Exploring the Developer Drive

A Case Study on the Motivation of Individual Developers in Diverse Actors to Start and Continue Participation in a PaaS SECO Through the Lens of Self-Determination Theory

By

Caspar R. Verver

in partial fulfilment of the requirements for the degree of

Master of Science

in Management of Technology

at the Delft University of Technology,

to be defended publicly on Wednesday, November 30, 2023 at 10:30 AM.

Graduation committee

First Supervisor & Chairperson:	Dr. ir. G. A. De Reuver	TU Delft
Second supervisor:	Dr. L. J. Kortmann	TU Delft
External supervisors:	L. Bras	Mendix b.v.
	H. van Duuren	Mendix b.v.
Project duration:	July 2023 – November 2023	
Faculty:	Technology, Policy and Management	

This thesis is confidential and cannot be made public until November 30, 2023.

An electronic version of this thesis is available at http://repository.tudelft.nl/

PREFACE

It's an incredible moment to write this preface – the first words you'll encounter as a reader and the final words I'll put down for this thesis. A few months ago, I couldn't have thought of the complete and finished work that now lies before you. With immense pride, I present this thesis as my graduation project, marking the fulfilment of the Master's programme in Management of Technology, and, consequently, the end of my academic journey.

Working at Mendix, immersing myself in the company culture, connecting with its people, and diving into its vibrant ecosystem of dedicated developers has been an incredible experience. Going forward, I know what I hold important in any future employer, as Mendix has been a truly inspiring workplace—compliments and many thanks to all the people who cultivate this space of creativity, diversity and enthusiasm. To my manager Leo, you have been an excellent and inspiring manager. Thank you for being confident in my abilities and your guidance throughout the internship. Also, I want to thank Hidde, who helped me a lot and guided me through challenging moments. And, not to be forgotten, my flatmate Bas, for introducing me to Mendix. Without you, this internship would not have been possible.

I'd also like to thank my academic supervisors, Dr. Ir. Mark de Reuver and Dr. Rens Kortmann. To Mark, your experience and valuable insights were of huge help throughout this process. Your guidance pushed me to give my best in this thesis, enhancing my research skills and encouraging me to constantly reflect critically on my work. I sincerely appreciate your contributions. Rens, your fresh perspectives at critical moments such as the start, midterm and final stage helped me to shape this thesis into its best form. Also, thank you both for your understanding during the final stage of this research project.

Furthermore, this thesis would not have been what it is now without the support of family and friends. Above all, I want to express my heartfelt appreciation to Liese, who has been my unshakeable pillar on this research journey, providing endless support. Words simply fall short of describing the depth of my appreciation of you. Also, my family has been of great support to me. To my parents, Rudolf and Corine, you have been incredibly supportive. Throughout my academic career, you have seen me struggle and overcome challenges. I know you are very proud of confidently achieving this last step in my academic journey. To my amazing sister, Cato, your critical insights on my thesis and confidence in me mean more than any big brother could wish for.

I am grateful to my flatmates, Cas and Bas, for their motivating pep talks and support when I needed it the most. I also acknowledge the support and friendship of my fellow students and friends, Ben, Micha, Casper, and all those who aided me along the way.

Lastly, I am deeply thankful to all the participants in my research. Their contributions and honesty have enriched the depth and quality of the thesis you now read. Their passion for Mendix inspires me to attain a career that ignites the same enthusiasm and joy.

This journey has been a remarkable chapter in my life. As I reflect, I am reminded of my growth and stand ready to take on the challenges of the next chapter. Hence, I approach the future with curiosity and excitement, eager to embrace its opportunities.

Caspar Verver
Delft, November 2023

EXECUTIVE SUMMARY

Research Objective & Methodology

This thesis addresses the motivation of developers to start and continue participation in a Platform-as-a-Service Software Ecosystem (PaaS, SECOs). In a SECO, diverse actors collaborate and compete within a shared market that is built on the existence of an underlying platform. PaaS is gaining importance in the SECO context due to its cost advantages and scalability. As demand for digitalisation is rising, competition among PaaS providers intensifies. To compete, the PaaS provider must keep the SECO healthy, as this is critical in fostering the growth and longevity of the SECO. In a PaaS SECOs health, the number of developers and the diversity of types of actors play crucial roles. Hence, understanding what motivates developers in diverse actors to participate is critical for PaaS providers to draw them into their SECO and keep them there. This thesis researches the motivators of individual developers in these diverse actors to start and continue participation in the PaaS SECO and explores any differences in motivation between developers.

Due to the scarcity of literature on the motivations of developers, the exploratory case study approach was selected. The Mendix SECO, a PaaS for low-code enterprise application development, is the subject of this case study. Given Mendix's success, the vast and growing developing community of the SECO and the growing importance of low-code within the digitalisation of industries, this makes a relevant and unique case. Through semi-structured interviews, developers from the following actors were investigated: Independent Software Vendors (ISVs), service providers and customers. Subsequently, Self-Determination Theory, a theory on motivation, was used to structure and understand the nature of the motivators of developers. This theory assists in structuring the motivators by distinguishing motivators over the full range of motivation types beyond the intrinsic-extrinsic binary scale, and includes the sub-types of extrinsic motivation and the controlled-autonomous types of motivation. Additionally, the underlying drivers of these motivators could be established by analysing the motivators through the basic psychological needs for autonomy, competence and relatedness. Lastly, this research aimed to find differences in motivation between various developer groups. It compared the motivations of developers between actor types and that of developers with and without high-code background.

Findings

The analysis emphasises the importance of intrinsic motivators over extrinsic ones. The most important intrinsic motivators for developers to start in a PaaS SECO are fun throughout the development process, intellectual stimulation, the ability to create apps fast and agile, the acquisition and improvement of skills, and the ability to contribute to something new. To start, developers are primarily motivated by the need for competence, which manifests in challenge and creativity, which, in turn, is part of experienced fun. Extrinsic motivators to start participating are the exchange of knowledge between developers and the size and quality of the developer community. The support offered by the SECO and its developer community is closely linked to the extrinsic motivators because it reduces the complications that could impede a developer's learning curve and progression.

In the continued participation in the SECO, intrinsic motivators remain crucial. The maturing of skills of the developer increases confidence and results in additional importance of fun.

Learning skills takes on new form of staying up to date with the continuous platform innovations. However, the learning of skills may also be enforced by external pressures as the continuous learning of skills is obligated by developers' employers. Additionally, the developer becomes more aware of what the SECO and its developer community offer, beyond support. is important, developers The social events are mostly viewed as a perk rather than a crucial motivation, but can form an additional source of motivation as this allows developers to establish connections with peers and get inspired. While not many developers identify with the developer community, physical events form the starting point for thos who do. Furthermore, increased emphasis is put on the value the developer creates with its development work and the ability to self-manage his work through agile practices.

Extrinsic motivators, predominantly in autonomous form, gain significance in the continuation. Developers are more attuned to the platform's strategic direction as they want to be part of a successful SECO. Satisfaction with the SECO is emphasised and mainly manifests in the PaaS providers efforts to innovate the SECO. Perceptions of the PaaS providers' reputation, combined with the satisfaction of the developer with the SECO, can deepen their motivation for continued participation or lead to separation. The presence of the PaaS provider in the social events and community forum can boost motivation as it positively influences developers.

These findings shed light on developer motivations and reveal overarching similarities and differences between low-code and other SECO types, raising questions about low-code developers' motivations compared to their peers in other SECOs. For example, the dominence of intrinsic motivators is consistent with previous research in the Mobile Ecosystem (e.g. Android or iOS). However, a key difference is that this study finds the ability to create apps fast and agile as a new motivator that refers to the developers' wants to avoid complexities and bottlenecks, such as database and cybersecurity-related designs, bug solving, and fast steering on changing requirements, which do not play into the intellectual challenge developers seek. This raises the question if this same motivator is equally relevant in other SECOs. Another key difference is the role of the community, as Mobile SECO developers show to be more sensitive to developing a strong identifications with the developer community, resulting in active open-source contributions and seeking a reputation in the developer community. Which variables are responsible for these differences is another question this raises.

Because actors were categorised based on their SECO activity, while in reality, actor activities overlap, the strict comparison of developers between actors was obscured. Nonetheless, based on preliminary findings, it was found that service providers are relatively considered more extrinsically motivated due to their work for external clients. External pressures partly enforce learning skills and creating value, whereas developers of customers are more intrinsically motivated as the connection to their organisation intrinsically motivates them to help their organisation forward with their work. However, all developers are primarily intrinsically motivating the significance of intrinsic motivators, making it challenging to pinpoint substantial differences in motivation.

High-code developers are initially attracted to low-code for its agility and simplified development processes. Still, they may find it lacking intellectual challenges over time, leading to diminished competence fulfilment. Frustration with a perceived loss of autonomy is caused by a lack of transparency in the low-code framework. The desire for a hybrid solution combining both low-code and high-code emerges among high-code developers to fulfil both

needs. In contrast, developers without high-code backgrounds may face initial challenges in transitioning to low-code but maintain intellectual stimulation over time. They find satisfaction in analytical and business aspects, with low-code presenting sufficient long-term technical challenges.

Theoretical contributions

This study builds on existing evidence of the dominant role of intrinsic motivators over extrinsic ones in proprietary SECOs by exploring the topic of developer motivation in the unexplored context of PaaS SECOs. It enriches the current understanding of developer motivation as the architecture configuration, in terms of actor types and underlying technology of PaaS and low-code PaaS, significantly differs from SECOs explored in previous studies.

The adoption of SDT reveals nuances in motivators, questioning previous assumptions and categorisations made by scholars. These nuances led to a restructuring of motivators along the dimensions of intrinsic and extrinsic motivation. As a result, the findings in this thesis ground the existing understandings of the dominant role of intrinsic motivators over extrinsic ones in theory and so strengthen them.

The application of SDT enhances the understanding of motivators, offering more precision and explanation for future research on motivation-related variables. As the majority of the extrinsic motivators are autonomous, this thesis also suggests re-evaluating the intrinsic-extrinsic binary scale, particularly as it adds substance relevant for a deeper, more precise understanding. Further detailing of this scale will allow future research to enhance the meaning and precision of findings and contribute to a more robust understanding of developer motivation. It lays a foundation for future studies, enabling a theoretical exploration of causal relationships and better strategies for motivation enhancement.

Moreover, the study suggests that recognising differences in motivation among various categories of developers can resolve the scholarly perception of SECO developers as a homogeneous group. By exploring differences in motivation along the roles and experiences of developers, the research provides a fresh perspective on motivation within SECOs. The preliminary results emphasise the significance of understanding this crucial aspect of developer behaviour and lay the groundwork for future research.

Practical implications

In the start of participation in the SECO, intrinsic motivators linked to competence needs are crucial, emphasizing that the platform should be perceived as intellectually challenging, and conducive to creativity. These competence-related morivational factors enhance the developers' perceived fun using the platform. Based on the findings, PaaS providers should also mitigate complexities of development to make the developers perception of developing in the SECO as fast and agile.

Using Self-Determination Theory (SDT) to understand motivation can guide practitioners in refining strategies to attract and retain developers. Notably, extrinsic rewards may be counterproductive for stimulating autonomously motivated developers, potentially decreasing their initial automous motivation. In cases of low or controlled motivation, deploying extrinsic rewards could be beneficial.

Furthermore, PaaS providers can boost motivation by actively participating in the developer community, interacting with representatives, and collecting insights and feedback. Furthermore, even though the study couldn't determine the exact extent of its influence on feature adoption and platform usage, delivering ecosystem innovations and introducing new features to the developer community is critical.

According to the findings, specific SECO features do not distinguish one actor type from another. Orchestrators should consider conducting quantitative research to investigate these differences further. If no significant differences are discovered, it is clear that the organisational context significantly impacts developers' decisions. This emphasises the significance of understanding and catering to developers' specific needs and preferences in various organisational contexts.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

Abbreviation	Definition
App	Software application
CET	Cognitive Evaluation Theory
FLOSS	Free/Libre Open Source Software
FOSS	Free Open Source Software
GDPR	General Data Protection Regulation
HREC	Human Research Ethics Committee
ISV	Independent Software Vendor
IaaS	Infrastructure-as-a-Service
LCDP	Low-Code Development Platform
MSECO	Mobile phone Software Ecosystem
MoT	Management of Technology
OIT	Organismic Integration Theory
OS	Open Source
OSS	Open Source Software
PaaS	Platform-as-a-Service
SDT	Self-Determination Theory
SECO	Software Ecosystem
SaaS	Software-as-a-Service
VAR	Value-Added-Reseller

1. INTRODUCTION

The first chapter introduces the research problem and explores the existing literature on the problem. Subsequently, the research objective, knowledge gap, and research question(s) are presented. The chapter ends by introducing the research methods used and explaining the relevance to the MSc Management of Technology.

