers is another identified cause of turnover along with employment policies, work-life balance, workplace innovation, product quality, alignment of offshore work hours with onshore, long working hours and adverse impact on health as factors affecting staff retention (BASS et al., 2018).Bass et al. (2018) still mentions that several engineers have stated that a relentless cycle of code production sprint after sprint served to create the impression of a treadmill work style causing stress and bad health, not-retained developers have a significantly higher value on the variance of working hours of project members for each month (BAO et al., 2017). Yet, the imbalance of working hours of project members in the first month might leave a bad first impression on the project and the company. This implies that the working environment has a big impact on developer turnover (BAO et al., 2017).

Payment and promotion have been some factors influencing intention to leave as both are part of distributive justice. Distributive justice relates to the perceived fairness of reward allocation and has been mentioned as the beginning of organizational justice which is the employees' perceived fairness in the workplace (UZOKA et al., 2011). This is also important because experienced IS professionals, especially those with scarce skills, can easily find positions at higher pay (BURNS; COLLINS, 1998). Further, whether or not a developer receives payment in exchange for his or her software development will impact his or her turnover intentions (DANIEL et al., 2020).

Autonomy, which can also be considered to be linked to freedom for decision making, is an essential factor as previous research work shows that, in general, autonomy negatively correlates with turnover intentions (MASSONI et al., 2019). Furthermore, even though incentives like salary and promotion are deemed as critical for leaving jobs, other determinants, such as job autonomy (MASSONI et al., 2019), most software engineers believe the organization is becoming too intrusive of their private space and many would want to disassociate themselves with employment entirely and set up their businesses instead, just to be able to exercise their freedom (UZOKA et al., 2011). However, such monitoring may undermine human resources (HR) practices intended to empower workers, give them greater autonomy, and encourage collaboration or information sharing. This suggests that firms should consider the implications of alignment between HR and IT practices (TAFTI; MITHAS; KRISHNAN, 2007). Therefore, distributed sites should have enough autonomy and freedom for decision-making.

Lack of motivation has been emphasized as a correlated factor with reported intentions to

leave by prior research related to turnover among IS professionals (SMITH; SPEIGHT, 2006; BASS et al., 2018) as tapping into the intrinsic motivation needs of the software engineer correlates to desirable outputs such as low staff turnover, higher productivity, and better quality software (BEECHAM; NOLL, 2015). Furthermore, highly motivated developers are more likely to remain in their current jobs, while lack of motivation may result in attrition and consequently turnover intentions (BASS et al., 2018).

Lack of satisfaction was found to be the most substantial, the most direct factor and a key predictor of staff turnover (UZOKA et al., 2011) as most software engineers need to have their job outcomes matching the desired ones (SMITH; SPEIGHT, 2006). However, ensuring high levels of job satisfaction among their developers is important for managers because lower job satisfaction will likely reduce the well-being of software developers and increase turnover intentions (FOERDERER et al., 2016) making those skilled and qualified individuals who frequently search for career paths and opportunities outside organizational boundaries that provide such satisfaction much more likely to leave (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013).

Lack of supervisor/management support is perceived as positively influent on software engineers turnover and on job insatisfaction (UZOKA et al., 2011). Not only management support but the more support individuals receive from their team, the more likely they are to feel empowered to carry out their tasks in a way they see fit and teams are more comfortable with allowing their members greater autonomy (JR; YOU, 2018). Therefore, once this lack of support is noticed turnover intentions might be generated (UZOKA et al., 2011).

Lack of connection with co-workers is a factor that stands for not having a connection with team members or not relying on others to make decisions and form attitudes (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013). On the one hand, this lack of connection is problematic as teams whose members do share similar characteristics are more cohesive, report higher levels of satisfaction and have lower turnover (GARRISON et al., 2010) but on the other hand, this lack of connection might be a good thing for the company in terms of turnover because the career moves of co-workers strongly influence individuals in attempts to define security, stability, and career success. When such referent others leave and cut their formal ties with the firm, the focal individual faces pressure to do the same (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013; MELAND; WAAGE; SEIN, 2005).

Lack of collaboration among the team is something that might cause an imbalance of workload and if the workload among project members is very different, the turnover of the project may increase (BAO et al., 2017) This lack of collaboration may also impact software engineers

at the early stages of their careers as they enjoy the support received from senior colleagues and usually need this kind of collaboration to evolve their skills (NOLL et al., 2017). Moreover, there could also be an impact on work estimation, management and team environment and these play a role in retention (BASS et al., 2018).

Poor communication among the team is a critical factor, especially in GSD projects where a well-established communication and well-documented decisions and alignments are usually required (JOHRI; TEO, 2018). Furthermore, having a poor communication with colleagues might not make the team able to properly help each other and consequently a lot of tension and pressure, which may lead to stress, in the environment will take place (BASS et al., 2018). This connects to the stress point mentioned in this section and could indeed lead the individual to turnover intentions. Yet, poor communication can decrease social interaction and influence on behavioural changes related to software engineers careers evolution (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013).

In summary, long working hours, mismatch between expectations and reality, stress, bad or incompatible payment as well as lack of a motivator environment, support and collaboration with a well established communication among the team are predictors of turnover intentions. On the other hand, improvements for each one of those would be a good sign from employers in order to minimize their effects and retain more people in projects or companies.

4.1.3 RQ1.3 - What factors mitigate turnover in GSD teams?

In the previous subsection 4.1.2 there were pointed factors that cause software engineers' turnover from a GSD project based on our SLR. In this one factors that mitigate turnover in GSD projects will be presented as it is important to know the what can affect the retention of developers and to have tools and methods to analyze and manage retention properly. Yet, understanding the effects of the factors related to the turnover rate in a project team can proactively adjust the project management to ameliorate these effects and improve the likelihood of success on a project (ZHOU et al., 2014).

Furthermore, even though it's been discussed about important factors related to turnover mitigation, certain organizations may not deliberately focus on retaining their IT workers, such as those with a "task focused," "utilitarian" or "incented technician" organizational mindset (BELLINI et al., 2019) and for some of them the number of negative conditions in the IS profession may be less significant to turnover (MOQUIN; RIEMENSCHNEIDER, 2013). On

Table 4 – Turnover mitigating factors by references

Related Factors	References	Occurrences
Collaborative environment	(BASS et al., 2018),(SMITH; SPEIGHT, 2006), (GARRISON et al., 2010), (UZOKA et al., 2011), (LIN; ROBLES; SEREBRENIK, 2017), (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013), (MONTEIRO et al., 2011), (ADYA; COTTON, 2012), (FOERDERER et al., 2016), (SUMNER; YAGER; FRANKE, 2005), (GALLIVAN, 2004), (MGAYA et al., 2009), (YENER; ARSLAN; KILINÇ, 2020), (JR; YOU, 2018), (REMUS et al., 2016), (CAMPBELL et al., 2013), (LACITY; IYER; RUDRAMUNIYAIAH, 2009), (STAPLES; WEBSTER, 2008) .	18
Satisfaction	(BASS et al., 2018),(SMITH; SPEIGHT, 2006), (GARRISON et al., 2010), (UZOKA et al., 2011), (LIN; ROBLES; SEREBRENIK, 2017), (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013), (MONTEIRO et al., 2011), (ADYA; COTTON, 2012), (FOERDERER et al., 2016), (SUMNER; YAGER; FRANKE, 2005), (GALLIVAN, 2004), (MGAYA et al., 2009), (YENER; ARSLAN; KILINÇ, 2020), (JR; YOU, 2018), (REMUS et al., 2016), (CAMPBELL et al., 2013), (LACITY; IYER; RUDRAMUNIYAIAH, 2009), (STAPLES; WEBSTER, 2008)	18
Autonomy	(JAIN; SUMAN, 2018), (MOURMANT; NIEDERMAN; KALIKA, 2013), (LUNDENE; MOHAGHEGHI, 2018), (MGAYA et al., 2009), (SETOR; JOSEPH, 2019), (JR; YOU, 2018), (FOERDERER et al., 2016), (MASSONI et al., 2019)	8
Supervisor support	(BASS et al., 2018), (NOLL et al., 2017), (SMITH; SPEIGHT, 2006),(MONTEIRO et al., 2011), (MOQUIN; RIEMENSCHNEIDER, 2013), (STAPLES; WEBSTER, 2008), (BEECHAM; NOLL, 2015), (BEECHAM, 2014).	8
Motivation	(BASS et al., 2018), (NOLL et al., 2017), (SMITH; SPEIGHT, 2006),(MONTEIRO et al., 2011), (MOQUIN; RIEMENSCHNEIDER, 2013), (STAPLES; WEBSTER, 2008), (BEECHAM; NOLL, 2015), (BEECHAM, 2014)	8
Good communication	(NOLL et al., 2017), (UZOKA et al., 2011), (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013), (LUNDENE; MOHAGHEGHI, 2018), (QUAN; CHA, 2010), (HYNNINEN; PIRI; NIINIMÄKI, 2010)	6
Balance of tasks along development cycles	(BASS et al., 2018), (UZOKA et al., 2011), (SETHI; KING; QUICK, 2004),(YENER; ARSLAN; KILINÇ, 2020),(HYNNINEN; PIRI; NIINIMÄKI, 2010).	5
Connection with co-workers	(BASS et al., 2018), (UZOKA et al., 2011), (BAO et al., 2017), (HYNNINEN; PIRI; NIINIMÄKI, 2010) .	4
Payment	(BASS et al., 2018), (UZOKA et al., 2011), (BAO et al., 2017), (HYNNINEN; PIRI; NIINIMÄKI, 2010)	4
Clear career orientation	(LUNDENE; MOHAGHEGHI, 2018), (SUMNER; YAGER; FRANKE, 2005), (MGAYA et al., 2009), (HYNNINEN; PIRI; NIINIMÄKI, 2010)	4
Workplace innovation	(BASS et al., 2018), (MELAND; WAAGE; SEIN, 2005), (MOURMANT; NIEDERMAN; KALIKA, 2013), (MONTEIRO et al., 2011)	4
All team members feeling involved	(SMITH; SPEIGHT, 2006), (BAO et al., 2017), (ZHOU et al., 2014)	3
Growth opportunity	(BASS et al., 2018), (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013), (TAFTI; MITHAS; KRISHNAN, 2007)	3
Good and active leadership	(BASS et al., 2018), (LIN; ROBLES; SEREBRENIK, 2017), (TAFTI; MITHAS; KRISHNAN, 2007)	3
Organizational commitment	(BASS et al., 2018),(BAO et al., 2017)	2
Work-life balance	(BASS et al., 2018), (FOERDERER et al., 2016)	2
Employment policies	(BASS et al., 2018)	1

the other hand, being able to predict who will leave the company early would enable the opportunity to retain the talented software developers and reduce the loss when they leave (BAO et al., 2017).

For instance, *balance of tasks along development cycles* was pointed as a great factor to mitigate turnover in GSD projects as balance collaboration and individualism, i.e., when

assigning maintenance tasks ensure that developers maintain both code developed by others and their own code. For those developers who mainly write new code, they could do more code maintenance tasks. Developers in charge of documentation should not only deal with documentations, instead, some coding tasks may increase their chances of staying in the projects (LIN; ROBLES; SEREBRENIK, 2017). Yet, a wise strategy for it would be allocating tasks to minimize the negative effect of cross-site communication (ZHOU et al., 2014).

All team members feeling involved in the software development cycle was found to be an important factor that fosters *connection with co-workers and organization commitment* as teams whose members do share similar characteristics are more cohesive, report higher levels of satisfaction, have lower turnover (GARRISON et al., 2010) as well as including off-site team members in decision making as Hynninen, Piri and Niinimäki (2010) states in their work where most of the technical tasks were moved to off-site, off-site team members were also given the responsibility to do the initial effort estimation on their own. These estimations were then peer reviewed with on-site, which gave a sense of autonomy to off-site in a sense that on-site actively let off-site to take part in the decisions made about off-site's work.

Moreover, *freedom for decision making* which relates to a certain kind of autonomy itself has been appointed as a factor that when in lack might lead to turnover intentions in the last section 4.1.2 while in this section job autonomy is also pointed as a factor to decrease work exhaustion and turnover intentions Foerderer et al. (2016). Structural determinants for turnover as autonomy, pay level, promotional chances, and social support are suggested to be positively related to organizational commitment and thus reduce the likelihood of voluntary turnover (HYNNINEN; PIRI; NIINIMÄKI, 2010).

While *supervisor/management support*'s absence was indicated as a factor that cause turnover in 4.1.2, the opposite is valid for this section as its presence is considered a great sign of satisfaction, specially for those at early career stages and is also related to turnover decreases (UZOKA et al., 2011).

Collaborative environment is perceived as a helper for reducing tension and pressure in the workplace which would decrease also stress level among the team and so the turnover intentions would be mitigated Bass et al. (2018). Moreover, the greater the involvement of more experienced technical staff in work, higher will be the sharing of workload among software teams (BASS et al., 2018).

Good communication as well as encouraging each other to ask questions is important when relying on mutual adjustment and frequent feedback which are related to low turnover rates.

For those who are perceived as good in asking questions, especially in the initial steps inside a project. Furthermore, it's important to state the main enabler for this communication running smoothly is a stable workforce while the team has a relatively low turnover (MOE et al., 2015).

Growth opportunities which covers opportunity for promotion, career development, training opportunities and annual assessments are found to be a factor which has a reasonable effect on organizational commitment and on job satisfaction and has a direct positive impact on job and career satisfaction Uzoka et al. (2011).Yet, it's very appreciated by software engineers as well as challenging work Remus et al. (2016) and are also key influences on turnover intentions in a way that employees who fail to find any growth opportunities at their places of work indicate that their careers have been a failure which in turn contributes to desires of leaving (UZOKA et al., 2011).

According to Bass et al. (2018), given that job satisfaction, motivators and de-motivation are considered predictors of staff turnover and software engineer are likely to stay longer in the job if they are satisfied motivation of developers to participate, and by implication remain, in projects is influenced by the identification of participants, the transformational leadership of leaders and an *active management style*. Agarwal and Ferratt (2000) also observed that explicit attention by managers to the needs of IT professionals at different stages of their careers can help alleviate the incidence of high turnover.

Satisfaction is another factor linked to turnover mitigation as job satisfaction is significantly impacted by growth opportunities and negatively correlated with turnover intentions. Items under job satisfaction cover issues such as pre-employment expectations, the easiness with which a new employee gets integrated at place of work, and job characteristics.Therefore, employers can try to reduce their employees' intentions to leave the job by ensuring that the above factors are taken care of Uzoka et al. (2011) and also hiring from within the organization is associated with lower IT turnover (ANG; SLAUGHTER, 2004).

Beyond the correlation cited in subsection 4.1.2 between turnover and lack of motivation, it's stated that when*motivation* can have a positive effect on staff retention Noll et al. (2017), Bass et al. (2018) and a positive correlation with engineering/management agreements on project success when perceived in its high level (BASS et al., 2018).

According to Bass et al. (2018) improving *work-life balance* and adopting family friendly *employment policies* are good strategies to improve development team member retention. For example, employee-friendly policies such as adjustments to work schedules or even allowing software engineers to leave work if they have any emergency at home are seen as supporting

staff retention.

Workplace innovation like developing new products, developing their own technology, introduce technological newness and innovations, is pointed as a driver for staff retention as greater effort to collate and reward innovations could be a useful tool to enable software product ownership to improve this retention. (BASS et al., 2018).

The emergence of recognition is clearly present in the IT Turnover literature, for instance through *payment or promotability* (MOURMANT; NIEDERMAN; KALIKA, 2013). Payment was found as a critical factor influencing intentions to leave with young software engineers feeling they are not being fairly compensated for the levels of performance that they are producing (UZOKA et al., 2011). Furthermore, it is also a factor positively related to motivation and productivity (DANIEL et al., 2020) which is, indeed, directly linked to lower levels of turnover intentions (NOLL et al., 2017). Among determinants for turnover, autonomy, pay level, promotional chances, and social support are suggested to be positively related to organizational commitment and thus reduce the likelihood of voluntary turnover (HYNNINEN; PIRI; NIINIMÄKI, 2010).

Clear career orientation is important because it directly addresses other software engineers' turnover factors such as satisfaction and commitment (SMITH; SPEIGHT, 2006). Therefore, it is essential to know the career orientation of employees because based on such knowledge, organisations can reduce high turnover rates by paying attention to several aspects of software engineers' wants and needs and by providing job opportunities and incentives that match their career orientation (MGAYA et al., 2009) which will enable companies to predict who will leave the company early and flourish the opportunity to retain the talented software developers and reduce the loss when they leave (SMITH; SPEIGHT, 2006) Yet, organizational support for with guidance and certification programs significantly reduces turnover intention (QUAN; CHA, 2010).

In summary, employment policies, work-life balance, workplace innovation, product quality, alignment of offshore work hours with onshore, long working hours and adverse impact on health were identified as factors affecting staff retention (BASS et al., 2018). However, organisations and IS managers seeking a reduction in turnover must find ways to close the gap between employee's wishes and what employees perceive they have in their current positions as careful attention should be paid to several aspects of employees' wishes (SMITH; SPEIGHT, 2006) such as reasonable balance of tasks along development cycles, involvement of all team members, their commitment and connection with co-workers providing a collaborative envi-

ronment with a great communication, freedom for decision making and on the other hand supervisors support and an active management style. This environment should be open to innovative ideas and full of growth opportunities and payment levels review with clear career orientation and work-life balance as well as friendly employment policies.

4.1.4 RQ1.4 - What impact does autonomy have on turnover within GSD teams?

We identified the main factors related to autonomy linked to turnover in GSD projects by mapping the quotes and categorizing the outcomes from autonomy and checking which of them were mentioned as a turnover precedent (See Table 5). The literature has widely addressed the importance of considering *employees' wishes* and aligning them with what the organization can offer. Furthermore, software engineer's opinions about autonomy and its levels in an organizational context need to be investigated, considering the specificities of work (SMITH; SPEIGHT, 2006).

Table 5 – Autonomy impacting factors on turnover

Related Factors	References	Occurrences
Work Regime	(BASS et al., 2018), (NOLL et al., 2017), (UZOKA et al., 2011), (FOERDERER et al., 2016), (JOHRI; TEO, 2018).	5
Employees' wishes	(NOLL et al., 2017), (MOURMANT; NIEDERMAN; KALIKA, 2013), (CHOI; TAUSCZIK, 2017),(HYNNINEN; PIRI; NIINIMÄKI, 2010).	4
Individual factors	(BASS et al., 2018), (NOLL et al., 2017), (LUNDENE; MOHAGHEGHI, 2018),(JR; YOU, 2018).	4
Career Satisfaction	(BASS et al., 2018), (SMITH; SPEIGHT, 2006), (GARRISON et al., 2010), (UZOKA et al., 2011).	4
Communication	(LUNDENE; MOHAGHEGHI, 2018), (ZHOU et al., 2014), (BEECHAM, 2014).	3
Contributor's opinion	(LUNDENE; MOHAGHEGHI, 2018), (UZOKA et al., 2011), (BAO et al., 2017).	3
Job opportunities	(UZOKA et al., 2011),(JOHRI; TEO, 2018), (MGAYA et al., 2009).	3
Work Routine	(SETOR; JOSEPH, 2019), (LUNDENE; MOHAGHEGHI, 2018), (MONTEIRO et al., 2011)	3

Communication was also pointed as an important factor related to autonomy, mainly in distributed projects. When a company cannot establish an adequate communication flow is common to find inefficient work processes and employees with low levels of commitment to the organization and, thereby, high turnover levels (NOLL et al., 2017).

Other factors related to *individuals* are also relevant such as mental overload, conflicts related to roles, social support, organizational noticed support, job demands, engagement and burnout. Individuals factors are directly related to autonomy because they can strongly influence motivation and satisfaction at work and might still contribute to more significant

intentions of turnover if they are not balanced (BELLINI et al., 2019).