1.1 BACKGROUND

While the concept of ecosystems stems from the field of biology, it has now found its way into the world of business and management. Moore (1999) was the first to introduce 'business ecosystems' as an analogy to describe complex business networks. Literature has taken off from here, and subsets such as Software Ecosystems (SECO) were introduced (Jansen et al., 2009).

Software Ecosystems are complex networks of actors revolving around technologies such as platforms. Jansen et al. (2009, pp. 187–188) define a SECO as "*a set of actors functioning as a unit and interacting with a shared market for software and services, together with the relationships among them. These relationships are frequently underpinned by a common technological platform or market and operate through the exchange of information, resources and artifacts*". This view shifts the lens from the traditional software engineering perspective from a single software-producing company to a group of actors collaborating on top of a platform (Jansen et al., 2009; Manikas & Hansen, 2013). Software Ecosystems provide a home to so called actors of the ecosystem. These are the orchestrator or platform owner, niche players and customers, among others. These niche players settle in the SECO and conduct business on top of the platform by providing services, products and extensions to the platform. The customers are the end-users of the platform (Manikas & Hansen, 2013).

A specific platform gaining increasing relevance is the Platform-as-a-Service (PaaS). In a PaaS, a service provider, also known as the PaaS provider, offers a platform for developing, deploying, and managing running applications (henceforth referred to as apps). This differs from other cloud computing service models, such as Software-as-a-Service (SaaS), where only the software app is provided or Infrastructure-as-a-Service (IaaS), where cloud providers only provide cloud infrastructure (Mell & Grance, 2011). The advantages of PaaS are dramatically lower costs of entry, immediate access to hardware resources and ease of scalability of services, among others (Marston et al., 2011). Gartner (2022) predicts that in 2027 half of all businesses will use industry cloud platforms. Since more and more firms aim for digitalisation, the market for PaaS platforms is growing. As a result, competition is becoming fiercer, and PaaS providers aim for constant growth.

To foster growth, the health of an ecosystem becomes an important theme. Ecosystem health is a method to assess the ecosystem's performance. Similar to a biological ecosystem, the survival of each individual actor within an ecosystem depends on the whole network rather than the individual actor's strength (Iansiti & Levien, 2004). In essence, it is said that if the SECO is healthy, all its participants will thrive. Specifically in the context of PaaS SECOs, developers play a significant role as they are both the biggest contributor to ecosystem health as well as the direct users of the PaaS (Lucassen et al., 2013). As health in this context it is measured in terms of the number of developers it is platform owners to attract and sustain

developers to their SECO and motivate them to participate in the ecosystem actively. At the same time, platform owners need to balance their own strategic goals and commercial success.

To gain new developers, it is essential to understand the motivations of developers to start and continue participation in a PaaS SECO. By understanding the motivators of developers, PaaS providers can actively engage in practices that attract new developers.

1.2 PROBLEM OUTLINE

This section explores the literature on developer motivations and aims to expose any knowledge gap in the state-of-the-art.

1.2.1 Literature Exploration

Self-Determination Theory (SDT) becomes a fundamental theory in search of the motivators of individuals. In their book, Deci & Ryan (2004) explain that motivation is a complex construct that can't be explained as a general and unitary concept. A distinction must be made between intrinsic and extrinsic motivation. Intrinsic motivation occurs when an individual performs behaviour out of interest and enjoyment. In contrast, in the case of extrinsic motivation, behaviour is carried out to pursue contingent outcomes (Deci & Ryan, 2004). Motivation is a complex construct that has received much attention in various disciplines. To understand what drives developers to join and participate in platforms, it is essential to understand how motivation works and how it can be enhanced.

In the existing literature, scholars have dedicated much effort to studying the motivators of developers to join, stay and leave platforms and SECOs, with an emphasis on the third-party developers (e.g. Ceccagnoli et al., 2012; Kude et al., 2012; Koch & Kerschbaum, 2014; van Angeren et al., 2016; Goldbach & Kemper, 2014; Tiwana, 2015). This group of developers complement the underlying technology of a SECO, the platform, by selling extensions on the marketplace. Famous examples are the marketplaces within the iOS and Android ecosystems.

It has been widely established that by joining a SECO, these app developers gain increasing revenue streams due to network effects and economies of scale (Ceccagnoli et al., 2012). However, this does not directly mean that all app developers will instantly join a SECO and start developing. Kude et al. (2011) researched additional motivators of third-party developers in the context of enterprise software and found that resources and capabilities form a significant role in the decision-making to join a platform. These are the ability to provide an integrated platform, the ability to innovate systems, the capability to provide app developers access to broad markets, and the reputation of the software platform. Furthermore, the governance mechanisms, such as the entry barriers and the partnership model the platform owner inflicts on its participating developers, play a significant role. Van Angeren et al. (2016) found that low entry barriers positively affect the SECO's growth and that the partnership model can positively affect the inter-firm collaboration between participating developers. However, these studies are mainly referring to firms. The motivators mentioned here are deciding factors for firms to move to a SECO. To what extent the motivators of developers within these firms drive them to become successful remains unclear.

Goldbach & Kemper (2014) are the first to link platform owner control to the intrinsic motivation of individual developers. By conducting an experiment on what control modes of platform owners provide better motivations for individual third-party developers in the mobile platform marketplace to stay active on the platform. They found that developers' self-

control and high perceived autonomy lead to a higher willingness to stick to a platform and higher intentions to continue developing for that platform. The perceived autonomy of individuals forms a vital link to their intrinsic motivation (Deci & Ryan, 2004).

Koch & Kerschbaum (2014) conducted a survey focusing on the individual developer, including intrinsic and extrinsic motivators for developers to join a platform. In their research, they compared Android and iOS developers. Interestingly, intrinsic motivators such as fun, intellectually stimulating work, and software development skills acquisition are ranked higher than extrinsic motivators such as financial gains. This aligns with the literature on why developers participate in Free and Open-Source Software (FOSS) SECOs (e.g. Ghosh, 2007; Hertel et al., 2003; Lakhani & Wolf, 2007). However, in contrast to OSS, developers in proprietary SECOs primarily strive for commercial success (Manikas & Hansen, 2013).

These studies show that intrinsic and extrinsic motivators are prominent in a developer's choice to join a SECO. Platform owners can play into these motivators through governance mechanisms and openness to attract and keep developers. However, while all these studies research third-party developers, there remains some ambiguity regarding who these developers are. Koch & Kerschbaum (2014) and Goldbach & Kemper (2014) specifically research individual developers, while van Angeren (2016), Kude et al. (2011) and Ceccagnoli et al. (2012) study the developer as a business entity. The latter provides insight into what motivates organisations to move to a SECO. However, to what extent the motivators of individual developers within these actors play a role is not mentioned.

Furthermore, these studies focus solely on third-party developers and do not cover the full range of developers in a SECO. As stated, SECOs are rich and diverse since many types of actors settle in a SECO. Current literature lacks a holistic perspective of the motivators of developers to join SECOs as most scholars focus their efforts on the marketplaces of SECOs. Especially in the case of PaaS, developers play a prominent role. Van Angeren et al. (2016) call for scholars to include these types of members in their research as well. Furthermore, since the efficacy of the enforcing governance mechanisms differs per type of SECO (van Angeren et al., 2016), research is needed to shed light on the motivators of developers to participate in different types of SECOs.

1.2.2 Knowledge Gap
The identified knowledge gaps are summarised here:

- While much research has been dedicated to the motivators of third-party developers to participate in a SECO, the role of the individual developers within these actors and what motivates them is not mentioned.

- Studies focusing on the individual developer only include the developers of one type of SECO actor: the third-party developer. Hence, the current state of literature does not provide insight into how motivations might differ per type of SECO actor developer.

- No studies on the motivators of developers have been done in the context of PaaS. PaaS SECOs are significantly different from other SECOs since the end-users of the platform are also developers. Furthermore, the health of a PaaS SECO is partly expressed in the number of developers.

1.3 RESEARCH OBJECTIVE

This research aims to contribute to the literature body on Software Ecosystems by investigating the motivations of individual developers to join and start participating in a PaaS SECO. In addition, this research aims to include the diverse actors that exist in SECOs. This research is conducted because these actor types' activities significantly differ, which could affect a developer's motivation. Hence, by including a diverse set of actors, a better and more holistic understanding of developer motivation is gained in a relatively unexplored context.

1.4 RESEARCH QUESTIONS

From the knowledge gap and the research objective, the main research question for this research is drawn:

RQ: *"What motivates individual developers in diverse actor types to start and continue participation in a PaaS SECO?"*

In support of answering the main research question, the following sub-questions have been defined:

- *SQ1: What types of actors that actively develop exist in a PaaS SECO?*

The first sub-question aims at dissecting the dynamics within a SECO by understanding the types of actors that exist within a SECO. The goal is to classify the types of actors that develop software in the SECO and understand what the types of development activities they pursue. This first research question lays the foundation for determining which actors must be investigated.

- *SQ2: According to the literature, what are the motivators of developers in a SECO?*

The second sub-question follows up on the first sub-question on related work by gaining insights into what work has been done on developer motivations in the context of SECOs. Understanding what the state-of-the-art is in terms of the motivations of developers in other SECO types will provide insight into potential motivational factors that drive developers in the PaaS SECO.

- *SQ3: How can the construct of motivation be conceptualised?*

Accordingly, the third sub-question aims to determine which theories on motivation exist and how each theory describes motivation to select the best fitting theory for this research. By asking how motivation can be conceptualised. Due to the complexity of this construct, it is a necessary step to undertake in this research. Furthermore, this lays the groundwork for understanding developers' motivations in formulating an answer to the final sub-question.

- *SQ4: What motivates the individual developers in a specific PaaS SECO?*

After defining the construct of motivation and finding potential motivational factors that drive developers in a SECO, the fourth sub-question will focus on the motivators for developers to join a PaaS SECO and how they differ per type of SECO. A specific type of PaaS is selected to conduct the research as the landscape of PaaS is rather broad. Drawing on the results of the prior sub-questions, this last sub-question will help answer the main research question and conclude this research.

1.5 RESEARCH APPROACH AND METHODS

This section covers the research methods selected to answer the aforementioned research questions. The research approach is selected first, followed by an elaboration on the selected research methods per research question. This section provides a brief explanation of the methods. Chapter 5 includes the complete methodology.

1.5.1 Research Approach

The approach selected for this research is the case study approach. This section will explain why the case study has been chosen and what type of case study will be conducted.

Yin describes the case study as "an empirical inquiry that investigates a contemporary phenomenon in-depth within its real-life context" (Yin, 2009, p. 18). It is purposefully meant to study phenomena in their natural environment. Case studies can be used to explain, describe, or explore phenomena. This research approach is chosen as it is desired to understand this phenomenon in a naturalistic setting. Considering the complexity of ecosystems, a real-world setting in which the research is conducted would yield better results. This is an advantage compared to experimental settings as this will help to get an in-depth understanding of the motivators from learning from developers' experiences. Also, the context in which this topic is studied is relevant since it is unsure which or how many variables play a role.

Furthermore, the lack of attention to this topic hints at an exploratory approach. Many variables could be in play, and the case study allows us to study this phenomenon with an exploratory approach. Hence, the context matters. In contrast, an experimental study is mainly used to study the causal relationship between two variables (Crowe et al., 2011). Since the variables are not established and a causal relationship is not to be tested in this study, the experiment has been ruled out.

Due to time and resource restrictions, it was decided to use a single case. The individual developer within the PaaS SECO is the unit of analysis in this study. Therefore, the PaaS SECO is defined as the case in which the developer will be studied. Additionally, devoting all efforts to a single PaaS will contribute to the depth in which the phenomenon can be researched.

1.5.2 The Case – Mendix PaaS SECO

The selected case in this research is the ecosystem of Mendix. Mendix is an application PaaS that provides a platform for developing apps in low-code. (Mendix, n.d.). This case is selected for several reasons:

- Mendix is a successful and leading firm in the low-code segment of the entire PaaS landscape;
- Mendix has a thriving developer community on a global level that consists of at least 300.000 developers;
- Mendix is a strict proprietary SECO, which makes this case all the more interesting as these types of SECOs have received little attention.

An extensive coverage of the case, including background information, can be found in section 5.2.

1.5.3 Research Methods

Now that the research question(s) and the research approach are established, the next step is to detail the research methods. Table 1 provides insight on what research method is deployed per research question.

Table 1: Research methods per research question.

		Main Question	Research Approach	Data Input
		"What motivates individual developers in diverse actor types to start and continue participation in a PaaS SECO?"	Case study	Literature and primary data
		Sub question	Research Methods	
	1	*What types of actors that actively develop exist in a PaaS SECO?*	*Desk research*	*Peer-reviewed literature*
	2	*According to the literature, what are the motivators of developers in a SECO?*	*Literature review*	*Peer-reviewed literature*
	3	*How can the construct of motivation be conceptualised?*	*Desk research*	*Peer-reviewed literature*
	4	*What motivates the individual developers in a specific PaaS SECO?*	*Semi-structured interviews*	*Developers within the selected case ecosystem*

Sub-question 1 – On the SECO actors within a PaaS SECO
There have been many studies on SECOs, including case studies and empirical research on the kinds of actors that can be found in a SECO. Due to the many classifications and types of SECOs, a necessary first step is to cover the literature on actors within a SECO. Literature will provide a complete answer to what types of actors, in general, can exist in a PaaS SECO and how these actors are classified.

Sub-question 4 – On developer motivation in the literature
The literature review in this research phase will be the foundation for understanding developer motivation in the context of PaaS. Furthermore, the deliverables from this phase will also form the foundation for the interviews that will be conducted and, later, theory triangulation. In short, this phase will set out the whole landscape in this field of study by determining the state-of-the-art on the given issues.

Sub-question 3 – On the construct of motivation
The most complete answer to this sub-question is developed by reviewing the extensive body of literature on motivation theory and drawing on scholars' knowledge. Since a plethora of literature is available on the construct of motivation, desk research is the most suitable method for this inquiry. Additionally, drawing on theory of motivations and applying theory to the case study results will enhance the findings' validity.

Sub-question 4 – On developer motivations in practice in the context of PaaS
The fourth sub-question will be answered by conducting interviews. As the selected approach is of an exploratory nature, the semi-structured interview is an appropriate method that facilitates the aim of this study. The strength of the interview in this context is to gain rich and in-depth data on motivation.

The findings and themes from the interviews will be further developed and checked by existing literature to strengthen the results. This will help establish converging lines between the data collected from the case and the existing literature, increasing the research validity (Sekaran & Bougie, 2016; Yin, 2009).

1.6 RELEVANCE TO MSc MANAGEMENT OF TECHNOLOGY

This master's thesis has been conducted as part of the MSc Management of Technology (MoT), a program of the Delft University of Technology. Reporting on a scientific study within a socio-technological context is vital for a MoT student to demonstrate an understanding of the topic and the capacity to "create" scientific contributions based on learning. According to the learning objectives of module MOT2910[1], the master thesis should contain an analytical component, be multidisciplinary in nature, focus on a technical domain, show an understanding of technology as a corporate resource and showcase the correct use of scientific research methods to analyse a problem as put forward in the MOT curriculum.

Analytical component – This thesis focused on the motivators of software developers within a socio-technological context, the SECO. Analytical components can be found in multiple stages throughout the thesis. First, the problem was analysed, resulting in a knowledge gap, a research objective and a planned route towards a possible solution. Subsequently, literature had to be analysed to understand the context of the scientific problem and find a suitable theory to approach motivation. After the data collection phase, the qualitative data was analysed, interpreted and displayed, and the findings' implications were reflected in practice and theory.

Multidisciplinary study – The study focuses on three different disciplines, which are technology-related, business-related and components from the field of psychology. Software Ecosystems are inherently multidisciplinary as they involve a technical, business and social dimension. Moreover, in psychology, the landscape of motivation theory is explored to find the most suitable theoretical lens for this topic. These components are integrated to find the best possible answer to the research question.

Focus on a technical domain – The context in which this study takes place is that of Software Ecosystems. This socio-technological context involves both technological and business-related components. Additionally, MOT students should report on a scientific study in a technological context. This study reflects this by focusing on the social dimension of Software Ecosystems and the motivations of developers to understand what drives developers towards this socio-technological scape.

Technology as a corporate resource – This thesis shows an understanding of technology as a corporate resource by understanding the motivations of developers to join a SECO, and turning this into practical implications and recommendations. The case study involves understanding what drives developers towards this technology. Focusing on the social dimension of SECOs, this socio-technological system is viewed from a business standpoint. On a corporate level, this study provides opportunities to produce implications for the governance strategies of PaaS providers to attract and retain developers.

---

[1] The objectives of module MOT2910 can be found in the online study guide:
https://brightspace.tudelft.nl/d2l/home/596899

Research methods – Several research methods were utilised to develop an answer to the central research question. These were the literature review, the case study, semi-structured interviews and thematic analysis for the analysis of the interview data.

1.7 THESIS OUTLINE

The thesis is structured as follows. First, Chapter 1 introduces the problem statement, objective, and research questions and provides a condensed overview of the research methods used. Chapter 2 exposits the related work on SECOs to provide background information on the topic. This chapter explores the literature to understand what actors can be found in a SECO and how they are classified, followed by a literature review on the motivators of developers within the context of SECOs. Chapter 3 aims to find a suitable theoretical lens for understanding motivation. Chapter 4 synthesises the results found in Chapter 2 and 3 to conclude the literature. Chapter 5 outlines the research methodology. Chapter 6 presents the results of the case study. Next, Chapter 7 discusses the results, implications and proposed future research avenues. Lastly, chapter **Error! Reference source not found.** presents the conclusion and includes recommendations to the orchestrator of the case study.

2 RELATED WORK

This chapter explores the literature on SECO actor types, motivation theory and developer motivation in the context of SECOs. In the first section, the literature on SECO actors is explored. Accordingly, a literature review is produced on developer motivation within SECOs.

2.1 SECO ACTORS, ROLES AND ACTIVITIES

This section explores the literature on which types of actors exist within a SECO and aims to answer the question, '*What types of actors that actively develop exist in a PaaS SECO?*'. First, literature on SECOs is explored to understand how various scholars define actors and activities in the SECO. Then, literature is explored on how scholars classify the types of actors, their roles, and what activities they pursue in the SECO. Lastly, this knowledge is transferred to the context of a PaaS.

2.1.1 Defining an Actor

Ecosystems consist of many actors, the network between these actors, and the platform that the actors revolve around (Manikas & Hansen, 2013). Directly or indirectly, these actors engage in collaborative or competitive interactions with each other (Knodel & Manikas, 2015). Actors are defined as any participant existing in the ecosystem in a paper by Jansen et al. (2012). They pursue some kind of incentive, such as business interest, which drives them to participate in a SECO (Knodel & Manikas, 2015). Pursuing business-driven activities adds value to the SECO (Jansen et al., 2012). Typical instances of these actors are individuals, commercial organisations or governmental entities, non-profit organisations or social communities (the latter is primarily true for FOSS) (Knodel & Manikas, 2015; Manikas & Hansen, 2013). Furthermore, the variability and frequency in which actor roles exist within the ecosystem are crucial factors influencing the overall health and functioning of the ecosystem (Hartigh et al., 2006). Hence, platform owners aim to attract a diverse set of actors.

It is crucial to understand these actors' different roles and activities they pursue in the ecosystem. Although this topic has received some attention in the literature, a consensus is still challenging as most scholars approach classifying actors in their own distinctive ways. This is amplified by the fact that SECOs come in various ways, and the type of actors they will attract can differ substantially as well (Manikas & Hansen, 2013). Moreover, the variety in which distinct types of actors are attracted to an ecosystem depends on the openness of the platform (Knodel & Manikas, 2015). Hence, in this exploration a classification of actors is produced containing the potential actors that exist within a SECO. The next paragraph will discuss the development of actor-type classifications throughout the literature.

2.1.2 Actor Types Development in the Literature

A first overview provided regarding the roles within the ecosystem is done by Jansen et al. (2009). Two roles are described: the orchestrator and the niche player. The orchestrator is the owner of the platform and is in charge of the governance of the SECO as it 'orchestrates' the dynamics and strategic direction. Typically, this unit operates the platform, develops and implements policies and business practices, sets and monitors quality standards, and/or orchestrates interactions amongst SECO actors to manage the SECO (Manikas & Hansen, 2013). All other participants are named niche players. These actors build their businesses on top of the platform by conducting various activities (Jansen et al., 2009). They complement the orchestrator. A third actor, that of the dominator, was coined by Jansen et al. (2009). The

role of dominator refers to a firm that progressively eliminates or assimilates other niche players. However, this role was dismissed as it proved not sustainable for the longevity of a SECO to eliminate other niche players since the ecosystem's success heavily relies on its network and the number of participants.

As the literature on SECOs progressed, the classification of the types of actors found in SECOs has extended since the term 'niche player' is insufficient to comprehend the diversity and differences in activities they pursue. As mentioned earlier, various classifications have been made and have changed over time. Notable writers in the SECO literature, such as Jansen, create changing classifications in different papers (e.g. Handoyo et al., 2013; Jansen et al., 2012, 2008; Valença et al., 2018). Hence, it is difficult to compile a definitive classification of actors.

In spite of this, some actors keep resurfacing, however, they are classified and arranged differently in different papers. Other than these actors, there is hardly any mention of new kinds of actors. In Table 2, an overview is created of how different scholars have classified the roles encountered in a SECO. More classifications found in the literature could be added to this table. However, this would not account for new significant insights. Hence, this selection is included as these papers are the most influential in the SECO literature or present new ideas.

Table 2: Overview of how different scholars classify the different actors.

Jansen et al. (2009)	Jansen et al. (2012)	Manikas & Hansen (2013)	Handoyo et al. (2013)
Orchestrator	(Orchestator)	Orchestrator	Infrastructure providers
Niche players	Developers	Niche players	Software Vendors
	Value-Added-Resellers	Vendors (ISVs & VARs)	Value-Added Resellers
	Service partners	External partners	Service providers
	Customers	Customers	Customers or end-users

Jansen et al. (2012) extend the initial view of actors to four types: developers, Value-Added-Resellers (VARs), service partners and customers, as extracted from a multiple case study. While not mentioned, the role of the orchestrator remains existent since the SECO can not function without this role (Jansen et al., 2009). In this work, the role of niche player has been divided into developers, VARs and service partners, while customers were added separately. In this overview, the developers build and sell components and/or solutions, the VARs add functionality to the platform and resell it for profit, and the service partners deliver services to customers.

Similarly, in their systematic literature review, Manikas & Hansen (2013) explore the different roles and relationships between these roles that are found in the literature. They define a set of five different categories of actors that can exist within an ecosystem. These are the orchestrator, the niche players, external actors, vendors and customers. Within this classification, the niche players are the actors that complement and extend the work of the orchestrator, whereas the external actors are external to the SECO management and develop on top of the platform. The category of vendors merges Independent Software Vendors (ISVs) and VARs into one category as they both sell a product to the customer, whereas the value added from the external partners and niche players might extend to all participants. This is a

hard-to-apply categorisation as it remains vague, and categories seem to overlap. For example, third-party developers are categorised as 'external partners' while complementing the orchestrator's work as the niche players in this categorisation do. Furthermore, the review is based on literature that includes proprietary and open-source SECOs studies. There is a significant difference as a platform owner or leader governs proprietary SECOs, and in contrast, open-source projects are often managed by a community (Manikas & Hansen, 2013). As a result, the proprietary SECO has a more centralised character. In combination with the participation model of the ecosystem, this can attract different roles. Hence, classifying the roles separately for proprietary and open-source SECOs could result in different categorisations.

Additionally, similarly and building on the work of Jansen et al. (2012) and Manikas & Hansen (2013), the work of Handoyo et al. (2013) revises a classification of types of actors. This reclassification is highly similar to that of Jansen et al. (2012), as vendors and service providers, are similar to the developer and service partner categories, respectively—however, Handoyo et al. (2013) base their work on a large extent of papers, such as key publications on SECOs as well as related fields of study such as Software Supply Networks. This has led to a comprehensive and detailed list of actor types that can be found in a SECO. Subcategories further detail all categories provided in Table 2.

2.1.3 Actor Type Classification

The types of actors that are expected to be found in a PaaS SECO are listed below. This list is configured based on Handoyo et al. (2013), as their categorisation, provides the most detailed and complete overview. They consider key literature in their review, including the aforementioned literature and literature on cloud computing in ecosystems.

The overview is provided in Table 3. Slight changes have been made to fit the overview into the context of the PaaS SECO—for example, Handoyo et al. (2013) mentions the role of the infrastructure provider. However, this role is inherently integrated into the role of the platform owner in the case of PaaS, since the orchestrator is the PaaS provider. Hence, the infrastructure provider and orchestrator are merged into the same role. VARs, ISVs, and service providers have also been grouped under niche players for structural purposes. This also provides insight into how the work of Handoyo et al. (2013) fits into the initial classification by Jansen et al. (2009).

Table 3: Actor type classification.