Job opportunities and *career satisfaction* directly influence turnover intentions. Autonomy relates to these factors because it can significantly improve professional satisfaction by allowing the employee to elaborate plan, prioritizations and practices within the work. Autonomy can also improve quality tasks by flourishing feelings of a more specialized work to the ones involved in execution when applied in a motivated team with a high level of satisfaction with their careers (MASSONI et al., 2019).

Work regime shows itself as a significant factor related to autonomy because even in a small way, autonomy is commonly perceived as a positive and preventive factor to turnover. However, for that to happen, managers must look for ways to ensure good communication among team members and ensure that team monitoring is not perceived in an authority way because such practice tends to reduce this feeling of individual autonomy (BEECHAM; NOLL, 2015).

Furthermore, *work routine* is also an essential factor related to autonomy. A certain level of autonomy is perceived as beneficial to tasks execution because it allows an employee to prioritize the project meant to him and establish practices judged as necessary to its execution. This higher freedom to prioritization gives the employee a feeling of higher control, which becomes a preventive factor for turnover intentions (REMUS et al., 2016).

4.2 SURVEY RESULTS

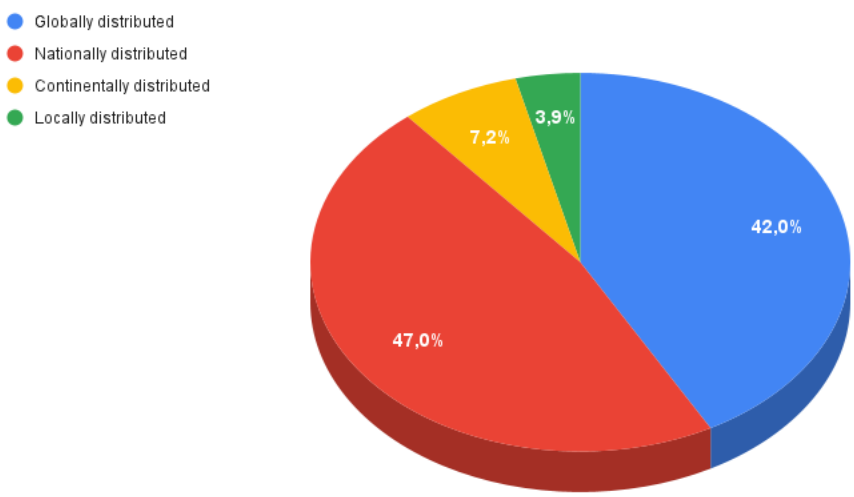
4.2.1 Study Population

The survey yielded complete answers from software engineers involved in the software development life cycle regarding autonomy, turnover, and their relationship inside distributed environments with 85 participants (47%) working remotely inside Brazil, 76 participants (42%) working with other software engineers globally located, 13 participants (7%) who worked with team members from the same continent and 7 participants (3%) working in locally distributed teams within the same city or state. This data can be seen in Figure 9.

Most of the identified participants were men (69.1%), some were women (28.7%) and completing the total number of participants there were 1.1% non-binary and 1.1% of not identified gender as shown in Figure 10.

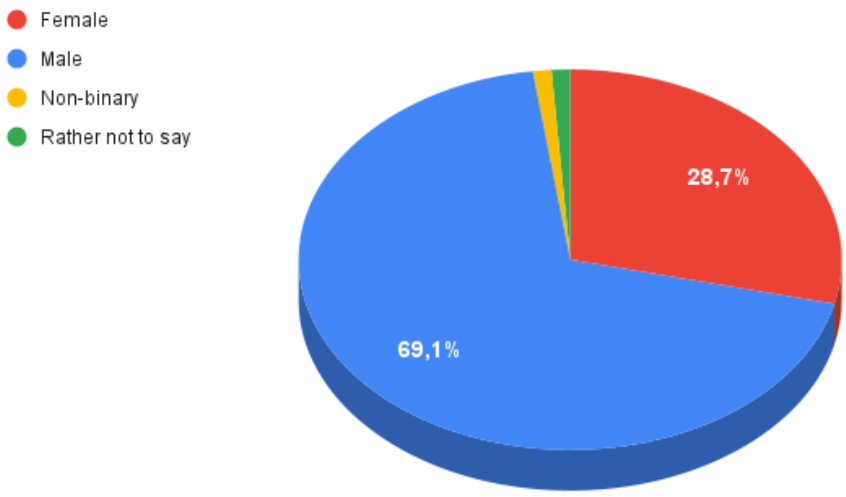
Yet, 87.3% of the participants were found to work in a large company, 3.9% worked in

Figure 9 – Participants' team distribution



Source: Author

Figure 10 – Participants gender

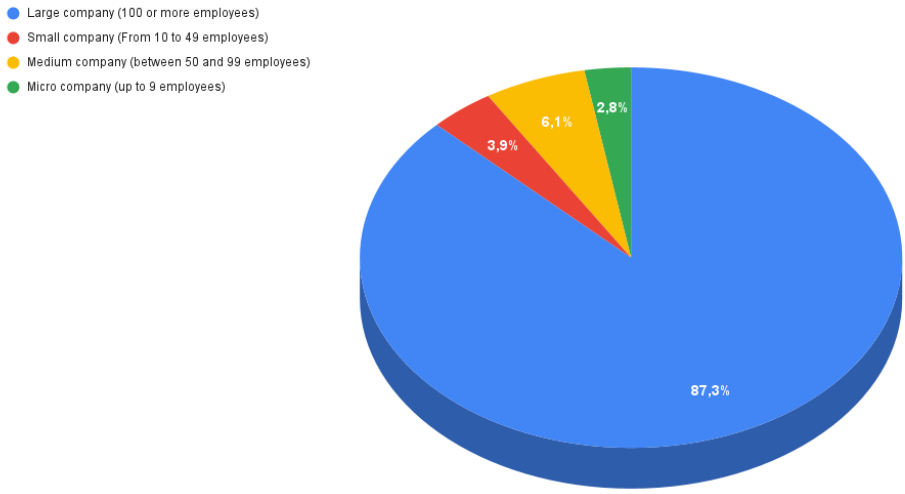


Source: Author

a small company, 6.1% worked in a medium company and 2.8% worked in a micro company as seen in Figure 11. Furthermore, Figure 12 shows a range of experience in the software development area were stated amongst them ranging from less than a 1 year (6), through 1-5 years (92), 6-10 years (36), 10-15 years (22) to more than 15 years (14). Yet, 54.1% of the participants were software developers, 22.7% were testers, 5.5% were software architects, 4.4% were requirements engineers and the rest of them were split among management/leadership

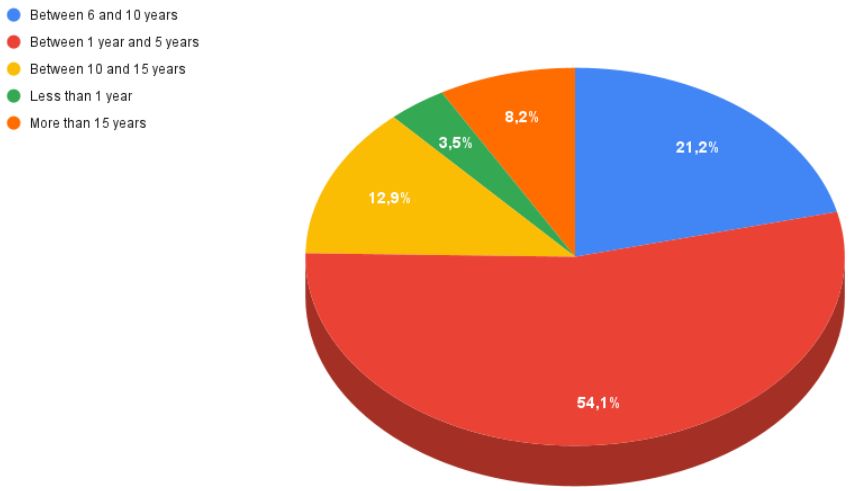
roles, UX/UI, Scrum Masters, trainees and other roles as seen in Figure 13.

Figure 11 – Participants' companies size



Source: Author

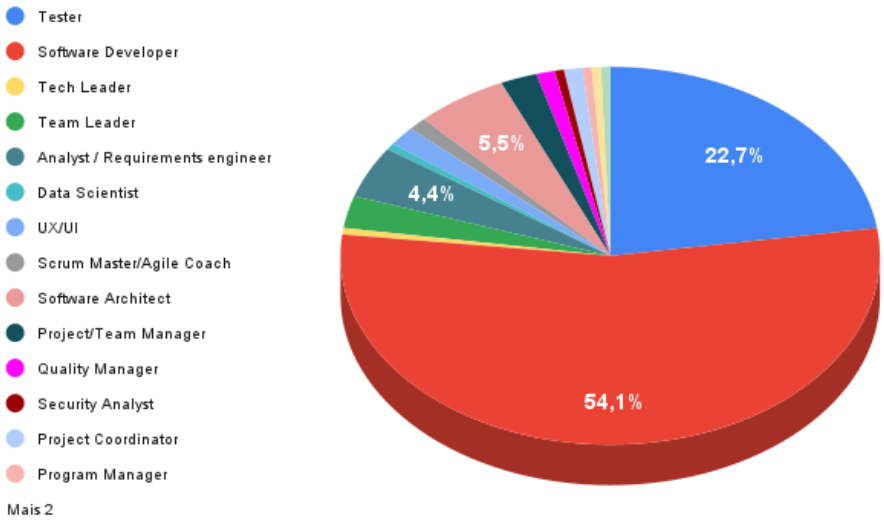
Figure 12 – Participants experience level



Source: Author

Moreover, 65.7% of participants acquired a bachelor degree, 13.3% had completed high school while 8.3% stated to have achieved a master's degree. There is still 5% of them with a post graduation degree and more 5% with a technical degree. Yet, only 3% declared to have an MBA or some sort of specialization concluded. See in Figure 15.

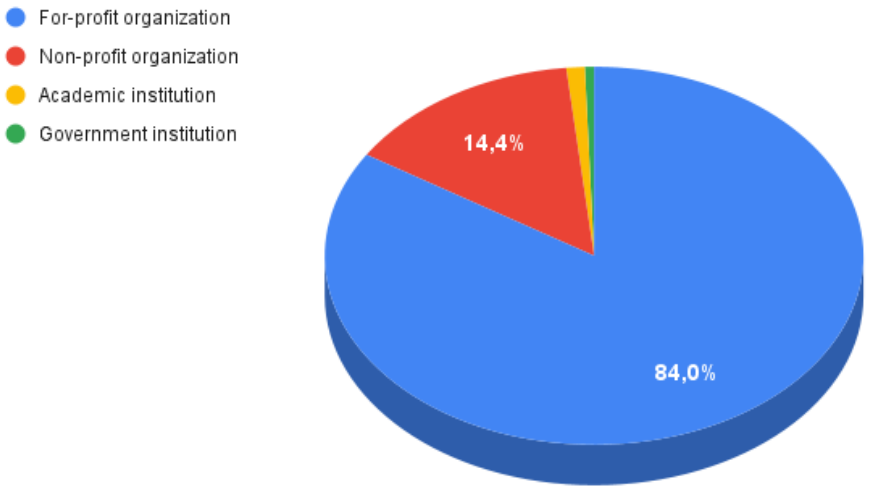
Figure 13 – Participants roles in SDLC



Source: Author

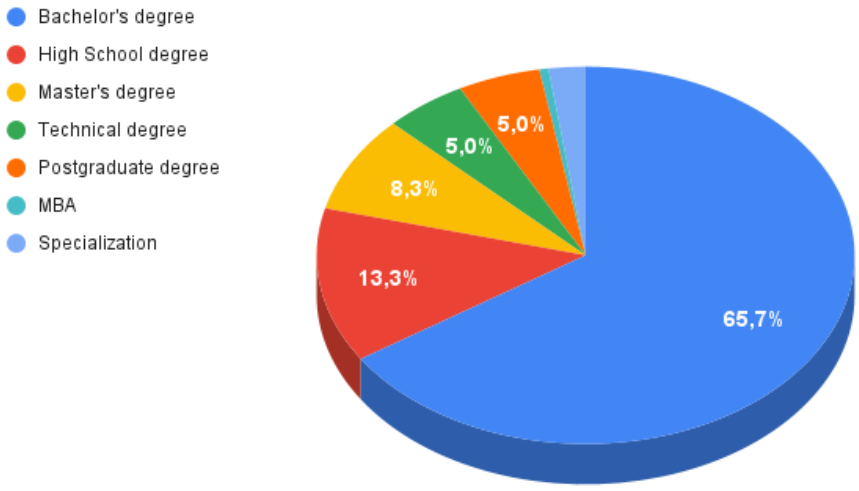
Most of them were part of for-profit organizations (84%) while 14.4% were from non-profit organizations and 1.6% declared to be part of government or academic institutions as seen in Figure 14.

Figure 14 – Participants organization type



Source: Author

Figure 15 – Participants education level



Source: Author

4.2.2 RQ01. How autonomy is perceived by software engineers within distributed software teams?

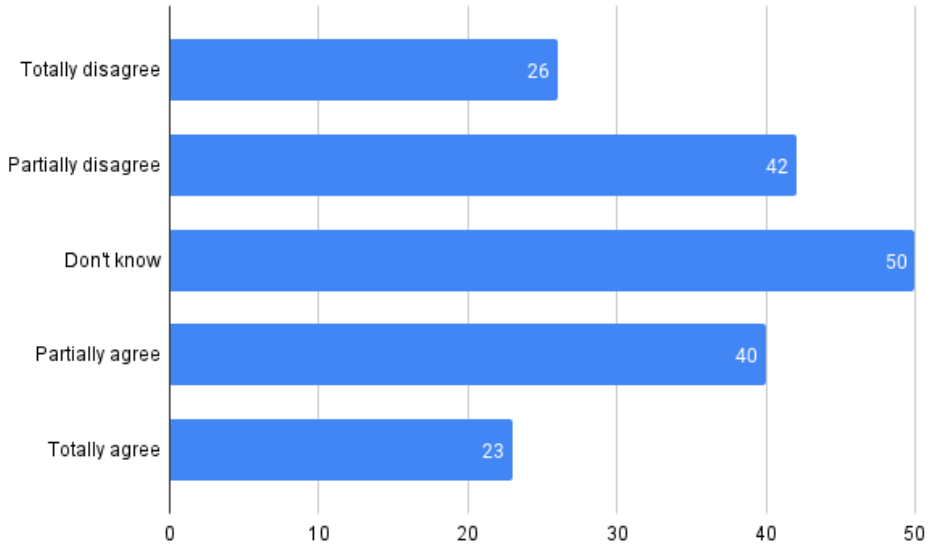
It was possible to notice a pretty well-balanced collected data where 50 (27.6%) participants did not know whether their work was being intensely monitored. At the same time, 63 (34%) declared there is at least some strong monitoring of their work and 68 (37%) of the participants informed there is little monitoring within their work routine in software development cycles as shown in Figure 16.

Regarding teams having a shared purpose, clear goal, necessary skills, and mutual trust among peers, 160 (88.3%) participants agreed at least partially to it. In contrast, only 12 participants (6.6%) denied it, and the other nine stated that they did not know whether this happened within their teams, as shown in Figure 17.

Yet, results show that 156 (86.1%) participants agree they have at least some healthy support and freedom to make decisions provided by managers and leaders. On the other hand, 12 participants (6.6%) denied it, and 13 stated that they did not know about any support or freedom from leadership members, as seen in Figure 18.

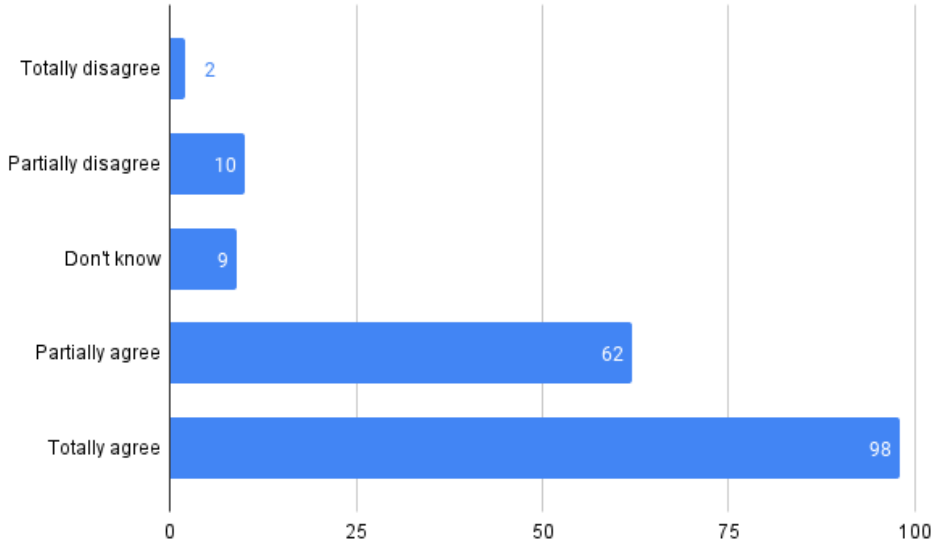
Furthermore, the data collected for those three questions were analyzed to find a connection between them. Thus, for those who stated their work was intensely monitored in RQ01, it was possible to identify most of them were in teams with a shared purpose, clear goals, necessary

Figure 16 – RQ1.1 - Your work is strongly monitored by the company or any other stakeholder



Source: Author

Figure 17 – RQ1.2 - Your team has a shared purpose and clear goal, the necessary skills, and mutual trust among the team members

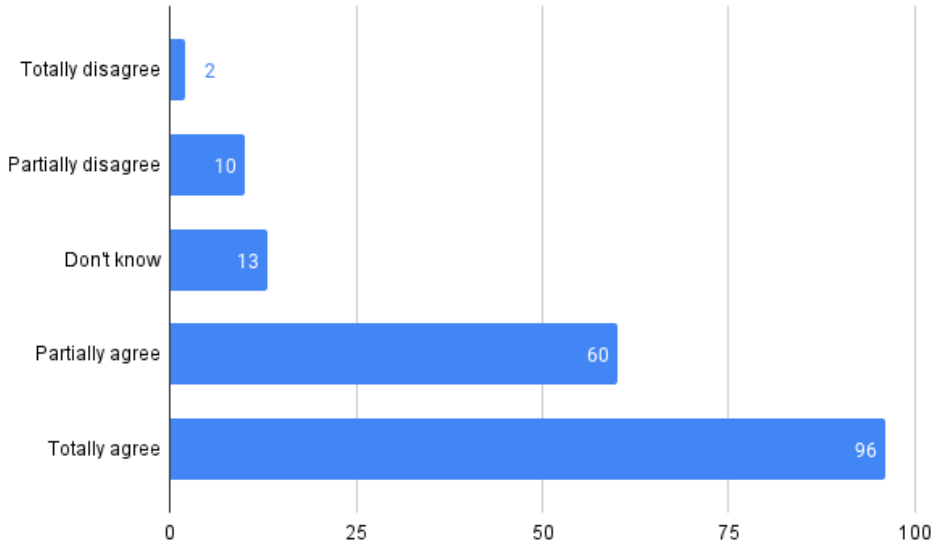


Source: Author

skills, and mutual trust among members, as can be seen in Figure 19.

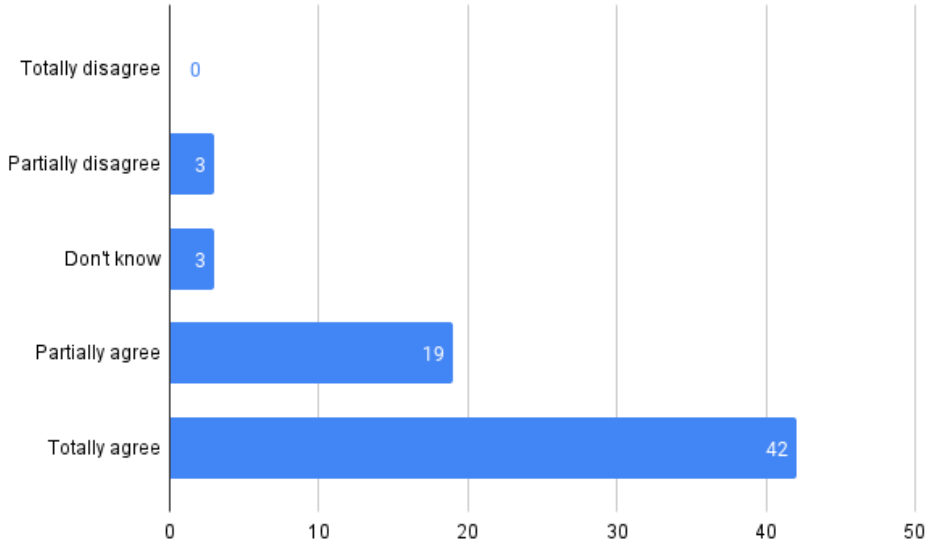
A similar scenario is found when we relate the identified low-monitored software engineers to an environment with healthy support and freedom to make decisions provided by managers and leaders. Most of them state this is true, as seen in Figure 20.

Figure 18 – RQ1.3 - People in management and leadership roles provide good and healthy support to your team and yet give you the freedom to walk in your own shoes



Source: Author

Figure 19 – Low monitored software engineers' answers regarding having a collaborative team



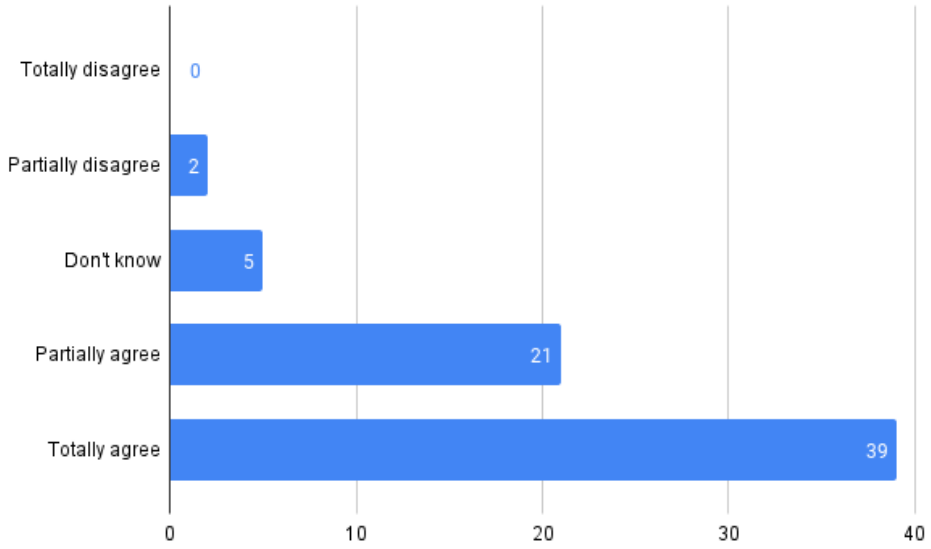
Source: Author

4.2.3 RQ02. Which turnover motivators are the most perceived for software engineers within distributed software teams?

To assess the causes and mitigation factors of turnover within distributed teams, we presented a section to the participants regarding this matter with questions and statements.

According to results, 85.5% of all 181 participants stated they have already decided to

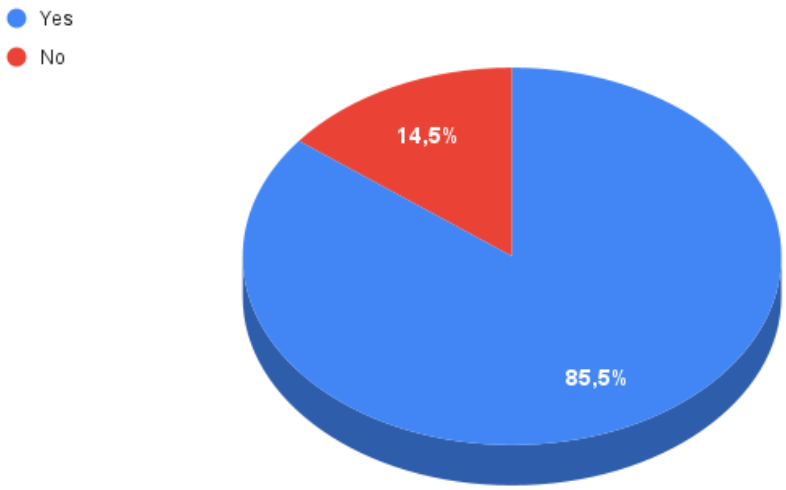
Figure 20 – Low monitored software engineers' answers regarding having healthy support from leadership



Source: Author

leave a project or job, while 14.5% of them have never been through such an experience as seen in Figure 21.

Figure 21 – Have you ever decided to leave a project or company?



Source: Author

Participants could also select the most critical factors in deciding whether to leave a project or company. The outcomes revealed a ranking of perceived factors such as payment, stress, lack of motivation, the mismatch between expectation and reality, lack of motivation, lack of supervisor support, long working hours, lack of satisfaction, poor communication among

Table 6 – Most important factors to leave a project or company according to participants. (*n=181*).

Option	#	%
Payment	145	80
Stress	130	72
Lack of motivation	112	62
Mismatch between expectations and reality	106	59
Lack of collaboration among the team	101	56
Lack of supervisor support	97	54
Long working hours	92	51
Lack of satisfaction	88	49
Lack of connection with co-workers	73	40
Lack of freedom for decision making	73	40

Table 7 – Most important factors to staying in a project or company according to participants (*n=181*).

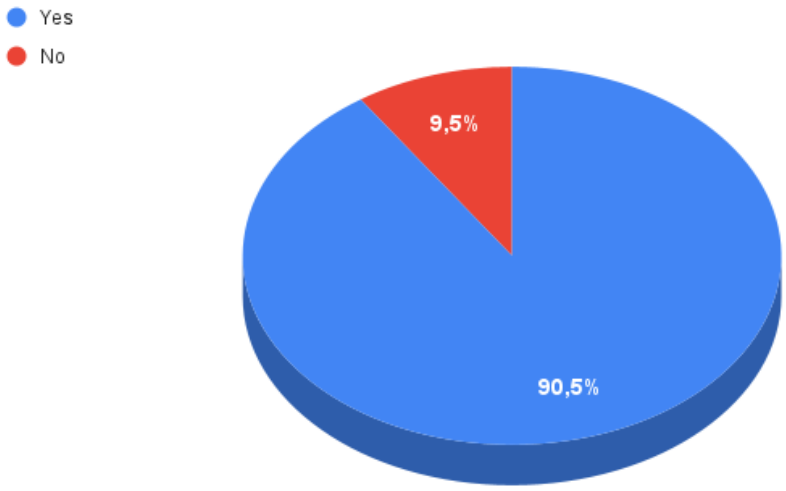
Option	#	%
Payment	158	87
Growth opportunity	153	85
Work-life balance	143	79
Collaborative environment	136	75
Good communication	119	66
Supervisor support	115	64
Motivation	110	61
Good and active leadership	107	59
Freedom for decision making	102	56
Connection with co-workers	101	56
Balance of tasks along development cycles	93	51
Satisfaction	90	50
Clear career orientation	79	44
All team members feeling involved	73	40
Organizational commitment	67	37
Workplace innovation	50	28
Employment policies	24	13

peers, lack of freedom for decision making and lack of connection with co-workers as can be seen in Table 6.

On the other hand, participants could select the factors they considered the most important ones in deciding whether to stay in a project or company. Based on their answers, another ranking was lightened with perceived factors such as payment, growth opportunity, work-life balance, collaborative environment, good communication, supervisor support, motivation, good and active leadership, freedom for decision making, connection with co-workers, the balance of tasks, satisfaction, clear career orientation, all team members feeling involved, organizational commitment, workplace innovation, and employment policies as can be seen on Figure 7.

Yet, 90.4% of them declared they had already been to a distributed project or company where a software engineer left, while 9.5% stated this never happened before, as seen in Figure 22.

Figure 22 – Have you ever been in a distributed project where a software engineer left?



Source: Author

Table 8 – Most important outcomes after a software engineer leaves a project or company according to participants (*n=181*).

Option	#	%
Loss of knowledge and experience	154	85
Lower levels of productivity	78	43
Software quality	57	31
Lack of commitment and mutual trust among peers	41	23
Economic loss to companies	28	15
Better conditions and performance for those who remain	24	13
Fresh and innovative ideas	18	10
Unsuccess of software project	17	9
Project success	8	4

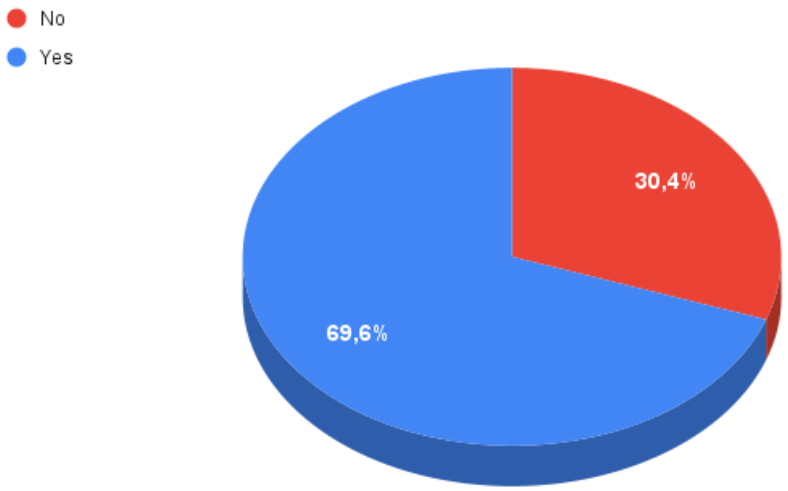
Also, all participants stated the most common outcomes that occur after a person leaves a project or company the participant is still a part of. Those outcomes were also brought up in a ranking that contained items such as lower levels of productivity, project success, loss of knowledge and experience, software quality, better conditions and performance for those who remain, economic loss to companies, lack of commitment, and mutual trust among peers, fresh and innovative ideas, software project unsuccessful as can be seen in Table 8.

4.2.4 RQ03. How does autonomy impact turnover in distributed software teams?

We dedicated a section to deeply understand how autonomy and turnover relate based on participants' views. First of all, we asked all of them if "they believe a higher level of autonomy

in a distributed project is a key factor for not willing to leave it" (RQ3.1), and 69.6% agreed to it, while 30.4% did not, as seen in Figure 23.

Figure 23 – Do you believe a higher level of autonomy in a distributed project is a key factor for not being willing to leave it?



Source: Author

Then, we aimed to identify how autonomy was perceived by participants in two career moments. Therefore, we stated two optional open questions to ask for real experiences (if there were any) regarding any moment where some certain autonomy was given when the participants were at early career stages and when they were at a more experienced level.

Firstly, based on the participants view we were able to collect common generated outcomes given the autonomy software engineers receive at the early stages of their careers as can be seen on Figure 24.

Trust was a notable one as some participants stated words like *"loved the autonomy and could notice an even greater growth based on trust to go on with activities"* (P3) or when P16 says *"I enjoyed receiving the acknowledgment and trust from team members"*. Yet, P20 stated *"It's a tough situation but really fulfilling because it demonstrates the team's confidence in my work"* and P32 completes saying that in addition, it's a really good sign there is mutual trust among team members.

Moreover, *recognition* has also been mentioned as flourishing from given autonomy at early stages when P28 says *"I was happy for being more acknowledged and for assuming more responsibilities with the autonomy of choosing code development pathway"* and P61 states *"it's important for personal and professional growth to have autonomy for decision making*

as it felt there was a recognition in its skills as it's necessary trusting the work of those who take decisions". Yet, P97 stated that *"receiving autonomy was very important for its own acknowledgment of skills because based on this autonomy a self-confidence took place professionally which made the simple fact of asking for help much easier. Indeed, asking for help had always been faced as an embarrassing moment before having received this autonomy because it sounded like some incompetence."* It indicates communication and collaboration are other outcomes for given autonomy as well.

Both *communication* and *collaboration* were also linked to given autonomy by other participants such as P40 stating that *"even though there was not much knowledge about how to execute the tasks there was freedom to execute them and to ask for support when needed"* as software engineers are always encouraged to seek for more experienced ones (P42).

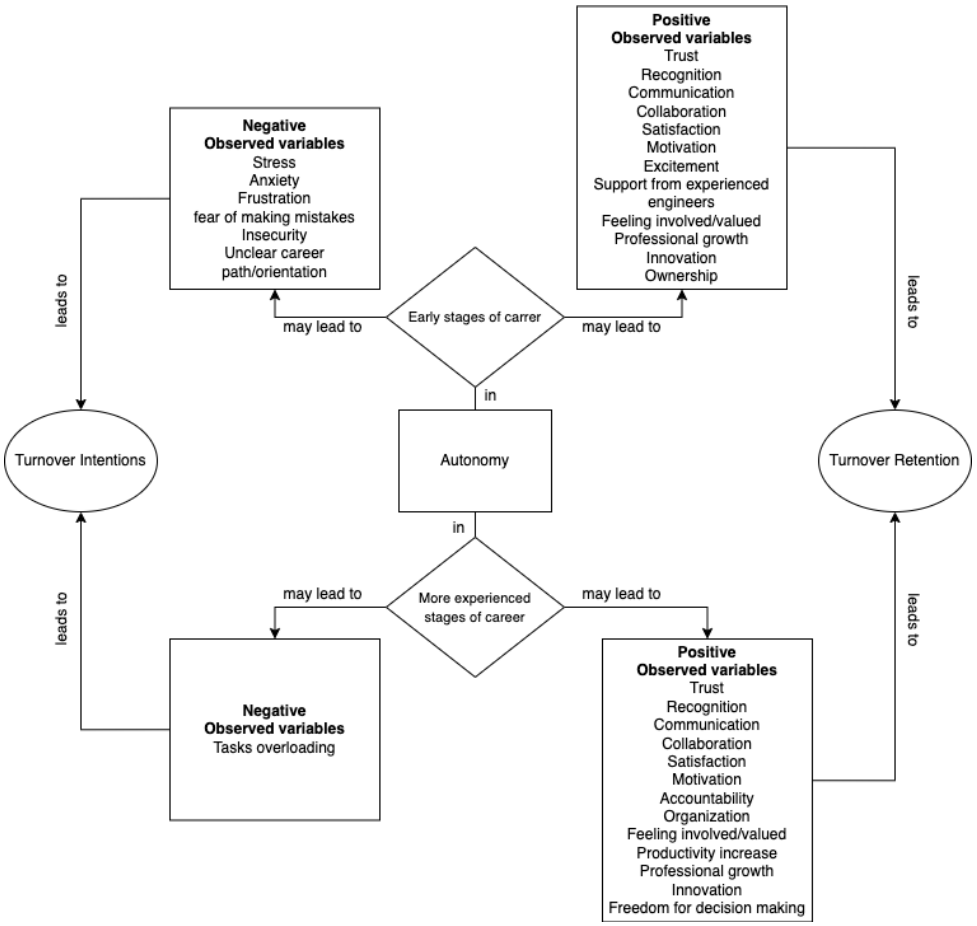
This *support from more experienced engineers* was also found to be a crucial point when providing early-stage professionals with some level of autonomy as those more experienced ones should act as a consultant trusting and promoting the less experienced engineers to think and come towards a solution and then guiding the way when needed and establishing a secure referential for the team (P57)(P67). Autonomy was also perceived to promote *motivation* in participants at this early stage as P50 confirms it by saying *"I felt like someone important and relevant to the team right away, and that motivated me a lot to keep going and do my best."* and P93 confirms it by stating *"autonomy is important to keep working with more motivation"*.

Nevertheless, not only positive outcomes have emerged from participant's statements. Anxiety, fear, and insecurity were the most negative feelings present in our collected data regarding the given autonomy to those software engineers at early career stages.

Anxiety has been mentioned as the first fact once some task was given to P134 and it's also linked to *fear* of not being able to execute it and match expectations or having code quality diminished (P36). Later on, this anxiety and fear are found to vanish as stated by P134 that *"in the course of time I felt more comfortable knowing I was backed up by the team in more complex situations"* and by P36 *"when I had questions I could reach to my tech leader and got proper help"*. Yet, P105 and P143 were afraid of having to take decisions when started but both of them declare that after a while this experience becomes the best career challenge in order to evolve and in the end classify this process as a good experience. On the other hand, P80 states *insecurity* was present for a long time because its work environment did not have a more experienced software engineer to provide this help when needed. *"In short, giving*

autonomy to a starter generates insecurity given the lack of experience and knowledge (about the project, company goals). It surely can be considered a good thing in cases where you have a bit more time within the project or field.", stated P135.

Figure 24 – Perceived outcomes raised from autonomy



Source: Author

However, participants had a different perspective when it came to receiving a certain autonomy at more experienced stages of their careers, as can be seen in Figure 24.

Freedom for decision making has been an important outcome highlighted by them in this more experienced scenario as P127 states, *"it is wonderful to have freedom for choosing the right path for task execution"* and P143 who was able to perform research about the best approach for the problem faced before solving it.

Although many participants state this freedom, they also mention an increase of *accountability* linked to this level of given autonomy to experienced engineers; as P32 says, *"It's very challenging because with autonomy also brings the responsibility of the tasks upon you in order to have them done which leads to a good* professional growth *at the end"*. Also, P50 states, *"there might be some nervousness when faced with this kind of accountability at first,*

but when overcame professional growth feelings are felt while satisfaction and motivation for the future take place".

In fact, it was possible to see through participants' answers that this professional growth is a common existent variable linked to autonomy during or once the task is finished.

"Autonomy is an important factor on my satisfaction *within the software team"*, says P43. Yet, P111 states viewing autonomy as the possibility and capacity of executing tasks and making decisions related to them and not only receiving them without questioning. That brings a very satisfactory and accomplished feeling. It's very important for both professional growth and productivity increase.

Another raised topic regarding this matter is that even though autonomy is faced as beneficial it does not exclude the need for *communication and collaboration*. Indeed, communication with team members is seen as very valuable as it allows technical discussions and opinions on solution decisions which leads the team feeling involved and satisfied. Asking for help is found to be as important as having autonomy inside a software development team.

Trust was mentioned by two perspectives at this stage: the first one based on the trust the team has on a person to provide some autonomy and on the other hand the trust more experienced engineers are encouraged to foster within the team in order to promote this autonomy throughout the whole development cycle.

Moreover, a set of 7 statements were presented to address how each participant perceived autonomy and turnover intentions within their workplace. The participants were asked to agree or disagree with those using the Likert scale.

The first statement presented was *"When I have the freedom to work as preferred, I feel more motivated, satisfied, and willing to stay longer at a project or company"* and to that 171 (94.4%) participants at least partially agreed while 8 (4.4%) declared to be neutral about it and only 2(1.2%) partially disagreed. Therefore, it's possible to see the majority of participants indicate a positive relationship between freedom to work as preferred and motivation and satisfaction decreasing the chances to leave a project or company.

The second statement was *"When I don't know anything about the task assigned to me, I feel stressed and start to think about giving up. If this becomes frequent, I tend to leave the project or company."* and based on that 52 (28.7%) of participants declared to at least agree to that partially, 33 (18.2%) participants stated neutrality about it and 96 (53.1%) participants denied this statement. Therefore, most participants indicate there is no connection between missing competence to stress and consequent turnover intentions when

given autonomy is present. This corroborates with the vision brought by some participants about feeling challenged and willing to learn new things within the work environment which are an indication of motivator to stay longer at a project or company.