Role	Activity	Source(s)
The orchestrator	Manage and govern the SECO	Jansen et al. (2009); Handoyo et al. (2013)
The niche player	Sell & service	Jansen et al. (2009)
Value-Added Reseller	Add & resell	Handoyo et al. (2013)
Independent Software Vendor	Build & sell	,,
Service providers	Sell service	,,
Customer	Develop, buy & deploy	,,
Individuals	,,	,,
Commercial organisations	,,	,,

Governmental entitities	,,	,,
Non-profit organisations	,,	,,
Social communities	,,	,,

The orchestrator
As was explained in section 2.1.2, the orchestrator is a business, division of a business, actor or group of actors, community, or autonomous organisation that is in charge of the SECO's smooth operation. In the case of a proprietary SECO revolving around a PaaS, this is the platform owner. Typically, this unit operates the platform, develops and implements policies and business practices, sets and monitors quality standards, and/or orchestrates interactions amongst SECO actors to manage the SECO. This entity attempts to attract business to its SECO through governance mechanisms. In the context of PaaS, the PaaS provider is the orchestrator.

Niche player(s)
ISVs are businesses that create products on top of the platform and sell them for profit. These can be components, modules, or complete integrations and tailored solutions. These are sold to the end-user. Especially for PaaS SECO, the relevant type of ISV is the SaaS vendor, also mentioned by Handoyo et al. (2013).

Furthermore, VARs profit from repackaging the platform by adding functionality and reselling it. The added functionalities come in different forms. VARs, for example, can buy a product, in this context, the PaaS, and add a service to sell the PaaS and service as a bundle. Additionally, VARs can add extensions and implemented functionalities to the product, increasing its value (Jansen et al., 2008). In the second scenario, the VAR could be seen as a developing firm, assuming it develops the extension itself.

Lastly, services can come in many forms. Handoyo et al. (2013) describe seven services generally provided within an ecosystem: Product distribution, software development, software design, application service providence, requirement engineering, system integration and content suppliance. As this study focuses on software developers' motivations within the SECO, the service providers of interest are the software development and software design consultancies. The rest of the list does not actively engage in software development processes.

It should be mentioned that niche players can only comprise instances of commercial organisations or freelancing individuals. However, freelancers or individuals in this category are not too common (van Angeren et al., 2016). As the incentive of these actors to participate in a proprietary SECO is to achieve commercial success, the actor instances of governmental entities or non-profit organisations, by definition, cannot fulfil this role. The same applies to social communities, which refer to development communities in open-source projects (Manikas & Hansen, 2013).

Customer
This category is the platform's end-user. This entity buys the products and services from the niche players (Handoyo et al., 2013). In the case of the PaaS SECO, the customer can also be a developing firm (Lucassen et al., 2013) or a customer of the SaaS build on top of the PaaS. As mentioned before, any actor can comprise a range of instances. It is assumed that in the case of the customer, this diverseness will be most visible in this category as the users of the platform and the customers of niche players can comprise any type of instance.

2.2 LITERATURE REVIEW ON DEVELOPER MOTIVATIONS IN THE CONTEXT OF SECOs

A literature review is conducted to establish insight into developers' motivation and incentives to join a SECO. First, the objective is described, and then the literature selection procedure follows. This section describes the search terms, refinements and exclusions used to come to the final selection of articles. Lastly, the findings are presented.

2.2.1 Literature Review Objective

The objective of this literature review is twofold. First, this review aims to create an overview of the state-of-the-art on the motivations of developers within the context of SECOs. Uncovering to what extent this has been researched, to what extent consensus has been reached on what motivations and in which type of SECOs this has been researched is crucial for developing the right methodology for this research. Secondly, the objective is to create an overview of the motivators. Whilst partly overlapping with the first objective, creating such a list is crucial for guidance in uncovering the motivations of developers in a PaaS SECO. Hence, the question '*What are the motivators of developers within a SECO?*' leads this review.

2.2.2 Search Criteria

The literature search was conducted on Web of Science and Scopus. The search terms used were 'motivation', 'ecosystem', 'developer' and any synonyms or themes related to these terms. The final string used for the literature search consisted of an 'AND'-joint string of the previously named terms.

Table 4: Assemblence of the search string.

Individual search terms	Related keywords
Motivation	motivators, incentives, factors, drivers, social
Ecosystem	-
Developer	-
Merged search string	
TITLE-ABS-KEY((motiv* OR incentive* OR factor* OR driver* OR social) AND "software ecosystem*" AND developer*)	

Table 4 shows the related search terms and how the final query was operationalised. Motivation was expanded by adding the terms 'incentives', 'factors', and 'drivers'. Incentives, drivers and factors may not directly translate to motivation, however, including these keywords helps integrate the scholarly articles on motivational factors that did not refer to their variables as motivations. While this requires scanning through more literature that may not be relevant, it will include articles that indirectly provide motivators. This helps to form a complete overview of the motivation of developers. Additionally, the term 'social' was added as this refers to dimension of interest for this study (Barbosa et al., 2013). Applying the final search query presented at the bottom of Table 4 amounted to 114 hits in Scopus and 76 on Web of Science.

The inclusion and exclusion criteria presented in Table 5 and Table 6 were applied to select the literature for the review. Using these criteria, an initial screening of the titles was executed. Doing so, 17 and 13 articles were selected in Scopus and Web of Science, respectively. After

removing duplicates, 15 articles were left. After a second screening of the articles, covering the abstracts of the selected articles, 8 articles covered the final selection.

Table 5: Inclusion criteria for literature selection.

Criterion	Inclusion Criteria
Unit of analysis	Developers (as individuals)
SECO dimension	Social dimension
Variables	Motivators, influential factors, incentives or drivers
Literature type	Empirical studies, peer-reviewed articles

Table 6: Exclusion criteria for literature selection.

Citerion	Excluded
SECO dimension	Business dimension, technical dimension
Language	Articles not written in English
Literature type	Non-peer-reviewed articles, reviews

2.2.3 Literature Search Result

The final selection of literature can be found below in Table 7. As shown, eight of the nine papers cover proprietary SECOs. More specifically, within the field of proprietary SECOs, research has only been conducted on the motivation of individual developers in Mobile SECOs (MSECOs). MSECOs refer to the ecosystems that revolve around smartphone platforms, such as the iPhone and Android. They are a distinct type of SECO with a whole different structure and network of participants than a PaaS SECO (Fontao et al., 2015).

Surprising is the scarce amount of literature found on FOSS SECOs. A reason for this is that FOSS ecosystems are coined using different terms. Some of the related terms used in the literature are Free/Libre Open-Source Software (FLOSS), Free Open-Source Software (FOSS), Open-Source Software (OSS), or Open-Source (OS), which influences how the term ecosystem is used. Additionally, in some cases, these ecosystems are referred to as projects (Manikas & Hansen, 2013). This interfered with the search process to include sufficient literature on FOSS SECOs. As the emphasis of this research is on proprietary SECOs, and some of the selected studies build on the motivations of these FOSS SECOs, it was decided not to address the scarcity of literature by reiterating the search.

Table 7: Final selection of literature.

Item #	Studies	SECO type
S1	Miranda et al. (2014)	MSECO
S2	Fontao et al. (2017)	,,
S3	Koch & Kerschbaum (2014)	,,
S4	Koch & Guceri-Ucar (2017)	,,
S5	Steglich et al. (2019)	,,
S6	de Souza et al. (2016)	,,
S7	Goldbach & Benlian (2015)	,,
S8	Constantinou & Mens (2017)	Open-source

2.2.4 Findings

In this section, the findings are presented and discussed. Not all motivators mentioned in the literature are elaborated on in the findings, as this section aims to find consensus and differences in the literature. In the following section, a list is created containing all the motivators from the literature.

Drawing on prior research on user innovation theories and FOSS developer motivations, Koch & Kerschbaum (2014) conducted a survey focusing on the motivators of individual developers to join a platform. In their research, they studied and compared Android and iOS developers and focused on their intrinsic and extrinsic motivations and, separately, the decision factors that play a role in joining a Mobile SECO. Interestingly, intrinsic motivators such as fun, intellectually stimulating work, and software development skills acquisition overall ranked higher than extrinsic motivators such as financial gains. These results align with prior literature on FOSS (e.g. Ghosh, 2007; Hertel et al., 2003; Lakhani & Wolf, 2007), as intrinsic motivation also seems to be the driving force in these SECOs.

Furthermore, significant differences were found between the motivations of developers to join one of the MSECOs, as financial gains ranked higher for iOS developers than for Android developers. Furthermore, it was found that iOS developers found the work more intellectually stimulating than Android developers, which could result from the distinct programming language developers have to learn to develop for iOS (Koch & Kerschbaum, 2014).

Lastly, this study presents a set of four decision factors that developers consider when choosing a certain platform (Android or iOS). These are the network size, the openness, the entry barriers and the support from the platform owners. Specifically for intrinsically motivated developers, the openness of the platform and the proposed entry barriers are important decision factors, while this could not be found for extrinsically motivated developers (Koch & Kerschbaum, 2014).

Another study by Miranda et al. (2014) approached the same issue of researching the decision factors through an exploratory approach, collecting data through semi-structured interviews. The main decision factors for developers to choose a certain platform, found in this study, are the size of the development community, the monetary costs of developing, testing and offering applications, the compatibility of the prerequisite skills and hardware to get started, the complexity of developing applications, and subjective advantages personal to the developer. These factors align with the decision factors found by Koch & Kerschbaum (2014). Interestingly, Miranda et al. (2014) study finds that Android has a higher complexity level, while Koch & Kerschbaum (2014) find that developers choose iOS to feel more intellectually stimulated. An explanation for this difference is that although iOS has a distinct dedicated programming language, there is little variation in the hardware models, the iPhones, while Android has a wide range of devices using its operating system (Miranda et al., 2014). Learning a new dedicated programming language might feel intellectually stimulating, while the latter adds a form of complexity that is not associated with intellectual stimulation.

Koch & Guceri-Ucar (2017) underline the importance and the success of intrinsically motivated developers in an ecosystem. In their study, they use the same motivational factors as Koch & Kerschbaum (2014) and link the type of motivation to the strategies these developers apply in app development and release. The most significant finding in this paper is that developers motivated by personal needs for apps show higher success in the marketplaces. Furthermore, no strong connections could be found between the developers'

motivation to offer an app for free, for free with in-app purchases or paid. Mainly, featuring apps in the marketplaces seemed to impact app success.

Another study focused on intrinsic motivation and its effect on app development is that of Goldbach & Benlian (2015). They focused on the informal control modes within ecosystems and how they influence intrinsic motivations and, in turn, app development outcomes and behaviour. They found that both self-control and clan control have a significant positive effect on intrinsic motivation. In turn, increased intrinsic motivation influences the app development efforts, the quality of the app and the intention to stay active within the ecosystem. Furthermore, self-control and clan control, mediated by intrinsic motivation, positively influence development efforts and a developer's intention to stay. However, these do not affect the quality of the app. Here, it becomes evident that intrinsic motivation plays an important role in the effort of developers and the quality of apps they produce. However, because extrinsic motivation was not considered in the explanatory model, the extent to which extrinsic motivation could be of the same importance is unknown.

A later study, conducted by de Souza et al. (2016), investigates social factors that enhance the adoption and continued participation of the developers in an MSECO, and uncovering the social aspects of mobile phone platforms. Social factors that were found to be important are that developers are often influenced by their social circle, previous experiences and perceptions of the local job market when choosing an MSECO. Furthermore, for continued participation in the MSECO, they found that social relationships (both online and face-to-face) and the presence of the orchestrator on online platforms and events can provide additional motivation. Lastly, the relationship with the end-users plays a profound role.

Additionally, building on the work of de Souza et al. (2016), Setglich et al. (2019) continued researching the social factors. They found that mostly intrinsic motivations are the driving force for developers to start contributing to an MSECO. The factors they found were most important are contributions to something new, learning and improvement of skills, having fun and knowledge sharing. These findings align with those of Koch & Kerschbaum (2014) since intrinsic motivations play a more important role than extrinsic ones. This is an interesting finding since 68% of the developers in the study by Koch & Kerschbaum (2014) are private developers compared to 100% of job-related developers in the study by Setglich et al. (2019). This would also explain why 'fun throughout the development process' ranks lower in the latter study since CET posits that providing an external stimulus (i.e. salary) will negatively affect intrinsic motivation. Also, the factor 'being intellectually stimulated' ranked lower in Setglich et al. (2019). A reason for this would be that Setglich et al. (2019) formulated that factor as 'Competition as an intellectual stimulus' to which most developers replied that they rather only challenge themselves personally, as argued for by Koch & Kerschbaum (2014).

While intrinsic motivation remains the most relevant motivation to continue developing in a SECO in the long term, and factors such as fun even increase in importance, developers rely more on external motivation to keep them engaged. Factors that increase in significance to continue development in a SECO are knowledge exchange among developers, the size and quality of the developer community, contribution to something new, fun during the development process, learning and improvement skills and developer self-management. Factors that emerge, yet in slightly less significance, are competition as an intellectual stimulus, developers' satisfaction with their MSECO and identification and commitment to the development community.