Then, *"When my team and I have freedom to work as preferred, we also have better control of tasks since the refinement phase through planning and execution. This process leads to quality improvements and connections with other teammates."* was the statement displayed to the participants in which 161 (88.9%) of them agreed at least partially to it, 19 (10.4%) were neutral and only 1 (0.7%) person disagreed. It's possible to state the cause-effect relationship between given autonomy and software quality and connection with peers.

Furthermore, the fourth statement presented was *"When I am free to choose the tasks I want to work with I usually seek the ones that will challenge me the most and during the development of it I feel delighted"* and then 122 (67.4%) participants agreed to this at least partially and 49 (27.1%) of them found it to be neutral and only 10 (5.5%) denied it at least partially. This positive statement agreement by most participants indicates how software engineers are motivated and satisfied and confirms the search for new challenges are beneficial towards staying in the project or company when autonomy is provided.

"When I take care of some complex tasks by myself I feel like someone that everyone can count on and I feel committed to the team's goals. This makes me more engaged." was the fifth statement presented and 147 (81.2%) participants declared to agree to this at least partially while 28 (15.4%) participants were neutral about it and only 6 (3.4%) participants denied it partially. Based on the positive outcome from most of the participants there is a strong indication that given autonomy provides individual trust and feelings of being valued by software engineers which positively influences commitment to the software team and its goals which can lead to positive impact on staff retention.

Yet, 133 (73.7%) participants agreed to *"I'd rather work from home or at a quiet place where there's only me and my computer when I take care of some complex tasks by myself and this gets me less exhausted at the end of the day."* while 31 (17.1%) disagreed to it at least partially and only 4 (2.2%) of them found it neutral. This result indicates physical isolation are negatively linked to exhaustion when autonomy is provided which can be positively related to lower turnover intentions. Based on distributed work environments this might be a positive outcome as physical team presence is less seen. It's also something that indicates a possible need for a stronger previous alignment about the work that will be executed autonomously.

Finally, 127 (70.1%) participants agreed at least partially to the following statement *"If*

I could choose the tasks I would like to work with I'd rather pick the ones related to new products or ideas than the ones related to support or legacy items. This would make me feel more involved and willing to stay longer at a project or company.", 45 (24.8%) were neutral about it and 9 (4.97%) of them disagree at least partially to it. Also, according to results there is a strong indication software engineers would rather work with and provide innovative ideas than support tasks which is positively linked to higher engagement and consequent reduction of turnover rates within software teams.

4.3 CLOSING REMARKS

This Chapter discussed the gathered results from the research study. We provided a comprehensive explanation of the aimed outcomes in the research, encompassing both the systematic literature review and the survey findings. Furthermore, SLR questions were answered revealing the state-of-the-art and related factors among autonomy and turnover relationship such as identified factors that contribute to autonomy in GSD, identified factors that contribute to turnover in GSD, identified factors that mitigate turnover and what impact autonomy have on turnover in GSD teams.

Based on the conducted SLR we found out internal autonomy is driven by an established work regime and routine as well as decreasing the monitoring levels in a GSD team will should flourish good autonomy levels. Also, individual autonomy is driven by the ability of a software engineer on taking more than one role or task which can be summarized as cross-functionality. This is also linked to mutual trust among team members as trust from others have the power to grant more autonomy to a software engineer. Job opportunities and career satisfaction are also considered as factors that contribute to an individual autonomy takes place. Also, a good support from management and leadership are factors that enable external autonomy in a GSD team. However, factors like lack of satisfaction, low levels of autonomy, lack of motivation, lack of management support, stress and payment were identified as contributors to turnover in GSD teams while collaborative environments, high levels of satisfaction and autonomy, management support, high levels of motivation, good communication and balance of tasks along development cycles were considered mitigating factors for turnover. Yet, when examining the influencing factors connecting autonomy and turnover in GSD teams, it was found that autonomy is associated with a decrease in turnover rates when combined with a favorable work regime, employees' aspirations, career satisfaction, effective communication,

team members' sense of involvement, job opportunities and a well-structured work routine.

Later on, it was possible to delve into applied survey results by describing the participants demography details and collecting software engineers perception regarding how autonomy takes place within their distributed software teams, which turnover motivators and mitigating factors were the most outlined ones and how autonomy impacts turnover within their teams based on their vision. Yet, shedding light on some main outcomes and moderators in autonomy-turnover relationship such as experience levels, communication, trust and recognition.

Based on results gathered, external autonomy was balanced among participants as there was no conclusive answer regarding them having their work strongly monitored by the company or any stakeholder. On the other hand, managers and leaders were found very supportive by providing freedom and a healthy support to their team. Furthermore, most of them declared to have a shared purpose and mutual trust among team member which indicated a good internal autonomy. Moreover, we also identified a collaborative environment for those low monitored software engineers and a good support from their leader as well which indicate a relation between provided autonomy and collaboration and support.

Payment was identified as the main motivator and mitigating factor related to turnover. Stress and lack of motivation were also found to contribute to higher turnover rates while growth opportunity and work-life balance were pointed as most important factors to staying in a GSD project or company.

In conclusion, the outcomes of autonomy are associated with several career stages of software engineers and are linked to both positive and negative results, which may or may not result in turnover intentions. Indeed, if autonomy is granted to a software engineer in the early stages of their career without adequate support, it may lead to anxiety and stress and increase the chances of turnover. Conversely, autonomy at this stage can serve as motivation and enhance communication and trust among team members. For experienced software engineers, excessive autonomy can result in task overloading, while also increasing productivity and job satisfaction.

5 DISCUSSION

Autonomy and turnover have been widely discussed in previous research (LIN; ROBLES; SEREBRENIK, 2017; NOLL et al., 2017; BEECHAM, 2014; CHAVES et al., 2022; MARINHO et al., 2021) separately and usually linked to some factors such as motivation, satisfaction, happiness, communication, collaboration, and many others. In our work, we addressed the perception of the relationship between them. Based on all gathered and analyzed data, it is possible to establish a discussion regarding the main research question: *"Does the software engineer's autonomy impact turnover in GSD Teams?"* declared at the beginning of this document. Therefore, some lessons learned are listed as follows in order to support its answer.

5.1 AUTONOMY PERCEPTION

Autonomy could be defined from different perspectives, such as individual autonomy, internal and external autonomy, where external autonomy is defined as the influence of management and other individuals outside the team on the team's activities. Internal autonomy refers to the degree to which all team members jointly share decision authority, while individual autonomy refers to the amount of freedom and discretion an individual has in carrying out assigned tasks (MOE; DINGSØYR; DYBÅ, 2008).

Yet, the autonomy of teams can be defined by their freedom to work in close cooperation with product owners and users to take ownership of their processes and practices and to take responsibility for their interfaces with other systems. Also, some pre-conditions to be realized are needed, such as redundancy of skills (since it affects the team's capability to adapt to changing situations), culture (such as team orientation), and sharing of information so that all team members have the knowledge to influence decisions and management support in order to create the right environment for the teams. Therefore, teams need a shared purpose, the necessary skills, and mutual trust among the team members to develop internal autonomy (LUNDENE; MOHAGHEGHI, 2018).

Our first survey section was designed to assess those autonomy aspects regarding participants within distributed software development teams. Figure 17 in Chapter 4 shows most participants in this context developed internal autonomy by having a work environment fostered with a shared purpose and clear goal towards the project, the necessary skills, and mutual

trust among team members which also indicates a higher level of motivation and satisfaction among them (NOLL et al., 2017) and consequent lower turnover expectations.

Although autonomy is usually not a problem when working remotely, a prerequisite for remote working is the ability to work independently. However, individuals can be undermined if the head office is heavy-handed, and interferes with communication, say with on-site customers, or if their work is monitored too stringently. For developers working under the customer's spotlight, autonomy can be problematic (LUNDENE; MOHAGHEGHI, 2018). In this study, a question was displayed in order to shed light on these monitoring levels within participants from distributed teams, and its results are shown in Figure 16 from Chapter 4 stating a balance of monitoring levels among participants work environments. Other studies reported that an individual's level of attachment to an organization is influenced by connections with co-workers, and the greater those connections, the slower the decay of attachment to the organization over time (BURT, 1995). To deepen the analysis, it was possible to establish a connection between low-monitored software engineers and how developed their internal autonomy is, as seen in Figure 19.

On the other hand, job satisfaction, supervisor support, and growth opportunities are key influences on turnover intention (UZOKA et al., 2011) while motivated software engineers enjoy the support they receive from senior colleagues or leadership, satisfying (in part) their need for connectedness (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013). Also, mentoring is positively correlated with commitment and negatively related to turnover behavior (PAYNE; HUFFMAN, 2005). and to influence personal learning and job satisfaction (LANKAU; SCANDURA, 2002). Therefore, an analysis regarding whether healthy support was being provided to low-monitored software engineers was established, as seen in Figure 20. Results show most of the participants within this context of low monitoring stated to have healthy support from leaders and managers, which indicates fewer stressors toward turnover intentions are identified for this scenario.

5.2 MOTIVATING FACTORS AND PREVENTATIVE MEASURES FOR REDUCING TURNOVER

Among many factors claimed to cause and prevent turnover in GSD teams, a list with the main ones from literature was identified as described in sections 4.1.2 and 4.1.3 in Chapter 4. Based on these results, a section about turnover causes in GSD was presented to participants to assess their perception regarding which of the presented causes were the most important

ones.

Over the years, IT companies have made efforts to retain their workers by increasing *salaries* and offering benefits in order to create an environment attractive for high-performing knowledge workers. However, although people were fairly satisfied with their prior positions, external market forces such as greater opportunities, salaries, and benefits in the IT field may have led them to leave their jobs (MELAND; WAAGE; SEIN, 2005). Therefore, low payment is seen in literature as an important factor associated to demean staff retention (QUAN; CHA, 2010; MASSONI et al., 2019) where even organizations that offer competitive salaries and work with leading-edge technologies experience high levels of dissatisfaction and higher than desired turnover among their IT staff (BASS et al., 2018). In this study, participants selected this factor as the most important factor influencing both retention and turnover, as seen in tables 6 and 7. Therefore, it indicates the strong need for companies to address and provide a good career path with recurrent salary reviews and feedback cycles regarding satisfaction and dissatisfaction points, including payment.

Stress among information system (IS) professionals is long recognized as a key factor affecting software engineers' productivity and turnover and leading to substantial associated costs (SETHI; KING; QUICK, 2004). Also, it is becoming increasingly clear that steps must be taken to address the high-stress problem because of its effect within work environments (SETHI; KING; QUICK, 2004) as reported by other studies in the literature (YENER; ARSLAN; KILINÇ, 2020). Also, stress and routinization are suggested to decrease organizational commitment, which is considered the most substantial and the most direct influence on turnover intentions among software engineers (IGBARIA; GREENHAUS, 1992). In this study, stress has been appointed as the second factor influencing software engineers to leave GSD teams. It indicates the importance and need of caring about this characteristic within the work environment by promoting actions that may ease stressors.

However, self-managing teams tend to encourage the participation and involvement of their members, resulting in greater commitment, motivation, and accountability to the work (DECI; RYAN, 2012). Also, high levels of motivation can have a positive effect on staff retention as Hall *et al.* found a positive correlation between software engineer motivation and employee turnover(HALL et al., 2008) while tapping into the intrinsic motivation needs of the software engineer correlates to desirable outputs such as low staff turnover, higher productivity, and better-quality software (BEECHAM; NOLL, 2015). Yet, highly motivated software engineers are likelier to remain in their current jobs, while lack of motivation may result in attrition (BEECHAM

et al., 2008; VERNER et al., 2014). Therefore, our results show lack of motivation and motivation itself were, respectively, considered crucial factors in order to leave a distributed team and remain on it.

Nevertheless, the imbalance of working hours of project members in the first month might leave a bad first impression on the project and the company (BAO et al., 2017). This implies that the working environment has a big impact on developer turnover. Additionally, the work-life balance of software engineers is found to decrease work exhaustion and turnover intentions (FOERDERER et al., 2016). Yet, staff member turnover could be reduced by improving work-life balance and adopting more family-friendly employment policies (BASS et al., 2018). In fact, results show work-life balance being highly considered by participants as a critical factor influencing a software engineer to stay within a distributed team. Also, working for long hours was a well-selected factor regarding leaving a distributed team, according to participants.

Furthermore, skilled and qualified individuals frequently search for career paths and opportunities outside organizational boundaries that provide such satisfaction (HESLIN, 2005). It is also important to know the career orientation of employees because, based on such knowledge, organizations can reduce high turnover rates by providing job opportunities and incentives that match the career orientation of their employees (MGAYA et al., 2009). In this study, growth opportunities were stated as the second most important factor to staying in a distributed team which indicates the need for companies to invest and establish opportunities to foster skills and develop their software engineers aiming further staff retention.

Moreover, turnover is also generated by the work environment, or specifically a mismatch between an individual's personality, desires, and aspirations (SMITH; SPEIGHT, 2006). This mismatch between expectation and reality is pictured in this study's results as the fourth-ranked factor linked to leaving a distributed team.

Also, one aspect of the boundaryless career is that individuals are often more strongly connected to co-workers than they are to the formal firm (ARTHUR; ROUSSEAU, 2001) while lacking connectedness can lead to a feeling of isolation (NOLL et al., 2017; RYAN; DECI, 2000). On the other hand, ties at the workplace can provide an individual with co-worker support and a sense of belonging and are found to be negatively related to voluntary turnover (FELPS et al., 2009; MOSSHOLDER; SETTOON; HENAGAN, 2005). Yet, increasing trust among the team members through close collaboration and face-to-face communication creates good relations and an environment for learning as it was observed that team members were adopting new skills from their team peers(LUNDENE; MOHAGHEGHI, 2018). In this study, collaboration among

peers was found to be the fourth more important factor influencing software engineers to stay within the project or company. This collaboration, usually fostered by connection, can also lead to collective turnover intentions as software engineers tend to follow others with who they are connected inside the team (LUNDENE; MOHAGHEGHI, 2018). However, the opposite is also true, according to participants, as lack of collaboration is mentioned as the fifth-ranked factor on the list that would leave participants from distributed teams to leave a project or company.

Additionally, support from more experienced engineers is acknowledged as an internal career-related variable influencing turnover intentions (UZOKA et al., 2011). Individuals driven by a desire for autonomy, on the other hand, derive satisfaction by disassociating themselves with employment (MGAYA et al., 2009) but aspects of good management, such as accurate work estimates and creating a supportive environment, play a role in staff retention (BASS et al., 2018). Yet, team members new to their roles may lack the skills and experience to make decisions without consulting others, as motivated junior developers enjoy the support they receive from senior colleagues, satisfying (in part) their need for connectedness (NOLL et al., 2017). Also, social support is suggested to be positively related to organizational commitment and thus reduce the likelihood of voluntary turnover (HYNNINEN; PIRI; NIINIMÄKI, 2010). Results show a collaborative environment as the fourth factor influencing software engineers to stay within a GSD team while lack of collaboration among team members is also highlighted as the fifth-ranked factor that would make a software engineer leave.

Yet, distributed members should respect each other's culture and values and have enough autonomy and freedom for decision-making (JAIN; SUMAN, 2018). Also, promoting organizational freedom could reduce the possible disagreement with the rules of the business game, hence reducing the chances for software engineers' turnover (MOURMANT; NIEDERMAN; KALIKA, 2013). Other researchers consider autonomy as the degree to which the work provides substantial freedom for the individual in scheduling the work and determining the procedures for carrying out work tasks (SETOR; JOSEPH, 2019). Therefore, participants ranked freedom for decision-making within GSD teams as one of the 10 most important factors for staying within a GSD team, while lack of this freedom has not been appointed as a reason so important in order to leave the workplace environment. However, freedom for decision-making and given autonomy have been widely mentioned in open questions as well as its impacts seen in Section 4.2.4 from Chapter 4 that will be broadly discussed in Section 5.4 in this very chapter.

5.3 THE CONSEQUENCES OF EMPLOYEE TURNOVER

Understanding the repercussions of employee turnover is crucial. It is imperative to recognize the impact it can have on the overall success and productivity of a company causing significant loss for the company because software engineers could depart with a lot of critical knowledge and experience (BAO et al., 2017). Ignoring this issue can lead to detrimental consequences and hinder the growth of the business by causing economic loss to companies. Therefore, it is essential to address employee retention and take proactive measures to ensure that turnover is minimized and reduce the loss if they leave (BAO et al., 2017). Previous studies have shown developer turnover has a significant impact on project success. Frequent developer turnover may lead to loss of productivity due to lacking relevant knowledge and spending extra time learning how projects work (LIN; ROBLES; SEREBRENIK, 2017) and impacting the overall quality of those projects (FERREIRA; SILVA; VALENTE, 2020). However, low turnover may also be a concern as fresh and innovative ideas are needed periodically (MELAND; WAAGE; SEIN, 2005).

In fact, participants in this study were asked whether they were in a project where a coworker left, as seen in Figure 22, and also asked to rank the most important consequences they identified within the project after the loss of a coworker, as seen in Table 8. Results indicate loss of knowledge and experience as the most important issue faced, followed by lower productivity levels and impacts on software quality. This corroborates the findings from the literature (LIN; ROBLES; SEREBRENIK, 2017; FERREIRA; SILVA; VALENTE, 2020; BAO et al., 2017) and opens possibilities for further research towards practices that could be established by companies regarding this scenario.

5.4 AUTONOMY IMPACT ON TURNOVER

Based on results displayed in Section 4.2.4, it is possible to discuss the findings compared to known literature and contribute with some lessons learned from this executed study.

Lesson 1: Autonomy is positively related to trust which is negatively related to turnover

Most software engineers believe the organization is becoming too intrusive of their private space and many would want to disassociate themselves with employment entirely and set up their businesses instead, just to be able to exercise their freedom (UZOKA et al., 2011). In

fact, freedom for decision-making has been cited in our results as one of the most important factors linked to autonomy in distributed teams, especially for those more experienced software engineers. Still, there was no indication of disassociation with companies. Instead, there were many indications that this freedom is strongly linked to trust and accountability. This mutual trust among peers and sense of accountability enables them to share more activities which consequently fosters them to share information so everyone has the knowledge to influence decisions (JR; YOU, 2018; LUNDENE; MOHAGHEGHI, 2018; BEECHAM, 2014).

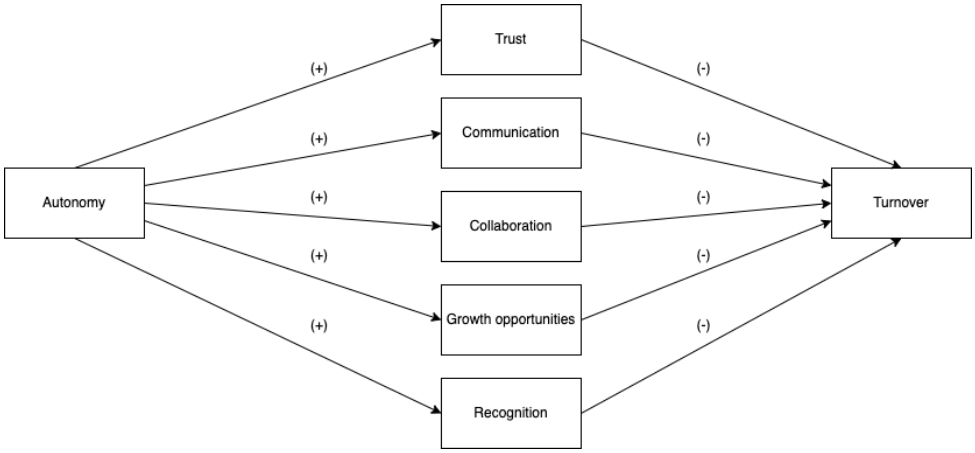
However, GSD faces many challenges, including communication, coordination, control, efficiency, lack of trust, higher conflict rates, issues regarding the protection of intellectual property, and socio-cultural distance, among others (COLOMO-PALACIOS et al., 2012). Yet, trust manifests itself in several different ways, and its building is an adaptive process that depends on several factors in global software engineering (ARAMO-IMMONEN; JAAKKOLA; LINNA, 2011; KIELY; BUTLER; FINNEGAN, 2022). This fact drives to the fact that building trust in global contexts where companies are networked geographically, socially, and culturally is a difficult task. Therefore, this study indicates autonomy as a positive factor within this trust-building process for GSD teams which could lead to reducing turnover intentions and practical staff retention as the level of trust between parties has a significant effect on performance measures while distrust creates a lack of commitment inside a team (ARAMO-IMMONEN; JAAKKOLA; LINNA, 2011). Low commitment itself has been established as the most substantial and the most direct influence on turnover intentions among software engineers in earlier literature studies (IGBARIA; GREENHAUS, 1992).

Lesson 2: Simply having autonomy is not enough to retain software engineers

According to previous research, most professionals who lacked autonomy in their job would be willing to leave as autonomy negatively correlates with turnover intention (MASSONI et al., 2019). However, autonomy was perceived differently according to software engineers' experience levels in this study, as seen in Figure 24 in Section 4.2.4. In fact, experienced software engineers lacking autonomy seem to have a probability of leaving a distributed team, but it can be commensurated with their competency (NOLL et al., 2017) while autonomy provided to less experienced or inexperienced software engineers can lead to stress and consequent turnover (BASS et al., 2018). Therefore, this study also shows that autonomy received in an early career stage has been stated to depend on support from more experienced engineers or leadership in order to foster mutual trust and not promote a scenario of fear of making mistakes and anxiety feelings on them and this is compliant with recent works where lack of

supervisor/management support is perceived as an influence on software engineers' turnover and dissatisfaction (UZOKA et al., 2011; JR; YOU, 2018). Also, autonomy is encouraged once this management support is facilitated in an effective way fostering an environment of trust and a culture of valuing individuals (TAFTI; MITHAS; KRISHNAN, 2007). Yet, organizational support with guidance and certification programs significantly reduces turnover intention (QUAN; CHA, 2010). On the other hand, autonomy provided to more experienced software engineers encourages their participation and involvement, resulting in greater commitment, motivation, and accountability to the work (MONTEIRO et al., 2011). Still, this study identified a need for balancing task overload to them prevent issues such as burnout and low engagement which are ultimately related to decrease commitment to the job and job-related behaviors (BELLINI et al., 2019).

Figure 25 – Autonomy impact on turnover and their main related factors



Source: Author

Lesson 3: Autonomy is positively related to communication and collaboration, which is negatively related to turnover

Past studies report that an individual's level of attachment to an organization is influenced by connections with co-workers and the greater those connections, the slower the decay of attachment to the organization over time (BURT, 1995). Those connections at the workplace can provide an individual with co-worker support and a sense of belonging and are found to be negatively related to voluntary turnover (FELPS et al., 2009; MOSSHOLDER; SETTOON; HENAGAN, 2005). In summary, according toGopalakrishnan, Halgin and Borgatti (2013), ties to other employees create inertial pressures on an individual and constrain him to conform to firm norms and thus stay with the firm.

Furthermore, this connection among peers has already been identified as both good and

bad regarding turnover intentions as it may retain software engineers working together for a longer period (GARRISON et al., 2010) but on the other hand, it may become an influence for higher turnover as the career moves of co-workers strongly influence individuals in attempts to define security, stability, and career success (GOPALAKRISHNAN; HALGIN; BORGATTI, 2013; MELAND; WAAGE; SEIN, 2005).

Even though autonomy is not a problem for those working remotely as it has the ability to work independently as a pre-requisite (BEECHAM, 2014), it's also found to have a strong link to communication inside distributed software teams (NOLL et al., 2017). Yet, a lack of adequate communication inside a company is commonly related to low levels of commitment and high turnover levels (NOLL et al., 2017). In our study, we could identify the need for communication stated by those software engineers who have been given autonomy at the workplace in different career stages and also that good communication was ranked as an important factor in order to stay in a distributed software team. Therefore, it supports the vision brought by Bass *et al.* where poor communication is related to a lot of tension and pressure, which may lead to stress (BASS et al., 2018), and this stress can have a considerable impact on organizational commitment and turnover levels (SETHI; KING; QUICK, 2004; YENER; ARSLAN; KILINÇ, 2020; HYNNINEN; PIRI; NIINIMÄKI, 2010).

Moreover, collaboration reduces workplace tension and stress levels among the team (BASS et al., 2018). A collaborative environment is an important factor for participants' decision to stay in a project. When collaboration is lacking, it can lead to workload imbalances and potentially increase project turnover (BAO et al., 2017). Experienced participants in our study mentioned that autonomy can result in task overload due to increased accountability. Lack of collaboration can also impact early-career software engineers who rely on support from senior colleagues for skill development (NOLL et al., 2017). Additionally, lack of collaboration can affect work estimation, management, and team environment, which influence retention (BASS et al., 2018). Our results indicate that lack of collaboration is a significant factor influencing participants' inclination to leave a project or company.

Lesson 4: Autonomy is linked to professional growth opportunities

Yet, other factors linked to autonomy and turnover are present such as growth opportunities and motivation. Some participants have stated that the autonomy received by them acted as a motivator during working days and also flourished professional growth opportunities after challenges had been overcome. This view matches prior research where autonomy has been reported to have a positive impact on job satisfaction and as a general motivator for software

developers while job satisfaction has been found to have a significant impact on turnover intentions as workers who have high job satisfaction are less likely to leave (LIN; ROBLES; SEREBRENIK, 2017; BASS et al., 2018).

Nevertheless, lack of motivation has been emphasized as a correlated factor with reported intentions to leave by prior researches related to turnover among IS professionals (SMITH; SPEIGHT, 2006; BASS et al., 2018) as tapping into the intrinsic motivation needs of the software engineer correlates to desirable outputs such as low staff turnover, higher productivity, and better quality software (BEECHAM; NOLL, 2015).

Lesson 5: Autonomy is positively related to recognition which is negatively related to turnover

Recognition was a positive outcome mentioned by participants who have experienced autonomy as they feel valued and involved. In fact, autonomy, pay level, promotional chances, and social support are suggested to be positively related to organizational commitment and thus reduce the likelihood of voluntary turnover (HYNNINEN; PIRI; NIINIMÄKI, 2010) while according to Uzoka (UZOKA et al., 2011), good payment and promotion can be considered a kind of recognition for the services provided, and they influence the intention to leave. Our work indicates that it not only payment is related to turnover intentions but also to turnover retention.

This perception is quite well aligned with the appreciation for challenges stated in (REMUS et al., 2016) and it indicates the positive impact on job and career satisfaction after opportunities for career development, promotions and training opportunities(UZOKA et al., 2011). Therefore, even though payment is identified as a factor linked to turnover, autonomy can be considered a factor that leads to factors like professional growth opportunities, motivation, the satisfaction which can be as important as payment.

5.5 IMPLICATIONS AND PRACTICAL POSSIBILITIES

Theoretically, this study plays a fulfilling role within software engineering area by addressing a relationship not yet established between two important factors within a distributed context: autonomy and turnover. Although autonomy may not be sufficient on its own to retain software engineers in GSD teams because of an interdependence of several other factors, this study can be seen as a first step in order to deepen the research about this matter and expand it into new practical tools such as frameworks and evaluation tools and guidelines to support managers,

leaders and entire distributed software teams on better leveraging their autonomy, behavioural factors and turnover intentions and rates.

In terms of research method, this study presents preliminary findings obtained through a Systematic Literature Review (SLR) followed by a survey conducted with 181 software engineers from distributed software teams over the course of one month.". However, other methods could and should be applied to broaden the analysis and deepen the understanding about the results shown. An updated SLR and a new survey with more participants from several companies around the world could be an interesting way for comparing the found results with the ones presented in this study as it would increase the gathered data and provide a higher accuracy to the explored subjects. Also, the survey protocol could be updated aiming the investigation of this autonomy-turnover relationship focused on software team roles and their respective activities and perceptions during a number of sprints as this would provide us a view by each role such as autonomy-turnover relationship in leadership, UX/UI teams, business teams or inside software quality assurance or developer teams. Yet, an action-research to validate how exactly autonomy takes place within distributed software teams environments for a longer period as well as interviews with those who leave the company during that would be another valid approach in order to identify a list of actions needed to balance this relation. Nevertheless, the influence of cultural habits from GSD teams could be studied based on an ethnographic approach once this autonomy-turnover relationship is better established.

In practical ways, this study could be used by many software companies as a source of found suggestions and actions that could be taken to prevent turnover intentions and turnover rates because of given autonomy at certain professional levels. It could also be used as a starting point for meetings and discussions regarding how well the company introduces challenges and arise trust to their teams in a balanced way that will not frustrate a less experienced software engineer making them willing to leave; or not overloading a more experienced software engineer in order to providing the opportunity to be creative, support the team in a better and paced time and therefore feeling valued during their daily routine. Moreover, this study also builds a path for companies to review their teams communication and collaboration process in order to provide a more motivating, engaged and autonomous environment which leads us to another key point provided by this study: autonomy is related to professional growth opportunities. Therefore, companies all over the world should use the gathered results from this study to check whether the given autonomy within GSD teams are followed by opportunities to develop their professionals and whether their recognition is the fairest possible. Based on found results,

the recognition types inside a company with GSD teams should also be reviewed together with the provided autonomy as salaries should follow the applied efforts, career opportunities and professional growth generated by autonomy which, at first, could play an essential impact on turnover retention.

Furthermore, software engineers within GSD context could use this study to evaluate their environment in comparison to shown results by sharing provided lessons learned with teams and improve their ways of working with given autonomy aiming a greater well-being to all inside the company. On the other hand, they could use this study to identify the need for searching another project or company where autonomy is provided in a balanced manner and join this other environment to satisfy and keep evolving in their careers.

5.6 THREATS TO VALIDITY

This study has some limitations that will be presented in this section. A qualitative perspective was applied based on perceptions from participants and it is important to be aware of different contexts, environments and individuals. Therefore, all data analysis was based on this evidence and we cannot state all results match the full picture regarding autonomy and turnover and distributed software development teams. Furthermore, all data and classifications presented in this paper need to be treated carefully as we are only providing indications.

Our survey was applied to 181 participants during 2 months, revealing time as a limitation. We encourage new researches to take place during a longer period.

5.7 CLOSING REMARKS

This chapter discussed our results aiming to establish a wider theoretical and practical view about the findings and how they relate to existent literature and GSD teams in real life. Autonomy perception, turnover motivators and preventives, the consequences of a software engineer turnover and the impact of autonomy on turnover within GSD teams were highlighted. Lessons learned regarding autonomy-turnover relationship were also discussed and shared as well as theoretical, practical and social implications. Lastly, we could also address threats to validity of this study.

6 CONCLUSION

Our study focuses on the relationship between autonomy and turnover in Global Software Development (GSD) teams, which have gained popularity as a working model in recent years (HERBSLEB, 2023). Furthermore, COVID-19 pandemic has considerably changed and fasten the way of working inside software teams worldwide and has also opened the possibilities of remote work (MARINHO et al., 2021) making software engineers able to work for whom and from anywhere they wanted. According to Neto et al. (2022) stress level increased since software engineers began working from home during pandemic. Also, complaints about the challenges related to remote teamwork and collaboration, as well as emotional issues, distractions and poor home office environment and equipment have become more frequent while this environment change positively affected individuals who benefited from better organization of work, increased flexibility and focus (SMITE et al., 2022). Balance between work and leisure is considered a challenge for those working from home, while job overall productivity and job satisfaction tended to benefit from the home environment from now on (LUNDE et al., 2022). Yet, many researchers believe it is crucial to continue monitoring the distributed work experiences to better understand the long-term effects of remote work that might not have yet surfaced (CLEAR, 2021) as well as their related factors such as autonomy and turnover.

Therefore, we have gathered valuable insights from real software engineers working in GSD teams. The findings of our study confirm that autonomy is indeed linked to turnover and turnover intentions in GSD teams, with various factors influencing this relationship. Effective communication, collaboration, trust, recognition, and task balance are key components influencing how autonomy impacts turnover among software engineers of different experience levels.

For early-career software engineers, the provision of autonomy can lead to feelings of anxiety, insecurity, and fear, which may result in lower commitment and increased turnover intentions. This highlights the importance of support from more experienced engineers, leaders, and managers in providing guidance and mentorship. On the other hand, experienced software engineers are better equipped to handle autonomy without experiencing adverse outcomes. However, task overloading becomes a concern for them, as increased accountability may lead to stress and an unbalanced working routine, potentially leading to a turnover.

Regardless of experience level, the perception of autonomy as beneficial for software en-

gineers is contingent upon effective communication, collaboration, and mutual trust among team members in the work environment.

Recognition plays a crucial role in sustaining the benefits of autonomy. Companies should establish practices to acknowledge and reward software engineers for achieving goals through their granted autonomy. Recognition, particularly when tied to salary increases or bonuses, is valuable and can potentially reduce turnover intentions.

Lastly, our study highlights the perceived outcomes for projects or companies when one or more software engineers leave. Loss of knowledge and decreased productivity were identified as the primary consequences. Further investigations are recommended to understand better and address these issues.

In conclusion, our study sheds light on the relationship between autonomy and turnover in GSD teams, emphasizing the significance of communication, collaboration, trust, recognition, and balancing task responsibilities. Understanding and effectively managing autonomy can create a positive work environment and reduce turnover intentions among software engineers.]

6.1 FUTURE WORK

Therefore, according to conclusions and threats to validity in Section 5.6 it is possible to point the following items as possible future work:

- Explore presented results with a quantitative method approach to statistically relate all factors regarding autonomy and turnover to provide deeper analysis about this relationship within GSD teams;
- Elucidate the real-world practices that would increase autonomy and decrease turnover within GSD teams;
- Develop a guideline with those good practices to be used by managers and companies within GSD teams;
- Update the executed SLR to identify more recent studies and contributions regarding this matter;
- Expand the applied survey to more software engineers worldwide to have a more accurate vision of the studied relationship between autonomy, turnover and side effects;

- Perform a study about autonomy and turnover by software development life cycle roles or areas;
- Perform a long action research with GSD teams to better investigate this relationship between autonomy and turnover;
- Develop a framework for companies regarding the provision of autonomy in each experience level and how to deal with its uncertainties and outcomes
- Investigate shared leadership role towards autonomy and GSD teams turnover intentions

REFERENCES

ADYA, M.; COTTON, J. Exploring the role of mentoring in the is profession: A cross-national comparison. In: *Proceedings of the 50th annual conference on Computers and People Research*. [S.l.: s.n.], 2012. p. 143–146.

AGARWAL, R.; FERRATT, T. W. Retention and the career motives of it professionals. In: *Proceedings of the 2000 ACM SIGCPR conference on Computer personnel research*. [S.l.: s.n.], 2000. p. 158–166.

AMORIM, L. F.; MARINHO, M.; SAMPAIO, S. How (un) happiness impacts on software engineers in agile teams? *arXiv preprint arXiv:2006.03546*, 2020.

ANG, S.; SLAUGHTER, S. Turnover of information technology professionals: the effects of internal labor market strategies. *ACM SIGMIS Database: The DATABASE for Advances in Information Systems*, ACM New York, NY, USA, v. 35, n. 3, p. 11–27, 2004.

ARAMO-IMMONEN, H.; JAAKKOLA, H.; LINNA, P. Trust building in globalized software engineering: A cultural perspective. *Journal of Global Information Technology Management*, Taylor & Francis, v. 14, n. 4, p. 28–47, 2011.

ARTHUR, M. B.; ROUSSEAU, D. M. *The boundaryless career: A new employment principle for a new organizational era*. [S.l.]: Oxford University Press, USA, 2001.

BAO, L.; XING, Z.; XIA, X.; LO, D.; LI, S. Who will leave the company?: a large-scale industry study of developer turnover by mining monthly work report. In: IEEE. *2017 IEEE/ACM 14th International Conference on Mining Software Repositories (MSR)*. [S.l.], 2017. p. 170–181.

BASS, J. M.; SARAH, B.; RAZZAK, M. A.; NOLL, J. Employee retention and turnover in global software development: Comparing in-house offshoring and offshore outsourcing. In: IEEE. *2018 IEEE/ACM 13th International Conference on Global Software Engineering (ICGSE)*. [S.l.], 2018. p. 77–86.

BEECHAM, S. Motivating software engineers working in virtual teams across the globe. In: *Software project management in a changing world*. [S.l.]: Springer, 2014. p. 247–273.

BEECHAM, S.; BADDOO, N.; HALL, T.; ROBINSON, H.; SHARP, H. Motivation in software engineering: A systematic literature review. *Information and software technology*, Elsevier, v. 50, n. 9-10, p. 860–878, 2008.

BEECHAM, S.; NOLL, J. What motivates software engineers working in global software development? In: SPRINGER. *International conference on product-focused software process improvement*. [S.l.], 2015. p. 193–209.

BELLINI, C. G. P.; PALVIA, P.; MORENO, V.; JACKS, T.; GRAEML, A. Should i stay or should i go? a study of it professionals during a national crisis. *Information Technology & People*, Emerald Publishing Limited, 2019.

BURNS, M. B.; COLLINS, R. W. The "new" contract between is employees and organizations: workplace and individual factors. In: *Proceedings of the 1998 ACM SIGCPR conference on Computer personnel research*. [S.l.: s.n.], 1998. p. 180–183.

BURT, R. S. *Structural holes: The social structure of competition*. [S.l.]: Harvard university press, 1995.

CAMPBELL, J.; BOELL, S.; KEATING, B.; CECEZ-KECMANOVIC, D. Temporal aspects of telework and its impact on work-family conflict. 2013.

CAMPION, M. A. Interdisciplinary approaches to job design: A constructive replication with extensions. *Journal of applied psychology*, American Psychological Association, v. 73, n. 3, p. 467, 1988.

CHATZIPETROU, P.; ŠMITE, D.; SOLINGEN, R. V. When and who leaves matters: emerging results from an empirical study of employee turnover. In: *Proceedings of the 12th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*. [S.l.: s.n.], 2018. p. 1–4.

CHAVES, A.; AMORIM, L.; MARINHO, M.; JÚNIOR, I. de F.; MOURA, H. Autonomy and turnover in distributed software development projects: a systematic literature review. In: SBC. *Anais do XXV Congresso Ibero-Americano em Engenharia de Software*. [S.l.], 2022. p. 105–119.

CHOI, J.; TAUSCZIK, Y. Characteristics of collaboration in the emerging practice of open data analysis. In: *Proceedings of the 2017 ACM conference on computer supported cooperative work and social computing*. [S.l.: s.n.], 2017. p. 835–846.

CLEAR, T. Thinking issues loosening ties: permanently virtual teams and the melting iceberg of relationship. *ACM Inroads*, ACM New York, NY, USA, v. 12, n. 3, p. 6–8, 2021.

COLEMAN GERRY O'CONNOR, R. Using grounded theory to underst software process improvement: A study of irish software product companies. *Information Software Technology*, Elsevier, v. 49, n. 6, p. 654–667, 2007.

COLOMO-PALACIOS, R.; CASADO-LUMBRERAS, C.; SOTO-ACOSTA, P.; MISRA, S.; GARCÍA-PEÑALVO, F. J. Analyzing human resource management practices within the gsd context. *Journal of Global Information Technology Management*, Taylor & Francis, v. 15, n. 3, p. 30–54, 2012.

CONCHÚIR, E. Ó.; OLSSON, H. H.; ÅGERFALK, P. J.; FITZGERALD, B. Benefits of global software development: exploring the unexplored. *Software Process: Improvement and Practice*, Wiley Online Library, v. 14, n. 4, p. 201–212, 2009.

CRUZES, D. S.; DYBA, T. Recommended steps for thematic synthesis in software engineering. In: IEEE. *2011 international symposium on empirical software engineering and measurement*. [S.l.], 2011. p. 275–284.