Constantinou & Mens (2017) researched the retention of developers within an OSS-SECO. They found that developers who do not communicate with other developers do not have a very strong social and technical activity intensity, communicate and commit on a lower frequency and do not commit or communicate for a long period of time have a higher probability of abandoning a SECO. The frequency of the core contributors, i.e. the small percentage of the developers that commit the most of the work, does not seem to affect the longevity of their contributions. These are not, per se, motivators, as these are predictors of abandonment. The researchers stress that these do not explain abandonment. Despite this, what stands out is that communication with a developer community is an influencing factor, as, for example, described by Setglich et al. (2019).

2.2.5 Developer Motivations

Additionally, to the literature discussed above, an exhaustive list of motivational factors is created that all directly or indirectly influence developers' motivation. It is important to note that these motivations have been established and researched in the context of FOSS ecosystems and MSECOs. This means that not all motivations apply to the context of a PaaS SECO. Hence, the list of motivations discussed in the literature review is assessed regarding applicability to the context of proprietary PaaS. Motivations such as 'Costs of development are lower than buying/adapting an existing application' or 'Feeling obligation to contribute because of use of apps' (Koch & Kerschbaum, 2014) have been removed, as the first is a somewhat strategic decision not applicable to any actor in a PaaS SECO and the latter is specific for SECOs where the developer is also the user of the application.

The remaining items were denoted as attributes of the developer in a PaaS SECO. In other words, to make this list of items a unitary attribute of the developer, the items were rewritten as an extension, an experience of, or an influence on the developer. Additionally, a differentiation has been made between motivators and facilitators of motivation. A technical resource, for example, is not a motivation or driver. However, its presence might facilitate or enhance the motivation of a developer. The latter group was not considered as it is not included in the scope of the research.

The final list of motivational factors is shown in Table 8.

Table 8: List of developer motivations from the MSECO literature.

Item #	Motivational factors	Source
_M1	Fun experienced throughout the development process.	Koch & Kerschbaum (2014), Goldbach & Benlian (2015), Koch & Guceri-Ucar (2017), Setglich et al. (2019)
_M2	The work being intellectually stimulating.	Koch & Kerschbaum (2014), Goldbach & Benlian (2015), Koch & Guceri-Ucar (2017), Miranda et al. (2014)
_M3	The acquisition & improvement of skills.	Koch & Kerschbaum (2014), Miranda et al. (2014), Goldbach & Benlian (2015), Setglich et al. (2019)
_M4	Contributing to something new.	Setglich et al. (2019)
_M5	Financial gains and rewards from the work.	Koch & Kerschbaum (2014), Goldbach & Benlian (2015),

		Koch & Guceri-Ucar (2017), Miranda et al. (2014)
_M6	To increase the position on the job market.	de Souza et al. (2016)
_M7	Creating and gaining a reputation in the developer community.	Koch & Kerschbaum (2014), Goldbach & Benlian (2015), Koch & Guceri-Ucar (2017), Setglich et al. (2019)
_M8	Identification and commitment to the development community.	Setglich et al. (2019), Koch & Kerschbaum (2014)
_M9	Attraction to the size and quality of the (online) developer community.	Setglich et al. (2019), de Souza et al. (2016)
_M10	The ability to exchange knowledge with other developers.	Setglich et al. (2019), de Souza et al. (2016), Constantinou & Mens (2017)
_M11	Participating in social community events.	de Souza et al. (2016)
_M12	Feeling the orchestrator's presence in the SECO.	de Souza et al. (2016)
_M13	Influences of one's social circle (both local and online).	de Souza et al. (2016)
_M14	Creating value for the end-user(s).	de Souza et al. (2016)
_M15	Satisfaction of the developer with their SECO.	Setglich et al. (2019)
_M16	Gaining the ability to self-manage work.	Goldbach & Benlian (2015), Setglich et al. (2019)

2.3 CONCLUDING REMARKS

This literature chapter consisted of two parts, of which both were aimed at answering a different sub-question of this research. This section aims to provide an overview of the results.

The first part of the chapter answered the first sub-question, '*What types of actors that actively develop exist in a PaaS SECO?*'. By exploring the literature, it was found that SECO actors are any participants in a SECO that add value directly or indirectly to the SECO by pursuing a certain activity, for example, creating products and extensions or providing services. Several instances of these actors can be found. Typical instances are individuals, commercial organisations, governmental entities, non-profit organisations or social communities.

The literature on SECOs is rather vague on the classification of actor types that may exist in a SECO. Several categorisations can be found throughout the literature, and not much research has been dedicated to providing a definitive categorisation. The main reason is that SECOs come in various forms, which makes it hard to create a one-size-fits-all type of solution as, consequently, different SECO types attract different actors. Despite this, the categorisation of Handoyo et al. (2013) is adopted as it is the most detailed and exhaustive classification and integrates literature on cloud computing. In this classification, the orchestrator, VARs, ISVs, service providers and customers are distinct actors in a SECO. Respectively, the activities of the orchestrator, VARs, ISVs and service providers are to manage and govern the SECO, add services and resell the platform, develop and sell products on top of the platform, and sell services. The customer develops apps on the platform for internal use and buys products and services from the niche players. In this list, the niche players are, by definition, commercial organisations or freelancers, whereas the customer can comprise all possible instances.

Secondly, the literature on developer motivations within the context of SECOs was systematically reviewed. This part aimed to answer the sub-question, '*According to the literature, what are the motivators of developers in a SECO?*'. A scarcity of literature on the topic was found, of which the most research was done in the field of MSECOs. These special types of SECOs have gained much attention in the literature. Results from this field cannot be directly transferred to the context of this research, yet provide relevant insights.

What becomes evident is that regardless of the commercial goals within these SECOs, intrinsic motivations come forward in most studies as leading developers to start developing and contributing to a SECO. The most notable motivational factors that were found were factors such as fun, intellectual stimulation and improving new skills. However, while still important, these factors might play less of a part for developers who develop job-related. The importance of intrinsic motivation becomes evident as it influences the app quality and efforts put into development. To continue developing in a SECO, intrinsic motivations remain a positive influence on the developer's willingness to continue developing in a SECO. However, extrinsic motivations are found to increase in significance in this stage.

3 THEORETICAL LENS ON MOTIVATION

This chapter focuses on motivation and how this construct can be approached for this research. This step is essential as motivation is a complex construct described differently in different theories. In this chapter, first, the landscape of motivation theory is unfolded. Then, a suitable theory is selected. Lastly, the selected theory is elaborated on in detail.

3.1 INTRODUCTION

Etymologically, the root of the word 'motivation' stems from the Latin word 'movere', which translates to "to move" in English. To translate this to the psychological concept of motivation, science aims to understand what drives individuals to undertake action and showcase directed behaviour. Such a straightforward explanation could give the impression that academics have a shared understanding of what motivation is and how this construct is defined. However, this is not the case. Even though the study of motivation is undoubtedly as old as the Latin word itself, much disagreement still exists regarding the definition (Kleinginna & Kleinginna, 1981). Especially throughout the past hundred years, many scholars have theorised about this topic, resulting in a rich and diverse body of literature with many different views on motivation. In spite of this, two common attributes shared by almost all theories are that motivation regards activation, i.e., the production of behaviour, and the directionality of that behaviour (Petri & Govern, 2004).

Several literature streams on motivation theory can be derived based on how motivation is viewed. Within these literature streams, many different theories exist. Selecting a theory that makes up a proper theoretical lens is crucial because this thesis places much emphasis on how motivation is defined. Hence, the following steps are taken to answer the question, '*How can the construct of motivation be conceptualised?*'. First, the literature landscape will be explored. Then, a single theory is selected and elaborated on in detail. The study is then conducted within the context of this theory.

3.2 MOTIVATION THEORY LANDSCAPE

As mentioned in the introduction, many theories and ideas on motivation date as far back as the old Greeks. In modern literature, after interest in motivation left the philosophical plain and entered the emerging field of psychology, several distinctions can be made.

3.2.1 Perspectives of Motivation Theories

Petri & Govern (2004), among many others, describe the different views on motivation. They propose four dimensions, each containing points representing opposing views, along which motivation can be categorised. They describe nomothetic versus idiographic, which, respectively, views motivation as universal laws applicable to any individual versus a phenomenon that is unique for any person. Innate versus acquired refers to motives that come forth from instincs versus assuming that behaviour is acquired. Thirdly, internal versus external, in which internal factors are often operationalised as needs and external factors are operationalised as goals and one's surroundings. Lastly, the mechanistic versus cognitive dimension refers to whether an individual's behaviour is a hard-wired automatic response or, in the latter case, a person has control over their behaviour.

Given these dimensions, two distinct literature streams are identified. A literature stream that takes the instinctive perspective, which is nomothetic, mechanistic, innate, and in response to

internal and external cues. The second perspective is cognitive and takes a monothetic, cognitive, acquired perspective in response to internal and external cues (Petri & Govern, 2004).

Furthermore, Arango (2018) distinguishes literature streams that are highly comparable to that of Petri & Govern (2004), though they identify a third stream emerging in the field of neuroscience. In these literature streams, the most influential theories are identified. These are literature on theories about instincts, drives, needs and reinforcement, theories about beliefs, values and goals and theories that take a neuroscientific perspective. The first two are comparable to instinctive theories and cognitive approaches. While the author is clear that she does not present an exhaustive review of the literature, she covers in detail what sets these literature streams apart. Furthermore, despite her book's focus on children's pedagogical processes, her list is still relevant as it presents the most influential theories used in work motivation literature. Hence, this is adopted in this research as well.

The first category, theories about instincts, drives, needs and reinforcement, pertain to the more historical theories that view motivation as intrinsic biological forces innate to the species and guide behaviour to live and maintain or restore homeostasis (i.e., a state of balance). These forces include instincts, drives, and needs (Arango, 2018). One famous example of the philosophy within this stream of motivation theory is Maslow's Hierarchy of Needs (Maslow, 1943). For this research, these theories do not prove useful since they focus more on the primal behaviours and biological processes, while this research emphasises why individuals engage in higher-level psychological behaviour.

A second, more modern, stream of theories is about beliefs, values and goals. These theories focus on the processes influencing motivation, assuming that the individual controls that process (Arango, 2018). These scholars view motivation as monothetic, cognitive, acquired and in response to internal and external cues. According to Eccles & Wigfield (2002), several types of theories can be derived from this stream of literature. Among others, these theories focus on 'Can I do this task?' and 'Why I do this task?'. The first focuses on how individuals perceive tasks and their expectancy of fulfilling those tasks, while the latter focuses on reasons for engagement.

Lastly, the third and most recent category approaches motivation from a neuroscientific stance. These scholars aim to understand the cognitive and neural processes happening within the individual that lead to goal-oriented behaviour. Within this stream, notable theories are the somatic marker theory (Damasio, 1996), which emphasises the combination of cognitive and emotional processes that leads an individual to act out certain behaviour, and the triadic model (Ernst et al., 2006), which emphasises neural processes. While this stream of literature is profoundly interesting, the depth of neuroscience is less applicable within the scope of this research.

This brief exploration clarifies that the cognitive approaches to motivation are useful for this research since the individual is assumed to be in control of their behaviour. Hence, these theories are elaborated on in the next paragraph, and based on these theories, a view on the construct of motivation is selected.

3.2.2 Cognitive Approaches to Motivation
Some theories focus on expectancy. Examples of these theories are self-efficacy theory (Bandura et al., 1999), expectancy-value theory (Atkinson, 1964; Wigfield & Eccles, 1992) and

attribution theory (Weiner, 1985). These theories focus on achievement motivation, i.e. the expectancies of success, and how these factors predict the motivation of individuals to act, for example, according to Bandura et al. (1999), in self-efficacy theory, the motivation of the individual is dependent on their expectations of whether a certain outcome or goal is attainable if the correct behaviour is shown and their expectations on if they can showcase that behaviour. On the other hand, the expectancy-value theory posits that motivation is determined by the expectancies for success and the subjective task value. The expectancies refer to a person's confidence level in their ability to carry out a task successfully. In contrast, the latter refers to how the task is perceived in terms of importance, usefulness and enjoyability.

Another prominent substream is that of Self-Determination Theory (SDT) (Deci & Ryan, 2000). This theory differs from expectancy theories as it shifts away from expectancies and moves towards the reasons for engagement. The theory posits that individuals have a natural tendency to move towards self-determined behaviour (i.e. intrinsically motivated) and researches under which conditions this is attained. Furthermore, this theory is unique in identifying various types of motivation as it describes intrinsic motivation, i.e. self-determined behaviour, and extrinsic motivation. The main focus of this theory is to understand how intrinsic motivation works, how it helps engage people, and under what conditions extrinsic motivation works well. By fulfilling basic psychological needs for autonomy, competence and relatedness, individuals are said to move towards intrinsic motivation. Although this theory appears to place much emphasis on needs, it should be noted that it specifically emphasises the individual's internal regulating mechanisms. Also, values and beliefs play a significant role in regulating different types of extrinsic motivation. As a result, it belongs to this category.