DANIEL, S.; JANANSEFAT, S.; DIAMANT, E. I.; REN, Y. Single-and double-loop learning: linking free/libre open source software (floss) developer motivation, contribution, and turnover intentions. *ACM SIGMIS Database: the DATABASE for Advances in Information Systems*, ACM New York, NY, USA, v. 51, n. 4, p. 68–92, 2020.

DECI, E. L.; RYAN, R. M. Self-determination theory. sage publications ltd, 2012.

DIKERT, K.; PAASIVAARA, M.; LASSENIUS, C. Challenges and success factors for large-scale agile transformations: A systematic literature review. *Journal of Systems and Software*, Elsevier, v. 119, p. 87–108, 2016.

DYBA, T.; DINGSOYR, T.; HANSSEN, G. K. Applying systematic reviews to diverse study types: An experience report. In: IEEE. *First international symposium on empirical software engineering and measurement (ESEM 2007)*. [S.l.], 2007. p. 225–234.

DYSVIK, A.; KUVAAS, B. Perceived job autonomy and turnover intention: The moderating role of perceived supervisor support. *European Journal of Work and Organizational Psychology*, Taylor & Francis, v. 22, n. 5, p. 563–573, 2013.

EBERT, C.; KUHRMANN, M.; PRIKLADNICKI, R. Global software engineering: Evolution and trends. In: IEEE. *2016 IEEE 11th International Conference on Global Software Engineering (ICGSE)*. [S.l.], 2016. p. 144–153.

FAGERHOLM, F.; IKONEN, M.; KETTUNEN, P.; MÜNCH, J.; ROTO, V.; ABRAHAMSSON, P. Performance alignment work: How software developers experience the continuous adaptation of team performance in lean and agile environments. *Information and Software Technology*, Elsevier, v. 64, p. 132–147, 2015.

FELPS, W.; MITCHELL, T. R.; HEKMAN, D. R.; LEE, T. W.; HOLTOM, B. C.; HARMAN, W. S. Turnover contagion: How coworkers' job embeddedness and job search behaviors influence quitting. *Academy of management journal*, Academy of Management Briarcliff Manor, NY, v. 52, n. 3, p. 545–561, 2009.

FERRATT, T. W.; ENNS, H. G.; PRASAD, J. Satisfaction of it professionals with employment arrangements in traditional and virtual contexts. In: *Proceedings of the 2001 acm sigcpr conference on computer personnel research*. [S.l.: s.n.], 2001. p. 78–86.

FERREIRA, F.; SILVA, L. L.; VALENTE, M. T. Turnover in open-source projects: The case of core developers. In: *Proceedings of the 34th Brazilian Symposium on Software Engineering*. [S.l.: s.n.], 2020. p. 447–456.

FITZGERALD, B.; STOL, K.-J.; MINÖR, S.; COSMO, H. *Scaling a software business: the digitalization journey*. [S.l.]: Springer Nature, 2017.

FOERDERER, J.; KUDE, T.; MITHAS, S.; HEINZL, A. How temporal work styles and product modularity influence software quality and job satisfaction. In: *Proceedings of the 2016 ACM SIGMIS Conference on Computers and People Research*. [S.l.: s.n.], 2016. p. 105–112.

GALLIVAN, M. J. Examining it professionals' adaptation to technological change: The influence of gender and personal attributes. *ACM SIGMIS Database: the DATABASE for Advances in Information Systems*, ACM New York, NY, USA, v. 35, n. 3, p. 28–49, 2004.

GARRISON, G.; WAKEFIELD, R. L.; XU, X.; KIM, S. H. Globally distributed teams: The effect of diversity on trust, cohesion and individual performance. *ACM SIGMIS Database: the database for Advances in Information Systems*, ACM New York, NY, USA, v. 41, n. 3, p. 27–48, 2010.

GOPALAKRISHNAN, G. M.; HALGIN, D. S.; BORGATTI, S. P. Voluntary turnover in a distributed work setting: an examination of the role of spatial propinquity and role similarity in project affiliation networks. In: *Proceedings of the 2013 conference on Computer supported cooperative work*. [S.l.: s.n.], 2013. p. 329–340.

GRAZIOTIN, D.; FAGERHOLM, F.; WANG, X.; ABRAHAMSSON, P. What happens when software developers are (un) happy. *Journal of Systems and Software*, Elsevier, v. 140, p. 32–47, 2018.

GUSTAVSSON, T.; BERNTZEN, M.; STRAY, V. Changes to team autonomy in large-scale software development: a multiple case study of scaled agile framework (safe) implementations. Sciencesphere, 2022.

HACKMAN, J. R.; OLDHAM, G. R. Motivation through the design of work: Test of a theory. *Organizational behavior and human performance*, Elsevier, v. 16, n. 2, p. 250–279, 1976.

HALL, T.; BEECHAM, S.; VERNER, J.; WILSON, D. The impact of staff turnover on software projects: the importance of understanding what makes software practitioners tick. In: *Proceedings of the 2008 ACM SIGMIS CPR conference on Computer personnel doctoral consortium and research*. [S.l.: s.n.], 2008. p. 30–39.

HERBSLEB, J. *Global software engineering in the age of GitHub and zoom*. [S.l.]: Wiley Online Library, 2023. e2347 p.

HERBSLEB, J. D.; MOCKUS, A. An empirical study of speed and communication in globally distributed software development. *IEEE Transactions on software engineering*, IEEE, v. 29, n. 6, p. 481–494, 2003.

HERBSLEB, J. D.; MOITRA, D. Global software development. *IEEE software*, IEEE, v. 18, n. 2, p. 16–20, 2001.

HESLIN, P. A. Conceptualizing and evaluating career success. *Journal of Organizational Behavior: The International Journal of Industrial, Occupational and Organizational Psychology and Behavior*, Wiley Online Library, v. 26, n. 2, p. 113–136, 2005.

HOEGL, M.; PARBOTEEAH, P. Autonomy and teamwork in innovative projects. *Human Resource Management: Published in Cooperation with the School of Business Administration, The University of Michigan and in alliance with the Society of Human Resources Management*, Wiley Online Library, v. 45, n. 1, p. 67–79, 2006.

HYNNINEN, P.; PIRI, A.; NIINIMÄKI, T. Off-site commitment and voluntary turnover in gsd projects. In: IEEE. *2010 5th IEEE International Conference on Global Software Engineering*. [S.l.], 2010. p. 145–154.

IGBARIA, M.; GREENHAUS, J. H. Determinants of mis employees' turnover intentions: A structural equation model. *Communications of the ACM*, ACM New York, NY, USA, v. 35, n. 2, p. 34–49, 1992.

IVARSSON, M.; GORSCHEK, T. A method for evaluating rigor and industrial relevance of technology evaluations. *Empirical Software Engineering*, Springer, v. 16, n. 3, p. 365–395, 2011.

JAIN, R.; SUMAN, U. A project management framework for global software development. *ACM SIGSOFT Software Engineering Notes*, ACM New York, NY, USA, v. 43, n. 1, p. 1–10, 2018.

JOHRI, A.; TEO, H. J. Achieving equilibrium through coworking: Work-life balance in floss through multiple spaces and media use. In: *Proceedings of the 14th international symposium on open collaboration*. [S.l.: s.n.], 2018. p. 1–11.

JR, L. P. R.; YOU, S. Are you satisfied yet? shared leadership, individual trust, autonomy, and satisfaction in virtual teams. *Journal of the association for information science and technology*, Wiley Online Library, v. 69, n. 4, p. 503–513, 2018.

KIELY, G.; BUTLER, T.; FINNEGAN, P. Global virtual teams coordination mechanisms: building theory from research in software development. *Behaviour & Information Technology*, Taylor & Francis, v. 41, n. 9, p. 1952–1972, 2022.

KITCHENHAM, B.; CHARTERS, S. *Guidelines for performing systematic literature reviews in software engineering*. [S.l.], 2007.

LACITY, M. C.; IYER, V. V.; RUDRAMUNIYAIAH, P. S. Turnover intentions of indian is professionals. In: *Information Systems Outsourcing*. [S.l.]: Springer, 2009. p. 393–421.

LANKAU, M. J.; SCANDURA, T. A. An investigation of personal learning in mentoring relationships: Content, antecedents, and consequences. *Academy of management Journal*, Academy of Management Briarcliff Manor, NY 10510, v. 45, n. 4, p. 779–790, 2002.

LIN, B.; ROBLES, G.; SEREBRENIK, A. Developer turnover in global, industrial open source projects: Insights from applying survival analysis. In: IEEE. *2017 IEEE 12th International Conference on Global Software Engineering (ICGSE)*. [S.l.], 2017. p. 66–75.

LINAKER, J.; SULAMAN, S. M.; HÖST, M.; MELLO, R. M. de. Guidelines for conducting surveys in software engineering v. 1.1. *Lund University*, v. 50, 2015.

LUNDE, L.-K.; FLØVIK, L.; CHRISTENSEN, J. O.; JOHANNESSEN, H. A.; FINNE, L. B.; JØRGENSEN, I. L.; MOHR, B.; VLEESHOUWERS, J. The relationship between telework from home and employee health: a systematic review. *BMC public health*, Springer, v. 22, n. 1, p. 47, 2022.

LUNDENE, K.; MOHAGHEGHI, P. How autonomy emerges as agile cross-functional teams mature. In: *Proceedings of the 19th International Conference on Agile Software Development: Companion*. [S.l.: s.n.], 2018. p. 1–5.

MARINHO, M.; AMORIM, L.; CAMARA, R.; OLIVEIRA, B. R.; SOBRAL, M.; SAMPAIO, S. Happier and further by going together: The importance of software team behaviour during the covid-19 pandemic. *Technology in society*, Elsevier, v. 67, p. 101799, 2021.

MARINHO, M.; NOLL, J.; BEECHAM, S. Uncertainty management for global software development teams. In: *11th International Conference on the Quality of Information and Communications Technology*. Coimbra, Portugal: IEEE, 2018. p. 238–246.

MARINHO, M.; NOLL, J.; RICHARDSON, I.; BEECHAM, S. Plan-driven approaches are alive and kicking in agile global software development. In: *International Symposium on Empirical Software Engineering and Measurement (ESEM)*. Porto de Galinhas, Brazil: IEEE, 2019. p. 1–11.

MASSONI, T.; GINANI, N.; SILVA, W.; BARROS, Z.; MOURA, G. Relating voluntary turnover with job characteristics, satisfaction and work exhaustion-an initial study with brazilian developers. In: IEEE. *2019 IEEE/ACM 12th International Workshop on Cooperative and Human Aspects of Software Engineering (CHASE)*. [S.l.], 2019. p. 85–88.

MELAND, H.; WAAGE, R. P.; SEIN, M. K. The other side of turnover: Managing it personnel strategically. In: *Proceedings of the 2005 ACM SIGMIS CPR conference on Computer personnel research*. [S.l.: s.n.], 2005. p. 67–74.

MGAYA, K. V.; UZOKA, F.-M. E.; KITINDI, E. G.; SHEMI, A. P. Examining career orientations of information systems personnel in an emerging economy context. In: *Proceedings of the special interest group on management information system's 47th annual conference on Computer personnel research*. [S.l.: s.n.], 2009. p. 41–56.

MIKALSEN, M.; NÆSJE, M.; REIME, E. A.; SOLEM, A. Agile autonomous teams in complex organizations. In: *XP Workshops*. [S.l.: s.n.], 2019. p. 55–63.

MOE, N. B.; CRUZES, D.; DYBÅ, T.; MIKKELSEN, E. Continuous software testing in a globally distributed project. In: IEEE. *2015 IEEE 10th international conference on global software engineering*. [S.l.], 2015. p. 130–134.

MOE, N. B.; DINGSØYR, T.; DYBÅ, T. Understanding self-organizing teams in agile software development. In: IEEE. *19th australian conference on software engineering (aswec 2008)*. [S.l.], 2008. p. 76–85.

MOE, N. B.; ŠMITE, D.; PAASIVAARA, M.; LASSENIUS, C. Finding the sweet spot for organizational control and team autonomy in large-scale agile software development. *Empirical Software Engineering*, Springer, v. 26, n. 5, p. 101, 2021.

MONTEIRO, C. V.; SILVA, F. Q. da; SANTOS, I. R. dos; FARIAS, F.; CARDOZO, E. S.; LEITÃO, A. R. do A.; NETO, D. N.; FILHO, M. J. P. A qualitative study of the determinants of self-managing team effectiveness in a scrum team. In: *Proceedings of the 4th International Workshop on Cooperative and Human Aspects of Software Engineering*. [S.l.: s.n.], 2011. p. 16–23.

MOQUIN, R.; RIEMENSCHNEIDER, C. K. Psychological contract in the information systems profession. In: *Proceedings of the 2013 annual conference on Computers and people research*. [S.l.: s.n.], 2013. p. 191–196.

MORGESON, F. P.; HUMPHREY, S. E. The work design questionnaire (wdq): developing and validating a comprehensive measure for assessing job design and the nature of work. *Journal of applied psychology*, American Psychological Association, v. 91, n. 6, p. 1321, 2006.

MOSSHOLDER, K. W.; SETTOON, R. P.; HENAGAN, S. C. A relational perspective on turnover: Examining structural, attitudinal, and behavioral predictors. *Academy of Management Journal*, Academy of Management Briarcliff Manor, NY 10510, v. 48, n. 4, p. 607–618, 2005.

MOURMANT, G.; NIEDERMAN, F.; KALIKA, M. Spaces of it intrapreneurial freedom: a classic grounded theory. In: *Proceedings of the 2013 annual conference on computers and people research*. [S.l.: s.n.], 2013. p. 33–44.

NETO, U. P.; KOCHHAR, P. S.; GAO, C.; AHMED, I. A deep dive into the impact of covid-19 on software development. *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING*, v. 48, n. 9, 2022.

NOLL, J.; BEECHAM, S.; RAZZAK, A.; RICHARDSON, B.; BARCOMB, A.; RICHARDSON, I. Motivation and autonomy in global software development. In: SPRINGER. *International Workshop on Global Sourcing of Information Technology and Business Processes*. [S.l.], 2017. p. 19–38.

PAYNE, S. C.; HUFFMAN, A. H. A longitudinal examination of the influence of mentoring on organizational commitment and turnover. *Academy of Management journal*, Academy of Management Briarcliff Manor, NY 10510, v. 48, n. 1, p. 158–168, 2005.

PETERSEN, K.; FELDT, R.; MUJTABA, S.; MATTSSON, M. Systematic mapping studies in software engineering. In: *12th International Conference on Evaluation and Assessment in Software Engineering (EASE) 12*. [S.l.: s.n.], 2008. p. 1–10.

PRIKLADNICKI, R.; AUDY, J. L. N.; EVARISTO, R. Global software development in practice lessons learned. *Software Process: Improvement and Practice*, Wiley Online Library, v. 8, n. 4, p. 267–281, 2003.

QUAN, J.; CHA, H. It certifications, outsourcing and information systems personnel turnover. *Information Technology & People*, Emerald Group Publishing Limited, 2010.

RAVN, J. E.; MOE, N. B.; STRAY, V.; SEIM, E. A. Team autonomy and digital transformation: Disruptions and adjustments in a well-established organizational principle. *AI & SOCIETY*, Springer, v. 37, n. 2, p. 701–710, 2022.

REMUS, U.; WIENER, M.; SAUNDERS, C.; MÄHRING, M.; KOFLER, M. Control modes versus control styles: Investigating isd project control effects at the individual level. 2016.

RICHARDSON, I.; CASEY, V.; BURTON, J.; MCCAFFERY, F. Global software engineering: A software process approach. In: *Collaborative software engineering*. [S.l.]: Springer, 2010. p. 35–56.

RUTNER, P. S.; WILLIAMS, F. I.; CAMPBELL, C.; RIEMENSCHNEIDER, C. K. The politics of emotion: Exploring emotional labor and political skill across job types within the it/is profession. *ACM SIGMIS Database: the DATABASE for Advances in Information Systems*, ACM New York, NY, USA, v. 46, n. 3, p. 52–73, 2015.

RYAN, R. M.; DECI, E. L. Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American psychologist*, American Psychological Association, v. 55, n. 1, p. 68, 2000.

SEDANO, T.; RALPH, P.; PÉRAIRE, C. Software development waste. In: IEEE. *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*. [S.l.], 2017. p. 130–140.

SETHI, V.; KING, R. C.; QUICK, J. C. What causes stress in information system professionals? *Communications of the ACM*, ACM New York, NY, USA, v. 47, n. 3, p. 99–102, 2004.

SETOR, T.; JOSEPH, D. When agile means staying: The relationship between agile development usage and individual it professional outcomes. In: *Proceedings of the 2019 on Computers and People Research Conference*. [S.l.: s.n.], 2019. p. 168–175.

ŠMITE, D.; MOE, N. B.; FLORYAN, M.; GONZALEZ-HUERTA, J.; DORNER, M.; SABLIS, A. Decentralized decision-making and scaled autonomy at spotify. *Journal of Systems and Software*, Elsevier, v. 200, p. 111649, 2023.

SMITE, D.; MOE, N. B.; LEVINTA, G.; FLORYAN, M. Spotify guilds: how to succeed with knowledge sharing in large-scale agile organizations. *Ieee Software*, IEEE, v. 36, n. 2, p. 51–57, 2019.

SMITE, D.; TKALICH, A.; MOE, N. B.; PAPATHEOCHAROUS, E.; KLOTINS, E.; BUVIK, M. P. Changes in perceived productivity of software engineers during covid-19 pandemic: The voice of evidence. *Journal of Systems and Software*, Elsevier, v. 186, p. 111197, 2022.

SMITH, D.; SPEIGHT, H. Antecedents of turnover intention and actual turnover among information systems personnel in south africa. In: *Proceedings of the 2006 ACM SIGMIS CPR conference on computer personnel research: Forty four years of computer personnel research: achievements, challenges & the future*. [S.l.: s.n.], 2006. p. 123–129.

STAPLES, D. S.; WEBSTER, J. Exploring the effects of trust, task interdependence and virtualness on knowledge sharing in teams. *Information systems journal*, Wiley Online Library, v. 18, n. 6, p. 617–640, 2008.

STRAUSS A CORBIN, J. Basics of qualitative research. techniques procedures for developing grounded theory pesquisa qualitativa. In: *Técnicas e procedimentos para o desenvolvimento da teoria fundamentada*. [S.l.]: Artmed Bookman, 2008. p. 103–148.

SUÁREZ-ALBANCHEZ, J.; GUTIERREZ-BRONCANO, S.; JIMENEZ-ESTEVEZ, P.; BLAZQUEZ-RESINO, J. J. Organizational support and turnover intention in the spanish it consultancy sector: Role of organizational commitment. *Cogent Social Sciences*, Taylor & Francis, v. 8, n. 1, p. 2051790, 2022.

SUÁREZ, J.; VIZCAÍNO, A. Stress, motivation, and performance in global software engineering. *Journal of Software: Evolution and Process*, Wiley Online Library, p. e2600, 2023.

SULAYMAN, M.; URQUHART, C.; MENDES, E.; SEIDEL, S. Software process improvement success factors for small and medium web companies: A qualitative study. *Information and Software Technology*, Elsevier, v. 54, n. 5, p. 479–500, 2012.

SUMNER, M.; YAGER, S.; FRANKE, D. Career orientation and organizational commitment of it personnel. In: *Proceedings of the 2005 ACM SIGMIS CPR conference on Computer personnel research*. [S.l.: s.n.], 2005. p. 75–80.

TAFTI, A.; MITHAS, S.; KRISHNAN, M. S. Information technology and the autonomy–control duality: toward a theory. *Information Technology and Management*, Springer, v. 8, n. 2, p. 147–166, 2007.

UZOKA, F.-M. E.; MGAYA, K. V.; SHEMI, A. P.; KITINDI, E. G.; AKINNUWESI, B. A. Stay or quit: It personnel turnover in botswana. In: *Proceedings of the 49th SIGMIS annual conference on Computer personnel research*. [S.l.: s.n.], 2011. p. 97–106.

VERNER, J. M.; BABAR, M. A.; CERPA, N.; HALL, T.; BEECHAM, S. Factors that motivate software engineering teams: A four country empirical study. *Journal of Systems and Software*, Elsevier, v. 92, p. 115–127, 2014.

VITHANA, V.; ASIRVATHAM, D.; JOHAR, M. An empirical study on using agile methods in global software development. In: IEEE. *2018 18th International Conference on Advances in ICT for Emerging Regions (ICTer)*. [S.l.], 2018. p. 150–156.

WALL, T. D.; JACKSON, P. R.; DAVIDS, K. Operator work design and robotics system performance: A serendipitous field study. *Journal of applied Psychology*, American Psychological Association, v. 77, n. 3, p. 353, 1992.

WESTER, F. K. krippendorff, content analysis. an introduction to its methodology:, 2005 9780761915447. *Communications*, v. 30, p. 124–126, 2005.

WHITE, M. D.; MARSH, E. E. Content analysis: A flexible methodology. *Library trends*, Johns Hopkins University Press, v. 55, n. 1, p. 22–45, 2006.

WIERINGA, R.; MAIDEN, N.; MEAD, N.; ROLLAND, C. Requirements engineering paper classification and evaluation criteria: a proposal and a discussion. *Requirements engineering*, Springer, v. 11, n. 1, p. 102–107, 2006.

YENER, S.; ARSLAN, A.; KILINÇ, S. The moderating roles of technological self-efficacy and time management in the technostress and employee performance relationship through burnout. *Information Technology & People*, Emerald Publishing Limited, 2020.

ZHOU, N.; GIFFORD, W. M.; RATAKONDA, K.; WESTERWICK, G. H.; ENGEL, C. On the quantification of global team performance and profitability. In: IEEE. *2014 IEEE International Conference on Services Computing*. [S.l.], 2014. p. 378–385.

7 SURVEY QUESTIONNAIRE

Does autonomy impact on software engineers' turnover in distributed software development projects?

Hi, my name is Felipe Cavalcanti. I've been working as a QA Engineer for 6 years, and I am also part of a scientific research group named FREVO (Fostering Research on ManagEment and InnoVatiOn) from UFRPE (Rural Federal University of Pernambuco). As part of my Software Engineering 's Master of Sciences thesis work, I am looking forward to understanding the relationship between distributed software developers' autonomy and turnover.

Therefore, I would like to ask 5 minutes from your valued time to collect some answers through this survey. Your answers will be highly valued for our research and will be considered for results analysis and further studies on this topic.

Moreover, while reading the questions and statements, please, keep always in mind your experiences inside your daily working context with your teammates.

For instance, we require your personal email address that **will remain confidential**, as will your answers. This research goal is **exclusive** for academic matters. By answering the questions and submitting them, you acknowledge us using the answers provided on our study analysis.

Researchers:

Luís Felipe Cavalcanti de Amorim (Software Engineering master's degree candidate - UFRPE)

Marcelo Luiz Monteiro Marinho (Department of Computer Science - UFRPE)

* Indica uma pergunta obrigatória

1. E-mail *

About autonomy

On this section we present some questions related to your working daily routine focusing on autonomy. Based on your experience and on the team involved on the software development cycle, answer the questions and classify the statements according to the following metrics using grades from 1 to 5 where:

1. Totally disagree
2. Partially disagree
3. I don't know
4. Partially agree
5. Totally agree

2. **Your work is strongly monitored by the company or any other stakeholder.** *

Marcar apenas uma oval.

	Totally disagree
1	◯
2	◯
3	◯
4	◯
5	◯
	Totally agree

3. **Your team has a shared purpose and clear goal, the necessary skills, and mutual trust among the team members.** *

Marcar apenas uma oval.

	Totally disagree
1	
2	
3	
4	
5	
	Totally agree

4. **People in management and leadership roles provide good and healthy support to your team and yet give you the freedom to walk in your own shoes.** *

Marcar apenas uma oval.

	Totally disagree					Totally agree
	1	2	3	4	5	

About turnover

On this section we present to you factors that many people consider as important in deciding to leave or to stay at a distributed software project. Based on your experience, answer and pick the options that best suits you.

5. **Have you ever decided to leave a project or job?** *

Marcar apenas uma oval.

- Yes
- No

6. **Based on your experience, which of the following factors do you consider the most important in deciding to stay at a project or company?** *

Marque todas que se aplicam.

- [ ] Balance of tasks along development cycles
- [ ] All team members feeling involved
- [ ] Organizational commitment
- [ ] Connection with co-workers
- [ ] Freedom for decision making
- [ ] Supervisor support
- [ ] Collaborative environment
- [ ] Good communication
- [ ] Growth opportunity
- [ ] Good and active leadership
- [ ] Satisfaction
- [ ] Motivation
- [ ] Employment policies
- [ ] Workplace innovation
- [ ] Work-life balance
- [ ] Payment
- [ ] Clear career orientation

7. **Based on your experience, which of the following factors do you consider the most important in deciding to leave a project or company?** *

Marque todas que se aplicam.

- [ ] Stress
- [ ] Mismatch between expectations and reality
- [ ] Long working hours
- [ ] Lack of freedom for decision making
- [ ] Payment
- [ ] Lack of motivation
- [ ] Lack of satisfaction
- [ ] Lack of supervisor support
- [ ] Lack of connection with co-workers
- [ ] Lack of collaboration among the team
- [ ] Poor communication among the team

8. **Have you ever been to a distributed project where a software engineer left?** *

Marcar apenas uma oval.

- ◯ Yes
- ◯ No

9. **Based on your experience, which of the following project or business outcomes do you consider that mostly happen when a software engineer decides to leave a project?** *

Marque todas que se aplicam.

- ☐ Lower levels of productivity
- ☐ Project success
- ☐ Loss of knowledge and experience
- ☐ Software quality
- ☐ Better conditions and performance for those who remain
- ☐ Economic loss to companies
- ☐ Lack of commitment and mutual trust among peers
- ☐ Fresh and innovative ideas
- ☐ Unsuccess of software project

About autonomy and turnover

On this section we present to you some questions and statements about your perception regarding autonomy and turnover. Based on your experience and on the team involved on software development cycle, answer the following questions and classify the statements according to the following metrics using grades from 1 to 5 where:

1. Totally disagree
2. Partially disagree
3. I don't know
4. Partially agree
5. Totally agree

10. **Do you believe a higher level of autonomy in a distributed project is a key factor for not willing to leave it?** *

Marcar apenas uma oval.

- ◯ Yes
- ◯ No

11. **Could you tell us an experience you had as a young engineer where you had been given some autonomy at your workplace? How did it feel? Did you like it or not? Did you feel like giving up this autonomy and looking for help from a more experienced engineer?**

12. **Could you tell us an experience you had as a more experienced engineer where you had been given some autonomy at your workplace? How did it feel? Did you like it or not? Did you feel like giving up this autonomy and looking for help from another engineer?**

13. **When I have the freedom to work as preferred, I feel more motivated, satisfied, and willing to stay longer at a project or company.** *

Marcar apenas uma oval.

	Totally disagree
1	
2	
3	
4	
5	
	Totally agree

14. **When I don't know anything about the task assigned to me, I feel stressed and start to think about giving up. If this becomes frequent, I tend to leave the project or company.** *

Marcar apenas uma oval.

Totally disagree	
1	◯
2	◯
3	◯
4	◯
5	◯
Totally agree	

15. **When my team and I have freedom to work as preferred, we also have better control of tasks since the refinement phase through planning and execution. This process leads to quality improvements and connections with other teammates.** *

Marcar apenas uma oval.

Totally disagree

1 ◯

2 ◯

3 ◯

4 ◯

5 ◯

Totally agree

16. **When I am free to choose the tasks I want to work with I usually seek the ones that will challenge me the most and during the development of it I feel delighted.** *

Marcar apenas uma oval.

Totally disagree

1 ◯

2 ◯

3 ◯

4 ◯

5 ◯

Totally agree

17. **When I take care of some complex tasks by myself I feel like someone that everyone can count on and I feel committed to the team's goals. This makes me more engaged.** *

Marcar apenas uma oval.

	Totally disagree
1	
2	
3	
4	
5	
	Totally agree

18. **I'd rather work from home or at a quiet place where there's only me and my computer when I take care of some complex tasks by myself and this gets me less exhausted at the end of the day.** *

Marcar apenas uma oval.

Totally disagree

1 ◯

2 ◯

3 ◯

4 ◯

5 ◯

Totally agree

19. **If I could choose the tasks I would like to work with I'd rather pick the ones related to new products or ideas than the ones related to support or legacy items. This would make me feel more involved and willing to stay longer at a project or company.** *

Marcar apenas uma oval.

	Totally disagree
1	◯
2	◯
3	◯
4	◯
5	◯
	Totally agree

Demography

On this section you will answer some questions about you that will help us to identify and to better categorize the extracted data from this form. It's quick and easy!

20. **What 's your gender?** *

Marcar apenas uma oval.

- ◯ Male
- ◯ Female
- ◯ I'd rather not to say

21. **You work on... ***

Marcar apenas uma oval.

- For-profit organization
- Non-profit organization
- Government institutions
- Academic institutions

22. **How old are you? (only numbers) ***

_____

23. **What's your role in the software team you are a part of? ***

Marcar apenas uma oval.

- Analyst / Requirements engineer
- Software Architect
- Software Developer
- Product Manager/Owner
- Project/Team Manager
- Quality Manager
- Scrum Master/Agile Coach
- Tester
- Trainer
- UX/UI
- Project Leader

24. **What's your company size?** *

Marcar apenas uma oval.

- ◯ Micro company (up to 9 employees)
- ◯ Small company (between 10 and 49 employees)
- ◯ Medium company (between 50 and 99 employees)
- ◯ Large company (100 or more employees)

25. **How distributed is the project you are working on in your company?** *

Marcar apenas uma oval.

- ◯ Locally distributed (people working remotely from the same city)
- ◯ Nationally distributed (people working remotely from cities of a unique country)
- ◯ Continentally distributed (people working remotely from different countries of the same continent)
- ◯ Globally distributed (people working remotely from different countries but not on the same continent)

26. **What's your current education level? (consider the most recent completed)** *

Marcar apenas uma oval.

- ◯ High School degree
- ◯ Technical degree
- ◯ Bachelor's degree
- ◯ Master's degree
- ◯ Doctorate degree

27. **How long have you been working in the software market?** *

Marcar apenas uma oval.

- ◯ Less than a year
- ◯ Between 1 year and 5 years
- ◯ Between 6 and 10 years
- ◯ Between 10 and 15 years
- ◯ More than 15 years

You just have to submit yours answers now!

Don't forget to click on Submit button at the end of this page to send your answers to us!

Thanks for contributing with this research. Your answers were extremely important to us and they will be carefully analysed.

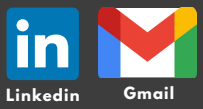
We wish you all the best in your life and work.

8 EXECUTIVE SUMMARY

Autonomy and turnover in distributed teams.

PROJECT BY :
LUIS FELIPE CAVALCANTI DE AMORIM

INTERACTIVE ICONS



Main Concepts

Autonomy

Could be defined from different perspectives, such as individual autonomy, internal and external autonomy, where external autonomy is defined as the influence of management and other individuals outside the team on the team's activities. Internal 19 autonomy refers to the degree to which all team members jointly share decision authority, while individual autonomy refers to the freedom and discretion an individual has in carrying out assigned tasks (MOE et al., 2021).

Turnover

Turnover may be classified as external when team members leave the organization or internal when they remain in the company but change their previous work. This process can happen voluntarily when the employee voluntarily decides to abandon the company and their role or involuntarily, which happens when the organization decides to terminate its relationship with the employee (CHATZIPETROU; ŠMITE; SOLINGEN, 2018).

Global Software Development

A global project is a group of people distributed in different locations that work united in a single project for an extended period. This kind of software project in which the human resources involved are spread by distance, regionally, nationally or globally, is defined as Global Software Development (GSD) (SULAYMAN et al., 2012; MARINHO; NOLL; BEECHAM, 2018; MARINHO et al., 2019).

SLR

What factors contribute to autonomy in GSD teams ?

Mutual trust among team members		
Decrease monitoring levels	Cross-functionality	Work Routine
Work Regime	Career Satisfaction	Job opportunities
Management/Leadership competence and support		

What factors contribute to turnover in GSD teams?

Mismatch between expectations and reality		
Stress	Long working hours	Payment
Autonomy	Lack of motivation	Lack of satisfaction
Poor communication among the team	Lack of connection with co-workers	Lack of collaboration among the team
Lack of supervisor and management support		

What factors mitigate turnover in GSD teams?

All team members feeling involved		
Clear career orientation	Organizational commitment	Connection with co-workers
Autonomy	Supervisor support	Collaborative environment
Good communication	Growth opportunity	Good and active leadership
Satisfaction	Motivation	Employment policies
Workplace innovation	Work-life balance	Payment
Balance of tasks along development cycles		

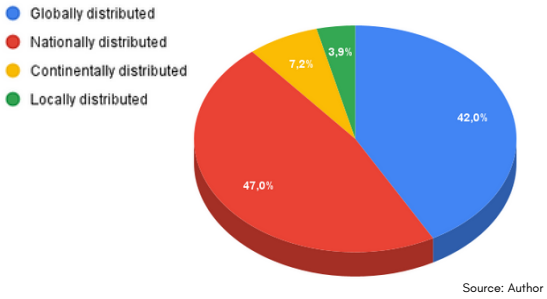
What impact does autonomy have on turnover within GSD teams?

Employees' wishes	Career Satisfaction
Communication	Job opportunities
Contributor's opinion	Work Regime
Individual factors	Work Routine

Survey

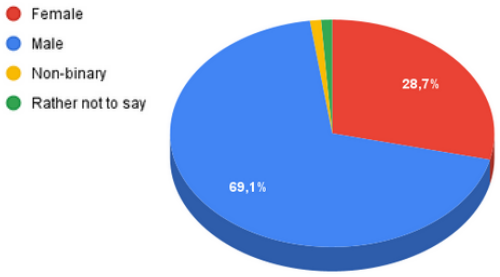
181 Participants

Participants' team distribution



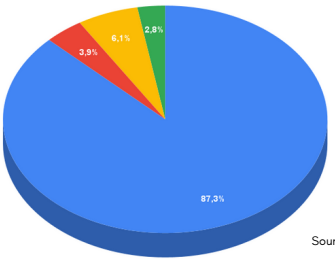
Source: Author

Participants gender



Source: Author

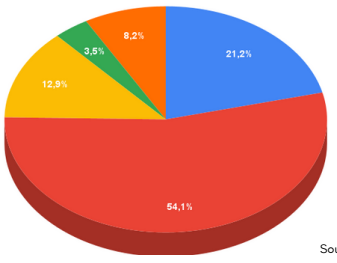
Participants' companies size



Source: Author

Large company (100 or more employees)
Small company (From 10 to 49 employees)
Medium company (between 50 and 99 employees)
Micro company (up to 9 employees)

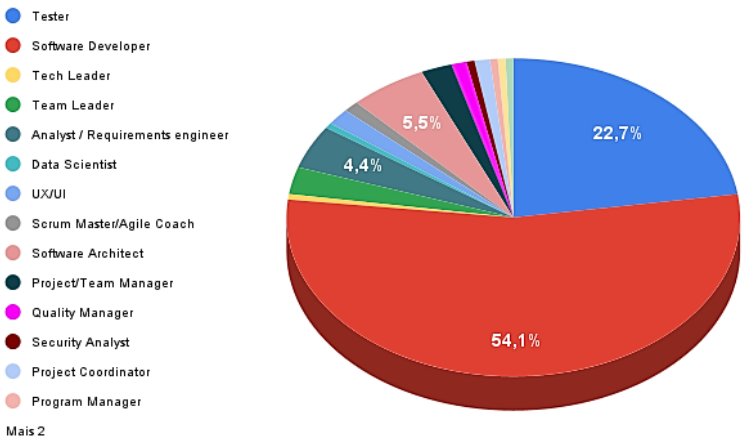
Participants experience level



Source: Author

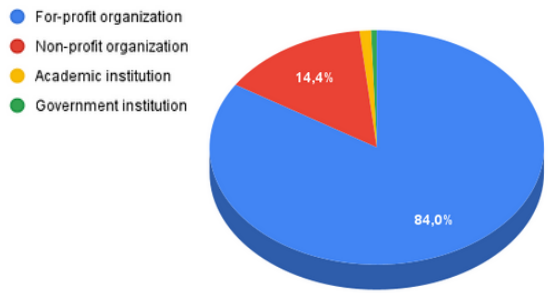
Between 6 and 10 years
Between 1 year and 5 years
Between 10 and 15 years
Less than 1 year
More than 15 years