3.2.3 Theory Selection

Self-Determination Theory is selected for further investigation as it has certain elements that suit the aim of this research. What makes this theory stand apart is its emphasis on the basic underlying needs of individuals and the regulation mechanisms of how the context influences a person to achieve self-determination (i.e. intrinsically motivated behaviour) (Deci & Ryan, 2000). Secondly, unlike most theories, it does not treat motivation as a unitary concept. It proposes different types of motivation and the sources of these motivations. This research is an excellent fit for this study since it acknowledges the various sources of motivation, which is essential for understanding why developers join SECOs.

The following section will delve into SDT and elaborate on its views of motivation to set the stage for the rest of this research and provide a theoretical framework.

3.3 SELF-DETERMINATION THEORY

SDT is a theory on motivation that has grown very popular. SDT is founded on the idea that psychological growth, internalisation, and well-being are people's natural tendencies and progressions. Furthermore, people influence and are influenced by their environment in ways that either help or hinder the realisation of these tendencies. This theory focuses on understanding under which conditions this behaviour towards growth is met, as the theory prescribes that psychological growth manifests itself in self-determined behaviour (Ryan & Deci, 2018).

While its roots are set in psychology, SDT has grown to many disciplines, such as management (van den Broeck et al., 2016), health sector (Ng et al., 2012) and education (Jang et al., 2010),

among others, as the leading framework for understanding the motivations of people within their contexts. Formally, SDT is a collection of different mini-theories, of which the Cognitive Evaluation Theory (CET) and the Organismic Integration Theory (OIT) are most significant. SDT will be outlined by addressing these theories in more detail.

Cognitive Evaluation Theory
According to CET, there are three basic psychological needs that must be satisfied in order to attain psychological growth: the need for autonomy, competence, and relatedness. SDT defines these needs as the "nutrients that are essential for growth, integrity, and well-being" (Ryan & Deci, 2018, p. 10). Besides physiological needs to feel safe and secure, SDT asserts that the existence of fundamental psychological needs must be satisfied in order to maintain psychological interest, development, and general well-being. The theory posits that these needs are as essential to psychological growth as physiological needs are essential for safety and a sense of security.

The fulfilment or deprivation of these psychological needs, like physiological needs, can significantly impact a person's development, integrity, and general well-being. Individuals are more likely to experience increased motivation and a sense of fulfilment in their lives by attending to these psychological needs (Ryan & Deci, 2018).

The need for autonomy refers to the need to self-regulate one's experiences and actions. In this sense, the individual's behaviour is self-endorsed or congruent with one's authentic interests and values. According to SDT, only a few intentional actions are truly self-regulated, while many other actions are regulated by external forces. Autonomy should not be confused with independence. It means to act with a sense of choice and volition. This means that while someone (with authority) might tell an individual to do something, the need for autonomy is still satisfied if the individual agrees volitionally (Ryan & Deci, 2018).

The need for competence refers to an individual's need to feel in control of the mastery and development of new skills. It pertains to the urge to feel successful, competent, and efficient in one's endeavours. Having a sense of mastery and achievement entails pursuing tasks that are both interesting and doable (Ryan & Deci, 2018).

The third basic need, according to SDT, is relatedness, which describes the need for a sense of community, belonging, and deep connections with other people. Being socially engaged with others generates a sense of emotional fulfilment since humans are naturally social beings. Positive and encouraging connections make people feel heard, valued and cared about (Ryan & Deci, 2018).

In short, by facilitating the needs for competence, autonomy and relatedness, motivation can be achieved and increased. Competence can increase intrinsic and extrinsic motivation, relatedness correlates positively with intrinsic motivation, and autonomy also has a significant role in intrinsic motivation.

Organismic Integration Theory
SDT states that several types of motivation are paired with different types of sources of motivation. The three main types are amotivation, extrinsic motivation and intrinsic motivation (Ryan & Deci, 2018). Amotivation is defined as the total lack of motivation while intrinsic motivation refers to motivation that comes forth from engaging in an activity purely for the enjoyment of it. Furthermore, extrinsic motivation refers to motivation triggered by

external sources or behaviour ignited to pursue a reward or specified end-state (Ryan & Deci, 2018).

Furthermore, OIT posits that extrinsic motivation can be divided into four separate types, based on the individuals' perceived autonomy. This means that extrinsic motivation differs depending on the internalisation and integration of the regulation of the activity (Ryan & Deci, 2000a). The four proposed types of motivation are external regulation, and introjected regulation, both controlled types of motivation due to low perceived autonomy, and identified regulation and integrated regulation, together with intrinsic motivation, both autonomous forms of motivation (Ryan & Deci, 2018). These types of motivation should be considered as a continuum, as shown in Figure 1. The more controlled the environment is, the less self-determined the behaviour will be, whereas higher perceived autonomy leads to autonomous forms of motivation. Below, the types of extrinsic motivation are further explained.
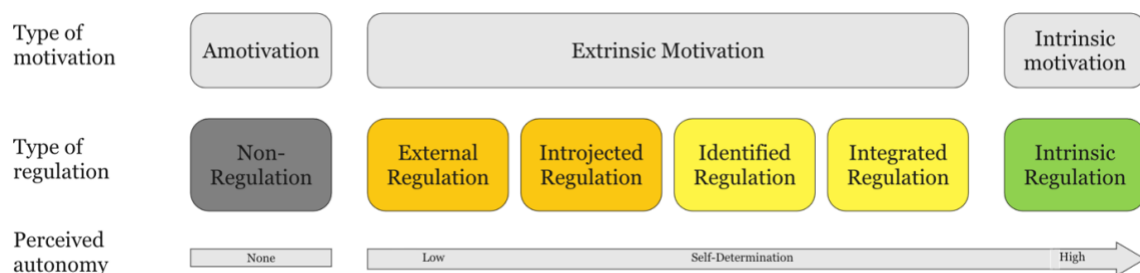
Figure 1: The control-autonomy continuum (based on Ryan & Deci (2018)).

External Regulation – This type of extrinsic motivation is based on external regulation. As the term hints, it is the most externalised type of motivation. It entails that individuals partake in an activity solely to reap benefits or avoid punishment. Within this type of motivation, individuals feel obligated to abide by external pressures (Ryan & Deci, 2018).

Introjected Regulation – Internalised pressure or guilt are characteristics of this motivational style. People participate in an activity to boost their self-esteem, prevent failure or shame, or win the favour of others. Although most of the regulation is still influenced by outside forces, the person has somewhat internalised these forces into their ego (Ryan & Deci, 2018).

Identified Regulation – People who are motivated in this way recognise the worth and significance of an activity and voluntarily participate in pursuit of their own aims and ideals. Even when external influences are present, they have internalised the meaning of the behaviour and recognise it as personally significant because it aligns with their values and beliefs. As an example, consider someone who studies for a test despite having some reluctance because they feel that education is essential to achieving their long-term goals (Ryan & Deci, 2018).

Integrated Regulation – When such identifications with one's beliefs are integrated with one's other values and beliefs, it is called integrated regulation, associated with a higher quality of persistence and performance compared to more controlled types of motivations (Ryan & Deci, 2018).

According to OIT, the various extrinsic motivations have distinct impacts on intrinsic motivation. The extrinsic motivations that arise when the environment is perceived as more

controlling (external and introjected regulation) negatively impacts intrinsic motivation. In other words, providing a reward negatively affects self-determined behaviour. On the other hand, when extrinsic motivation is perceived as autonomous, it will positively affect intrinsic motivation (Ryan & Deci, 2000b).

3.4 CONCLUDING REMARKS

In this chapter, various theories of motivation have been explored to answer the sub-question '*How can the construct of motivation be conceptualised?*'. SDT is selected because it does not treat motivation as a unitary concept but breaks it into several categories. Also, focusing on the degree to which an individual's choices are self-determined and what needs must be met to facilitate this behaviour can provide crucial insights into their reason for engagement.

Three basic psychological needs, autonomy, competence and relatedness, are facilitators of motivation. The frustration of these needs decreases motivation, while facilitation increases motivation. Since these needs affect motivation, understanding to what degree individuals feel these needs are met can provide insights into their motivation.

SDT posits three types of motivation, amotivation, extrinsic motivation and intrinsic motivation, of which external motivation can be divided into controlled and autonomous forms of extrinsic motivation. The controlled types of motivation are external and introjected regulation and harm an individual's intrinsic motivation. Whereas the autonomous types of regulation, identified and integrated regulation, positively affect an individual's intrinsic motivation. The autonomy an individual perceives within their environment has a crucial effect on their motivation.

In the context of this study, the motivation of developers can be mapped in terms of the motivation types posited by SDT. Additionally, various factors that facilitate or explain the motivation can be expressed in terms of the basic psychological needs for competence, autonomy, and relatedness.

4 SYNTHESIS

This chapter summarises and merges the findings of the previous literature chapters that discuss sub-questions one to three as a main conclusion to the desk research phase. Sub-question four regards the case study, and to prepare for the design of this research phase, case study questions are created based on desk research findings. These questions will support in designing the methodology.

4.1 MERGING THE WORK

This study assumes that the actor type classification provided in Table 3 is sufficient to specify the actors in a PaaS SECO. Actors are entities within the SECO that pursue activities or some sort of value increase in SECOs. They differ based on what value they add to the SECO and their activities within that SECO. Additionally, actors among each other interact differently as well.

From an SDT perspective, this is interesting because while all developers engage in application development, the purpose of development is different depending on the SECO actor. For example, service providers create apps for external actors, mostly customers. In comparison, the customer developers will create in-house for their own organisation. This could result in differences in motivation as developers have different reasons for engaging in that activity. While the health of the SECO is highly dependent on the variation and the number of different actors an orchestrator can attract, the current state of the literature does not explain nor explore how the motivations of these different actors might vary.

As shown by the covered literature, the topic of motivators of developers in the context of SECOs has received little attention within the context of SECOs. Only for the MSECO a large amount of literature is available. Based on the existing literature, a set of potential developer motivators was produced. From the literature, it also becomes evident that motivation evolves over time, as some studies point out that some of these motivators are more important to join than to stay in the SECO. This means, according to SDT, that the fulfilment of needs of developers shift.

Since the architecture and the type of SECOs differ highly from PaaS SECOs, it is unclear to what extent the motivators apply to PaaS SECO developers. Nonetheless, to pursue a definitive list of motivators for the PaaS SECO developers, this list supports finding the motivators in this research. Also, it helps uncover potential differences with the other SECOs.

Most of these studies on the motivators of developers have been conducted without a specific lens through which the construct of motivation is approached. While several articles do mention SDT or concepts related to SDT (e.g. Koch & Guceri-Ucar, 2017; Koch & Kerschbaum, 2014; Steglich et al., 2019), it's unclear how much a particular theory of motivation is being used throughout the studies. Goldbach & Benlian (2015) is an exception to this. Blindly applying SDT theory to the motivators found in literature is difficult since the motivation is highly personal. Uncovering the nature of motivators and how they drive the developer is impossible without in-depth information. Hence, applying SDT to the motivators found is to be explored in this study. This would also yield interesting insights into the benefits of applying SDT.

Applying SDT is crucial for this study and will help the research in three ways. First is that the theory will allow the motivators of developers to be classified along the control-autonomy continuum of SDT so that it becomes clear how each motivator is experienced. Second, the theoretical concepts help to establish a more thorough analysis of motivation. Since it is unclear how the found motivators are applicable in the context of PaaS, the SDT lens can assist in arranging the motivators. Lastly, the basic psychological needs posited by SDT can help nuance influential factors and help clarify underlying influences. In light of the research objective, these points are critical as they help to contextualise developer motivation in the context of PaaS and ground the findings in theory, strengthening the validation of the results.

The following section proposes suitable case study questions based on the findings and synthesis.

4.2 CASE STUDY QUESTIONS

As Yin (2009) proposed, case study questions can help guide the direction of the research. Case study questions are an essential component of all case study research. Hence, this section presents questions to guide the researcher throughout the case study. The questions also guide the design of the case study presented in Chapter 5.

Case study question 1: What motivates developers to start participating in a PaaS SECO?

This first question is a direct replicate of the fourth sub-question and also the most important question. This question aims at getting a general understanding of developers' motivators within a PaaS SECO. This is a relevant first step since this has not been studied before. Additionally, applying SDT to this sub-question will enhance the strength of the findings.

Case study question 2: How do the motivators to continue in the SECO differ from those to start?

As seen in the literature review, the motivations of developers in the continuation of participation in the SECO differ from the motivations to join. The motivations for continuing should also be investigated to achieve the research objective. Hence, the second sub-question explores this in the context of a PaaS SECO.

Case study question 3: What are the differences in motivation between the developers of the SECO actors in a PaaS SECO?

SECO actors differ based on the activities they pursue within the SECO and how they add value to the SECO. As developers also engage in different activities, the question is whether this is reflected in different forms of motivation.

Case study question 4: How do the motivations of developers differ along the dimensions of competence, autonomy and relatedness?

The last question aims to give a critical look at the motivations of developers and find ways motivation can differ between developer groups. It is important to remain critical in case studies and seek rival explanations for occurring phenomena (Yin, 2009). Hence, this question aims to observe any potential explanations for differences in motivations between developers.

5 METHODOLOGY

This chapter describes the research methodology that is conducted for this research. First, the approach is selected based on the previous findings, then the appropriate research methods are selected, how these methods are deployed and what sample design is used. Lastly, the data analysis approach is elaborated on.

5.1 RESEARCH APPROACH

In the previous chapter, the theoretical background is provided. Thus far, it has become evident that literature on the motivations of developers is scarce, covering only two types of SECOs: The MSECO and FOSS SECO. Both are distinct SECOs with an incomparable architecture and network to a proprietary PaaS SECO.

An exploratory approach is chosen because it is unclear how these developer motivations translate to the PaaS SECO. The objective further justifies the exploratory methodology, as this study seeks to determine the motivation of the developers of the diverse actors – a phenomenon that has not been covered to any extent in the literature. A drawback of this approach is that the results of exploratory research are typically not generalisable to the population (Sekaran & Bougie, 2016). Yet, as in this stage, it is hard to define which motivations influence which actors and how, It is difficult to draft an explanatory model and approach this by employing a (field) experiment, for example.

Hyrynsalmi et al. (2015) asserts that since the results heavily depend on the type of SECO, generalising the results from one SECO to another may not be precise. This also becomes evident within a single SECO type, such as the MSECO, in which apparent differences exist between the motivations of iOS and Android developers (Koch & Kerschbaum, 2014). Hence, the case study is selected to gain an in-depth understanding of this phenomenon in a single, demarcated, real-world setting, as this is facilitated by the case study approach (Yin, 2009). A case study focussing on the developers within a single ecosystem will benefit the reliability of the results compared to research that includes developers from various PaaS SECOs.

Therefore, the PaaS SECO is defined as the case in which the developer, the unit of analysis, will be studied. Due to time and effort restrictions, it was decided to conduct a single case research. Additionally, devoting all efforts to a single PaaS will contribute to the depth in which the phenomenon can be researched. For this research, a representative case is selected. The case selection is elaborated on in the next section.

5.2 CASE SELECTION – MENDIX

A case study will be conducted on the SECO of Mendix. Mendix is a PaaS specifically for developing enterprise applications (Mendix, n.d.). Mendix has attracted many businesses to settle into their ecosystem by offering a low-code modelling language. Low-code, compared to high-code programming languages such as C#, Java, and Python, is a simplified modelling language that requires less expertise and skills in the development of apps. Due to the simple nature of modelling, the time to develop and launch an app significantly decreases compared to high-code programming languages. Additionally, because less expertise in IT is needed, a broader range of participation in app development is possible. Mendix targets and promotes the deployment of fusion teams with their Low-Code Development Platform (LCDP). Fusion teams are a new software development team that blends IT and business domain experts into

multidisciplinary teams. These teams are emerging as the boundaries between IT and business are blurring due to the digitisation of business (Cain & Severson, 2023).

The low-code PaaS SECO represents a particular type of PaaS SECO and is not representative for all PaaS. Since the landscape of PaaS is very broad, picking a specific type of PaaS case, as suggested by the fourth sub-question, is most desirable. A case within the segment of low-code PaaS is selected as this market is significantly gaining momentum. Gartner predicts that in 2027, 70% of app development will use or embed low-code (or no-code) (Vincent, 2022). Given the growing significance of low-code, investigating the developer motivations within a low-code PaaS SECO becomes all the more relevant.

Figure 2: Magic Quadrant of LCDP vendor market positions (Vincent et al., 2022).

Within the segment of low-code PaaS providers, the case of Mendix is selected for several reasons. The first is that Mendix is a successful and pioneering firm in PaaS, more specifically in the segment of low-code (Vincent et al., 2022). Mendix's position in the market is shown in Figure 2. Being a successful and leading orchestrator, the Mendix platform provides fruitful business opportunities for many organisations. Since its participants consist of many types of actors, this case assists in reaching the research objective. Second, due to its success and growth, Mendix has established a thriving ecosystem with a global developer community of 300,000 developers and is still expanding (Mendix, n.d.). This developer community is rich and diverse and consists of developers and actors located all over the world. Gradually, per geographical region, communities are growing that, without the help of Mendix, are organising meet-ups and events. The developer community's growth and size pose a second argument for the case of Mendix, as the maturity of the developer community provides a better chance of understanding the importance of all motivators. Of all communities, the largest and most densely networked community of developers is located in the Netherlands. Furthermore, Mendix has established partnerships with many organisations that have settled in their ecosystem. These organisations increase the value of the Mendix platform significantly.

Since Mendix is the leader in the low-code segment of PaaS, all actors are present, and the developer community is rather mature, it makes a representative case for low-code PaaS. At

the same time, it is a unique case for PaaS, given the upcoming technology it deploys and the new type of developers it attracts.

5.2.1 SECO Actors

One of the reasons to select Mendix is the diverse set of SECO actors that have settled within the Mendix ecosystem. After an inquiry and evaluation, most actors have been found to exist within the Mendix SECO. This is shown in Table 9. All actors are present, whereas not all instances that actors can comprise are. This means that not all actors can be represented in their full diversity. However, it is not considered a drawback since all actors are covered.

Table 9: SECO actors present in the Mendix ecosystem.

Actor	Present in Mendix
The orchestrator	Yes
The niche player	
Value-Added Reseller	Yes
Independent Software Vendor	Yes
Service providers	Yes
Customers	
Individuals	No
Commercial organisations	Yes
Governmental entitities	Yes
Non-profit organisations	No
Social communities	No

Orchestrator

Mendix, the organisation, is the PaaS provider of the SECO and, thus, the orchestrator. Besides the orchestrator's role in governing the ecosystem, Mendix also has a developing role here. First and foremost, they develop and innovate the LCDP. Second, the LCDP is utilised to create apps for internal use. Lastly, Mendix provides expert services to niche players, end-users, and partners, setting up the LCDP for strategic use. These expert services are the equivalent of internal service providers. Despite their development efforts, these developers are not counted in the developer community.

Niche players

Of the niche players defined in the literature, all actors are present within the Mendix ecosystem. Mendix research regarding the spread between niche players and customers shows that niche players employ more developers than customers. The VARs in the ecosystem do not develop directly but help customers deploy the LCDP as a service and, in some cases, provide training programs. This category will not be considered as these firms do not employ developers. The ISVs play a prominent role in the ecosystem, as they both deliver SaaS solutions to end-users and module solutions in Mendix's marketplace. There is a large extent of service providers that provide services to customers, starting with Mendix.

From the literature, it became clear that niche players could arise in the form of freelancers or commercial organisations. In the Mendix ecosystem, 99% of the developers are employees, while less than 1% are freelancers. Therefore, the chances of finding freelancers to include in this study are low and will not be considered.

Customers

The developers of end-users make up roughly 40% of the ecosystem. Three exceptions to the existence of actors can be found in the ecosystem. These are the instances of individuals, non-profit organisations and social communities. This is because Mendix is a strict proprietary SECO that does not allow free use, and the individuals and social communities are primarily found in FOSS SECOs. Furthermore, due to the high costs of Mendix licenses, it is reasoned that non-profit organisations need more financial funds to switch to Mendix.

5.2.2 Developer Characteristics and Developer Community

The previous section discussed what software-developing actors exist in the SECO. This section dives into the different software developer characteristics. As mentioned, Mendix promotes the deployment of fusion teams. While being promoted, the current state of the developer community consists mainly of low-code developers with a background in high-code programming languages. Roughly 60% of the developer community comprises developers with a high-code background. Developers that did not have a prior background in high-code is around 30%.

5.3 DATA COLLECTION

This section describes the selected data collection methods, sample design and execution of data collection.

5.3.1 Data Collection Method Selection

A data collection method was selected based on the complexity of the motivation construct and the selected research approach. Additionally, data collection methods have to align with the available resources, such as time and effort and the availability of the data sources. Combining these arguments led to the choice of interviews. The interview is chosen over other collection methods, such as questionnaires, since interviews allow the researcher to ask follow-up questions, which may result in new insights, thus being highly appropriate for exploratory research. Moreover, interviews allow for a lower sample than questionnaires. Due to the uncertainty of access to the data sources, the interview method poses a project managerial advantage.

Interviews are described by Sekaran & Bougie (2016) as guided, purposeful conversations between two or more people. They are used in qualitative research and are highly suitable for exploratory studies. Semi-structured interviews will be conducted to gain an in-depth understanding of motivations. This allows the researcher to ask follow-up questions, improvise and explore the answers to pre-determined questions, which may result in new insights (Runeson & Höst, 2009). According to Yin (2009), this semi-flexible interview format helps reveal how case study participants construct reality and provide depth in their understanding of the situation instead of merely answering the researcher's questions. In short, interviews will help create rich data on the topic suitable for the exploratory nature of this research.

This format also poses some limitations. One limitation of this method is that it may be hard to compare the results of the interviews due to the semi-structured nature of the protocol. This could reduce the validity of the research (George, 2022). Also, it is advised that interviews are conducted until saturation is achieved, meaning that no new information surfaces by conducting additional interviews (Corbin & Strauss, 2008). This limitation poses a threat as

there might be insufficient resources to conduct interviews until saturation. However, since this study is exploratory and not correlational, these limitations are accepted beforehand.

5.3.2 Population

Although the Mendix ecosystem is home to a globally spreading developer community, this research geographically delineates its population to the Netherlands. The main reason is to mitigate any effect cultural differences can have on motivation since, according to SDT, motivation can be value-driven (Deci & Ryan, 2004). Hence, sampling only from one culturally delineated area increases the validity of the research. Additionally, the community in the Netherlands is the most mature, making it the most suitable population to find an answer to the research question. However, a drawback is that generalisations to the global ecosystem might prove difficult or slightly inaccurate.

5.3.3 Sample Design

First, the diversification criteria are elaborated on. Then, the selection criteria for potential participants is discussed. Lastly, the sample method is discussed.

Participant Selection

Selection criteria must be implemented to assess potential candidates deemed sufficient for interviewing. As this research regards the entrance and continuance of developers, the developers need experience and knowledge in Mendix. Hence, selection criteria based on experience and expertise are required. Concerning the expertise criterion, Mendix provides four certifications degrees of expertise. These are Rapid, Intermediate, Advanced and Expert. The Rapid, Intermediate and Advanced certificates are gained through online exams, whereas to gain the Expert certificate, the developer is interviewed by a Mendix expert within the Mendix organisation. While these levels do indicate the developer's expertise level to an extent, it may occur that the developer's certification is not representative of their actual expertise. Apart from the expertise criterion, participants require at least two years of experience in the SECO. This is needed to gain valid insights on the continuance of developers. Lastly, active developers are required, since inactive developers cannot be counted as part of the ecosystem. Table 10 shows an overview of the criteria.

Table 10: Selection criteria for the participants.

Selection criteria
The developer is at least Intermediate certified
The developer has at least 2 years of experience in Mendix
The developer is active on the platform at least once a week

Sampling

When sampling participants for a qualitative study involving interviews, gathering a diverse set of participants in terms of personalities and roles is recommended (Runeson & Höst, 2009). Regardless of the priorly set criteria, there are many variables such as age, prior career, background, and organisation size, that may play a role. Due to the exploratory character of the research approach, the ideal sample design would be that of purposive sampling, as this allows the researcher to pick participants based on their judgment to achieve diversity. Within purposive sampling, quotas are set for each of the actor categories.

Mendix company policy only allows gathering potential participants through their professional network of Mendix or dedicated channels, as communication with external

developers has to align with the General Data Protection Regulation (GDPR). Also, the reputation of Mendix is to be considered when approaching developers. Hence, in accordance with Mendix, the participants are approached through the following channels:

- The partner account managers for ISVs and service providers;
- customer success managers for customers;
- and the online Mendix forum in Slack.

Diversification

To maintain a reliable and consistent design, achieving an equal number of participants for each actor or sample would have been ideal. Based on the available resources, a list with the number of participants is aimed for was made. A total of 12 participants was deemed realistic. Table 11 presents the intended number of participants.