Participants roles in SDLC



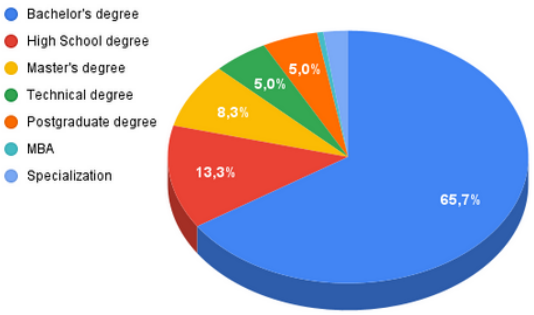
Source: Author

Participants organization type



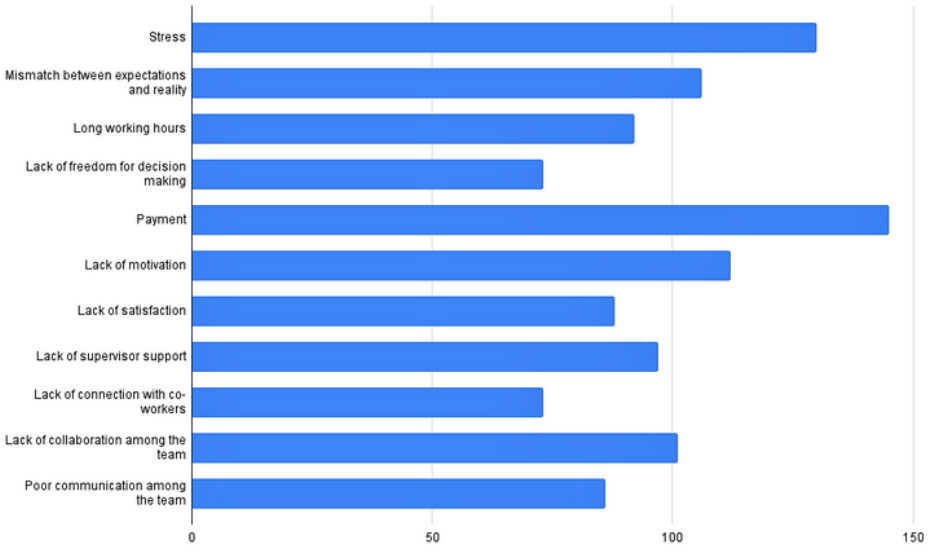
Source: Author

Participants education leveL

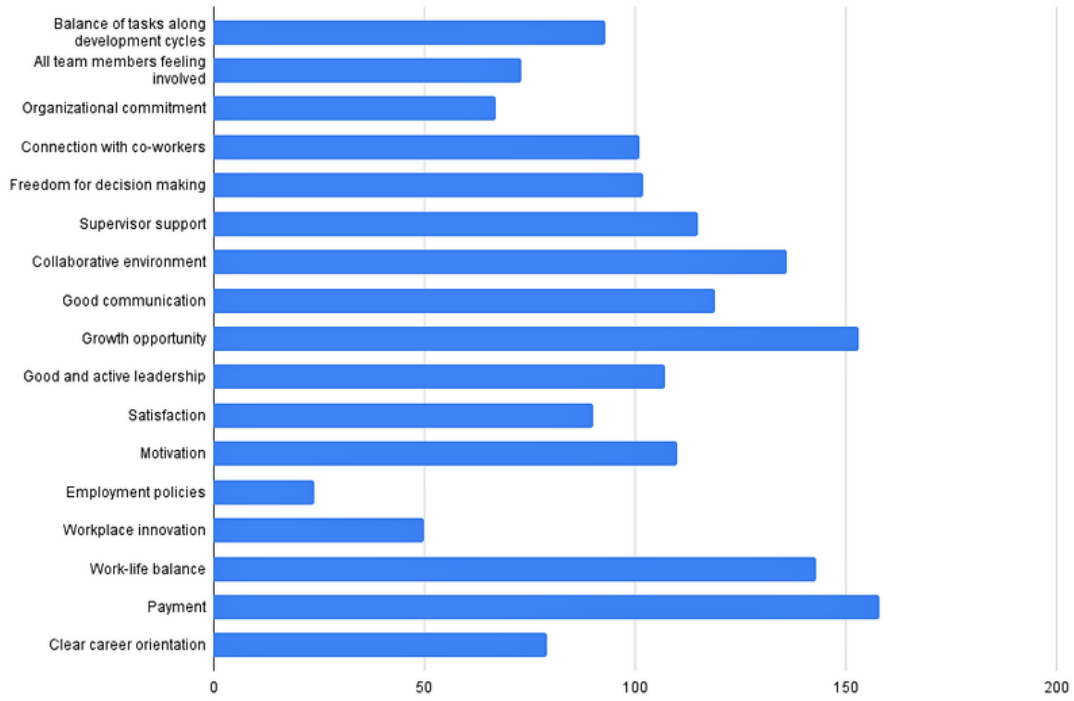


Source: Author

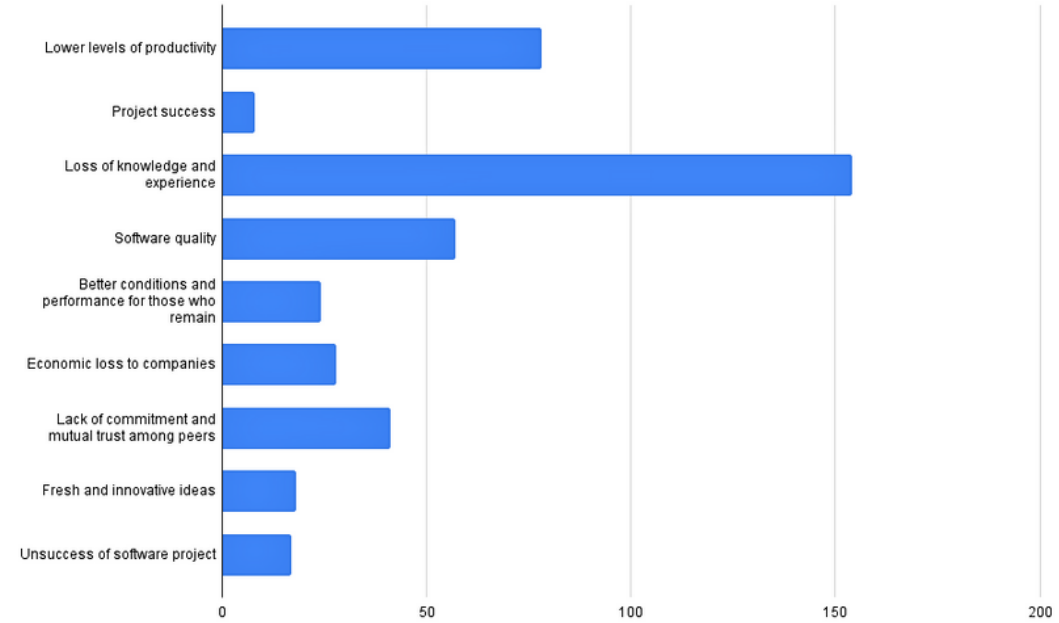
Most important factors to leave a project or company according to participants



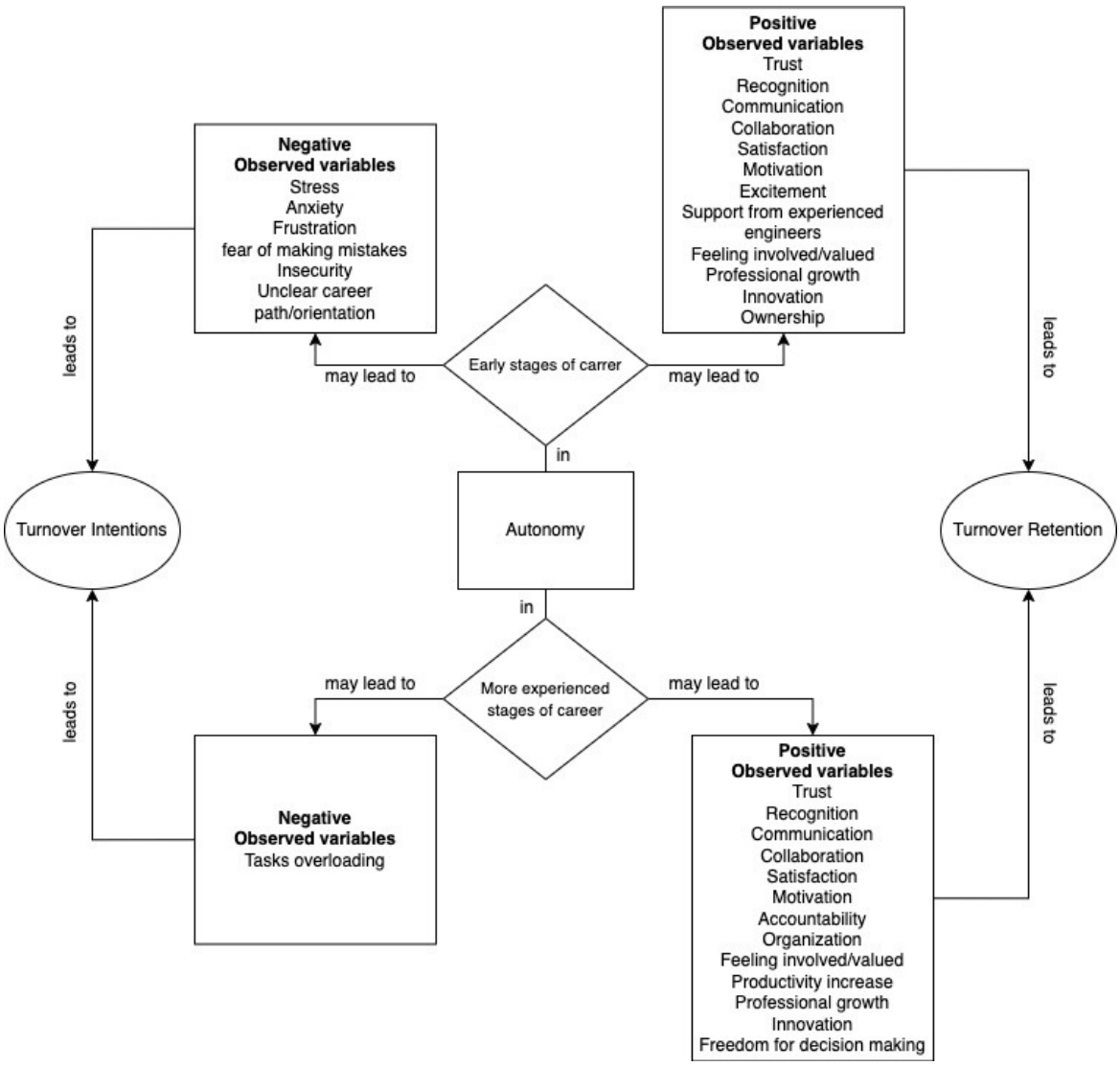
Most important factors to staying in a project or company according to participants



Most important outcomes after a software engineer leaves a project or company according to participants



Perceived outcomes raised from autonomy

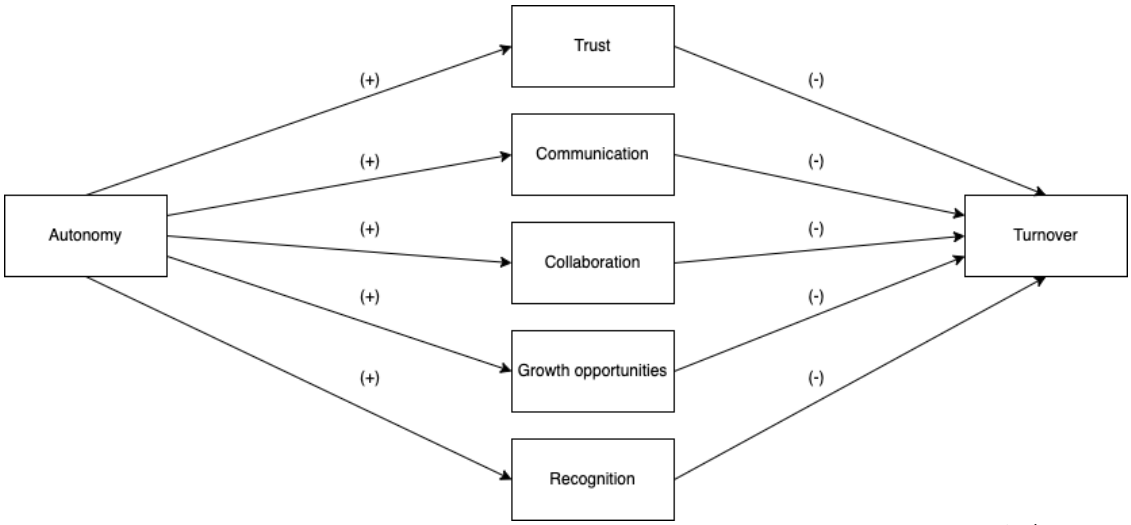


Autonomy Impact on Turnover

- ***Lesson 1: Autonomy is positively related to trust which is negatively related to turnover***

- ***Lesson 2: Simply having autonomy is not enough to retain software engineers***

Autonomy impact on turnover and their main related factors



Source: Author

- ***Lesson 3: Autonomy is positively related to communication and collaboration, which is negatively related to turnover.***

- ***Lesson 4: Autonomy is linked to professional growth opportunities***

- ***Lesson 5: Autonomy is positively related to recognition which is negatively related to turnover***

9 SYSTEMATIC LITERATURE REVIEW PROTOCOL

Context:

Turnover causes various changes in the human resources of companies and software projects [1]. Studies indicate that there are different factors linked either positively or negatively to the occurrence of turnover [2]. Since 1979, literature has been developing models attempting to relate specific factors to turnover. For example, one can mention Mobley's model, which devised a model for understanding motivation at an individual level of user behavior [3].

Software companies experience high levels of turnover [1], and this constant personnel change and turnover create various problems, such as increased costs for the company, difficulty in managing teams, decreased workplace harmony, and an impact on project success, among others [4]. For instance, a study conducted by the University of California pointed out that approximately 90% of projects suffer from turnover. They used a Constitutive Cost model and analyzed 16 organizations for this purpose [5].

Objective:

A high level of turnover becomes problematic because it indicates that professionals do not stay with the company for long, leading to low retention rates in companies and projects. Turnover results in significant costs to the project, whether they are monetary in nature or impact the quality of work and individual performance [6].

Models present both positive and negative factors that contribute to turnover, differentiating between actual turnover, when the professional has already left the company and the project, and turnover intentions, where you can perceive the professional's intentions while they are still part of the project [7].

One of these models recognizes a discrepancy in individual needs of professionals and distinguishes between Job Satisfaction and Career Satisfaction. Job Satisfaction is related to contentment with the current job and is influenced by what happens in the workplace. Job satisfaction is a part of career satisfaction, which is associated with a broader scope, taking into account external factors and aspirations [8].

The literature points out several factors that impact turnover and turnover intentions [7,8]. Autonomy is an important and motivating factor for software engineers in this context. A misalignment between the autonomy needs of software engineers and the level of autonomy they have can affect their performance and motivation [2]. Therefore, our focus will be on the relationship between autonomy and turnover among software engineers involved in distributed software projects.

Table 1 - Definition of factors related to turnover and turnover intentions

FACTOR	DEFINITION(S)
Autonomy	● A practice or set of practices involving the delegation of responsibility down the hierarchy so as to give employees increased decision-making authority with respect to the execution of their primary work tasks; ● job control; ● schedule control [10].

A distributed project is characterized as a group of people in different locations working together on a common project for an extended period of time. This type of software project in which the human resources involved are geographically dispersed, whether at a local, national, or global level, is referred to as Distributed Software Development (DSD) [9].

DSD can be classified based on two factors:

1. The distance between the working teams: Onshore (teams located in the same country) and Offshore (teams located in different countries).
2. The level of control the parent organization has over the remote teams: Outsourcing (hiring a third-party company) or Insourcing (establishing a remote unit of the company) [10].

Research Questions:

RQ1 - How does autonomy occur among software engineers in Distributed Software Development Projects?

RQ1.1 - What are the barriers to autonomy among software engineers in Distributed Software Development Projects?

RQ1.2 - What are the benefits of autonomy among software engineers in Distributed Software Development Projects?

RQ2 - What factors determine the turnover of software engineers in Distributed Software Development Projects?

RQ2.1 - What measures are taken to reduce the turnover of software engineers in Distributed Software Development Projects?

RQ3 - What is the relationship between Autonomy and Turnover in Distributed Software Development Projects?

RQ3.1 - How does the autonomy of software engineers impact turnover in Distributed Software Development Projects?
RQ3.2 - How does the turnover of software engineers impact autonomy in Distributed Software Development Projects?

Search Strategy:

Our search will focus on combining terms in the most effective way to maximize results that are most relevant to our research.

Search String:

To achieve this, it will be necessary to create a combination of synonyms, abbreviations, and variations of the main themes. In our case, we are looking for Distributed Software Development, Turnover intention, and autonomy.

Identified terms	Terms to be used
Distributed Software Development	• global software engineering • global software development • distributed software engineering • distributed software development • GSE • GSD • distributed teams • global team • dispersed team • spread team • virtual team • offshore • outsource • DSD • DSE
Turnover intention	• turnover • turnover intention • departure • rate of replacement • employee retention
Autonomy	• Autonomy • self-government • self management • independence • self-rule • freedom • self-sufficiency • job control • schedule control • isolation • job autonomy

Main search String
("global software engineering" OR "global software development" OR "distributed software engineering" OR "distributed software development" OR GSE OR GSD OR "distributed teams" OR "global team" OR "dispersed team" OR "spread team" OR "virtual team" OR offshore OR outsource OR DSD OR DSE) AND (turnover OR "turnover intention" OR departure OR "rate of replacement" OR "employee retention") AND (autonomy OR "self-government" OR independence OR "self-rule" OR freedom OR "self-sufficiency" OR "job control" OR "schedule control" OR "self-management" OR isolation OR "job autonomy")

Upon applying the previous search string to the chosen databases, we achieved satisfactory results in terms of both quantity and quality in most of them. However, it was noticed that the number of results obtained from the IEEE database was significantly lower than expected. To address this issue, we decided to adapt the original search string by grouping all the synonyms for turnover and autonomy into a single search block while retaining the same terms. This adaptation resulted in a more promising quantity of results. Below, you can review the adapted search string.

IEEE String
("global software engineering" OR "global software development" OR "distributed software engineering" OR "distributed software development" OR GSE OR GSD OR "distributed teams" OR "global team" OR "dispersed team" OR "spread team" OR "virtual team" OR offshore OR outsource OR DSD OR DSE) AND (turnover OR "turnover intention" OR departure OR "rate of replacement" OR "employee retention" OR autonomy OR "self-government" OR independence OR "self-rule" OR freedom OR "self-sufficiency" OR "job control" OR "schedule control" OR "self-management" OR isolation OR "job autonomy")

Inclusion and Exclusion Criteria:

Inclusion Criteria:

The following inclusion criteria will be applied to select studies for this research:

1. Studies published in peer-reviewed journals and conferences.
2. Studies directly related to the research questions.
3. Studies in which the keywords from the search string appear in the abstract or in keywords provided by the authors.
4. Studies that are accessible through the library services of the University of Federal University of Pernambuco during the research period or are freely available on the web.
5. Studies related to Distributed Software Development (DSD), turnover, turnover intentions, and/or autonomy.
6. Studies related to the factors of autonomy and turnover in the context of turnover and turnover intentions in DSD.

Exclusion Criteria:

Studies meeting any of the following criteria will be excluded:

1. Studies not written in English.
2. Studies that are books, theories, workshops, technical reports, or experiments, as well as systematic reviews.
3. Studies that primarily present personal viewpoints or expert opinions.
4. Studies that address Distributed Software Development (DDS) but not Turnover.
5. Studies that address Turnover but not DSD.
6. Studies that address Turnover Intentions but not DSD.
7. Studies not related to Software Engineering.
8. Studies that do not address autonomy or turnover.

Note:
In cases where there are multiple publications of the same study in different journals and conferences, all versions will be examined, but only the first published study will be included in the protocol.

REFERENCES

1 Ullah Khan, S., Niazi, M., Ahmad, R.: Critical Success Factors for Offshore Software Development Outsourcing
Vendors: An Empirical Study', in Product-Focused Software Process Improvement' (Springer Berlin/Heidelberg, Germany, 2010, vol. 6156), pp. 146–160

2 Sulayman, M., Mendes, E., Urquhart, C., Riaz, M., Tempero, E.: 'Towards a theoretical framework of SPI success factors for small and medium web companies, Information and Software Technology, 2014, vol. 56, pp. 807–820

3 Goldenson, D. R., Herbsleb, J. D., 'After the Appraisal: A Systematic Survey of Process Improvement, its Benefits, and Factors that Influence Success' (DTIC Document, 1995)

4 Rainer, A., Hall, T.: 'Key success factors for implementing software process improvement: a maturity-based analysis, Journal of systems and software, 2002, 62(2), pp. 71–84

5 Ramasubbu, N.: 'Governing Software Process Improvements in Globally Distributed Product Development, IEEE Transactions on Software Engineering, 2014, vol. 40, pp. 235–250

6 El Emam, K., Koru, A. G.: 'A replicated survey of IT software project failures, IEEE Software, 2008, vol. 25, pp. 84–90

7 Khan, A. W., Khan, S. U.: ‘Critical success factors for offshore software outsourcing contract management from vendors’ perspective: an exploratory study using a systematic literature review, IET software, 2013, vol. 7, pp. 327–338

8 Kautz, K., Nielsen, P. A.: ‘Implementing software process improvement: two cases of technology transfer’. 33rd Annual Hawaii International Conference on System Sciences, 2000, p. 10

9 Sulayman, M., Urquhart, C., Mendes, E., Seidel, S.: ‘Software process improvement success factors for small and
medium Web companies: A qualitative study, Information and Software Technology, 2012, vol. 54, pp. 479–500

10 Richardson, I., Casey, V., Burton, J., McCaffery, F.: ‘Global software engineering: A software process approach: ‘Collaborative Software Engineering’ (Springer Berlin Heidelberg, Germany, 2010), pp. 35–56): Springer.

11 Sulayman, M., Mendes, E.: ‘Quantitative assessments of key success factors in software process improvement for small and medium web companies’. in Proceedings of the ACM Symposium on Applied Computing, 2010.

12 Khan, S. U., Niazi, M., Ahmad, R.: ‘Critical barriers for offshore software development outsourcing vendors: a systematic literature review’. in Asia-Pacific Software Engineering Conference (APSEC), 2009, pp. 79–86

13 Bayona-Oré, S., Calvo-Manzano, J. A., Cuevas, G., San-Feliu, T.: ‘Critical success factors taxonomy for software process deployment, Software Quality Journal, 2014, vol. 22, pp. 21–48