Table 11: The number of intended and attained participants per SECO actor.

Diversification	Number of participants
ISV	3
Service partners	3
Customer	
Public organisation	3
Commercial organisation	3

The participant sourcing execution is discussed in the next chapter in section 6.1

5.3.4 Interview Protocol

The list of pre-determined interview questions was revised multiple times as pilot interviews were conducted with internal Mendix developers to get preliminary insight into how developers would respond. Due to the limited number of questions and heavy reliance on SDT concepts in the initial draft, comparing the outcomes of the pilot interviews was challenging as highly personal stories dominated the results. To avoid comparing the results on a very high abstract level, it was decided to shift the focus towards more directed but open questions regarding the developer motivations that resulted from the literature review. The final version contained a set of twelve interview questions. The interview would be conducted using a funnel-like structure, beginning with four warm-up questions about the interviewee's professional background and how they were introduced to Mendix, moving on to eight in-depth inquiries about their motivations. The list of developer motivations from the literature review was used as a checklist to ensure each motivator was addressed after the interview. The interview protocol can be found in Appendix A.

The warm-up questions are meant to activate the interviewees and direct their attention towards thoughts about their motivation and how they started in the SECO. The in-depth questions started with a general open question about what drove them to start their participation the SECO and what drives them now. This first question is initiated to understand what developers would mention themselves without any of the motivators from the checklist being mentioned. From whatever would be mentioned, the researcher will follow up.

The aim is to start a free-flowing conversation to discuss all motivators from the checklist, allow new motivations to arise, and understand the impact and importance of all motivators mentioned. Throughout the interview, the checklist will be kept up to date. Additionally, the interview protocol allows the researcher to get into one of the motivations from the checklist at any time, if possible, even during the warm-up questions, to ignite the conversation and deepen the answers provided by the interviewees.

Questions six to twelve, in some way, all relate to the motivator checklist. These questions are used in case the conversation comes to a halt. These questions aim to remain somewhat open while asking about motivators from the checklist. Motivators were grouped into themes, if possible and then formulated as questions that can lead to a free-flowing conversation. This is done to maintain an open, semi-structured format and avoid a static conversation where each motivator is asked about sequentially. Question six is an open question about the personal satisfaction they gain from Mendix, which refers to intrinsic motivations from SDT (Ryan & Deci, 2018), without specifying any motivator specifically to avoid limitations in the interviewee's response. Any follow-up questions are asked regarding _M1 to _M4 if needed since these potentially cover intrinsic motivations. Question seven regarded career prospects, and follow-up questions were asked about _M5 and _M6 if needed. Furthermore, Question 8 referred to a question regarding all motivators related to the developer community (_M7 to _M12). The remainder of the motivators could not be grouped and were asked separately in questions nine to twelve if needed.

The last question was asked to check with the participant that everything concerning their motivation was addressed. The in-depth questions on motivation were built up on the list of motivators. Motivations on a specific topic, such as career or community-related, were grouped and formed into open questions to stay true to the open character of the interview.

5.3.5 Interview Procedure

Before any interview, the consent form was mandatory to be signed by the interviewee. Then, at the start of the interview, an introduction was provided on the topic and the researcher conducting the interview. The interviewee would be notified of the start of the recording to provide full transparency. All interviews were conducted in English. The interview protocol was followed as described. Throughout the interview, follow-up questions such as 'How important was X?' and 'Why was that important?', followed by 'How is that now?' or 'When did that start to become important?' were repeatedly asked to understand an interviewee's answer better. Then, based on the conversation, other questions would be probed, and items from the checklist would be coined if the interviewee did not provide any information regarding these motivators. The interview ended if all boxes were checked and the interviewee had shared everything regarding their motivation. Generally, the course of the interview was such that half to three-quarters of the interview was free-flowing, whereas the last half became somewhat static as the latter discussed motivators are of less influence on the developer. Interviews roughly lasted between 24-60 minutes. The interview time also depended on the interviewees' available time during their work day. Conversations occasionally turned more personal than expected, and the developer was asked again for permission to use all of the data after the interview.

5.4 DATA ANALYSIS METHODS

This section discusses the data analysis methods utilised to interpret the data. First, the choice of data analysis methods is described. Then, the procedure is elaborated, and initial coding themes are displayed.

5.4.1 Data Analysis Method Selection

A deductive coding approach is initially selected for analysing the collected interview data. The deductive approach is most suitable since the motivations from the literature are used throughout the interviews, meaning an initial set of themes can be denoted. These are the motivations covered in Table 8. In addition to the deductive analysis practices, thematic analysis is performed to identify additional themes that may arise. This second step is deemed necessary since it is unclear how the motivations of developers from other SECOs can be transferred to the PaaS SECO, and this study aims to explore these potentially unknown motivators. The procedures taken in the analysis are described below in the next section.

5.4.2 Qualitative Analysis Procedure

Braun & Clarke (2006) offer an approach to thematic analysis for psychology research. Since the construct of motivation is rather psychological, and the theoretical lens of SDT has its roots set in psychology, this approach was deemed most appropriate for data analysis. Therefore, the approach to thematic analysis is adopted as described by Braun & Clarke (2006), which consists of the following phases: familiarisation, generating initial codes, searching for themes, reviewing the themes, defining and naming themes, and lastly, producing the report. Below is a description of these phases and how they were executed in this study.

Familiarisation

Familiarisation is the first phase of the analysis approach and starts with getting familiar with the data and texts by reading them and taking notes to form initial ideas. The initial step was to finalise the transcripts automatically generated by the video conference tool through which the interviews were hosted and recorded. Additionally, anonymisation was applied while finalising the transcripts, as in some cases, the interviewees coined the names of their organisations and colleagues working there. Furthermore, after reading the interview transcripts, summaries were created, capturing the most relevant information offered in the interview. In addition to the textual summaries, a visual overview of the motivators per interviewee was created in Miro, as a graphical representation creates a better overview of the data. The order in which the interviews were assessed was per actor category. This measure was taken to familiarise with potential overarching concepts and similarities among the actor categories defined in Table 11.

Generating initial codes

This phase marks the coding phase of the analysis. The goal is to code interesting features of the data systematically across the entire data set. Hence, every interview received the same effort and attention. An initial coding round was done using the motivators listed in Table 8. Additionally, a second round of theory-driven thematic coding was performed. In this second coding round, the theoretical concepts from SDT were applied, and with these concepts in mind, the interview transcripts were coded. In this stage, it is recommended to code for as many potential themes and patterns as possible and add surrounding text to the code to include context and code extracts of data in multiple themes if relevant. Hence, this advice was followed throughout this phase.

Searching for themes

In this next phase, the codes are grouped into initial themes. Apart from the already existing motivators, the codes were reviewed and grouped into potential themes and sub-themes. While themes are formed in this stage, the existing motivators are reviewed to understand their significance. This phase ended when all codes were assigned to potential themes.

Reviewing the themes

This fourth phase was used as an additional round of reviewing themes and to understand their relationships to the codes and text extracts. As Sekaran & Bougie (2016) argue, coding is not a linear process and moves back and forth from data reduction, data display and data collection. This need to move back and forth was most present in this phase as themes were emerging. Hence, in this phase, the researcher moved back and forth from reviewing the themes to the interviews with deductive and inductively emerged themes.

Defining and naming themes

The last stage consists of defining what a theme is and what its boundaries are. This is the first step towards writing the report, as this phase aims to determine how the themes are defined. Also, in this phase, changes were made, and themes were analysed with the data. Illustrations of the coding tree, including examples of codes can be found in Appendix B.

Producing the report

The findings are presented in this last phase. Quotations from developers were used throughout the writing of the results. Several developers were asked for input and reviewed the emerging themes and motivators. In addition, developers were asked to review quotations and how they were applied to the findings' context to ensure validity. The final report of the findings can be found in Chapter 6.

5.5 VALIDITY AND RELIABILITY

Three criteria reflect the quality of an exploratory case study research. These are the construct validity, the external validity and the reliability. The internal validity is a criterion for research that attempts to establish a causal relationship between variables (Yin, 2009). Hence, this criterion is not considered. Based on Yin (2009), an overview has been created to address these criteria explicitly regarding threats and mitigating actions to ensure sufficient research and report quality.

Construct validity

Construct validity is one of the more challenging tests of case study reports as it can prove difficult to create operational measures free from subjective judgements. In this study, the motivation construct has been extensively researched to provide the best possible operationalisation. By applying a single theory, this study aims to keep the interpretation of the construct at a minimum. In addition to the applied theoretical lens on motivation, the motivators in other SECOs have been addressed and are used as guidance throughout the case study. This adds to the construct validity as concrete motivators, combined with the theory on motivation, will significantly reduce interpretation bias.

To ensure construct validity, multiple tactics are available. These are to use multiple sources of evidence, establish a chain of evidence and let key informants review the draft report. These tactics proved hard to apply as the study aims twofold: (Case study questions 1, 2:) to establish the motivators of developers in the PaaS SECOS, and (Case study question 3:) explore if these

differ per type of SECO actor. For the first objective, multiple sources of evidence were used as a diverse set of developers were interviewed to get a multi-perspective on motivations within a PaaS SECO. This means that, in this perspective, data triangulation is applied.

For the second objective, effectively deploying multiple sources of evidence would have been in the form of using various research methods. However, the sequential use of different research methods could not be established in the time frame. Hence, the construct validity of the second objective needs to be tested by having key informants review the draft. It was not possible to let developers review the full draft. However, the newly emerged motivators and differences between the actor groups were discussed, and input was asked from at least two developers per category. The feedback in the reviews has been processed afterwards.

Despite these efforts, Yin (2009), among many other scholars, stresses the need for using multiple research methods in a case study. As argued in section 5.3 on data collection, interviews are the best approach to initiate a theory on developer motivations within the context of PaaS. In this stage, simultaneously applying different methods, such as questionnaires or focus groups, would not add anything significant to the findings because it is still an exploration of potential motivators. Hence, using methods in a sequential fashion would be better suited. However, the time frame of the research does not allow multiple iterations using different methods. Hence, to mitigate the threats of this flaw, this research relies heavily on other strategies to enhance construct validity.

External validity
External validity shows whether and how case study findings can be generalised. Specifically, for single case studies, this can prove to be difficult. Theory should be used to improve the external validity of a single case study research. The external validity has been addressed by exploring literature and developing case study questions to direct the efforts of the research in the right direction. After the data analysis, the literature is revisited to draw a comparison and establish meaning.

Reliability
Reliability concerns to what extent the research is reproducible. By demonstrating that systematic procedures have been followed and are clearly explained, reliability in the study is shown and tested. This means minimising any biases and errors throughout the research and reporting. Consequently, a replication of the study should produce the same results. The case study protocol was used to enhance reliability, and systematic procedures for data collection and analysis were also put in place.

5.6 RESEARCH ETHICS AND DATA MANAGEMENT
Data collection from interviews is sensitive as the researcher collects data from human subjects. Considering human research ethics and legislation regarding privacy is crucial. Hence, the researcher drafted a GDPR-proof data management plan and a risk assessment and mitigation plan. A data management steward from Delft University of Technology reviewed the plan, and was approved by the Human Research Ethics Committee (HREC). The letter of approval can be found in Appendix C.

6 RESULTS

The following chapter presents the results and findings of the case study and answers the fourth sub-question: '*What motivates the individual developers in a specific PaaS SECO?*'. The execution of data collection is presented before answering this sub-question. Following that, the findings are presented. The uncovered overall motivators for developers to start and continue in a PaaS SECO are discussed first, followed by the differences in motivators per SECO actor category. Finally, this chapter presents an alternative perspective on how motivations differ between developers.

6.1 EXECUTION OF PARTICIPANT SOURCING

The Methodology chapter presented the intended sample design and the participant sourcing strategy. The attained number of participants is shown in Table 12. As can be viewed, quota sampling proved successful, and all quotas were attained. In the category of service partners, four participants were interviewed. This is because one participant was recruited from the Mendix ISV program, yet the developer turned out to be a consultant during the interview. To avoid researcher bias, the developer was not excluded.

Table 12: The number of intended and attained participants per SECO actor.

Diversification	Intended number of participants	Attained number of participants
ISV	3	3
Service partners	3	4
Customer		
Public organisation	3	3
Commercial organisation	3	3

Also, a new strategy for gathering developers was adopted due to the low response rate to invitations to partake in the study. Additional participants were sourced from the Mendix Developer meet-up events and the Mendix MVP program. The first are events hosted by Mendix for developers to network, socialise and learn more about Mendix. The latter is a program that includes the top 50 most community-involved developers. These developers contribute community-related content such as blogs, other media content, widgets and modules, hosting meet-ups and knowledge-sharing on the forum in return for insights into the Mendix strategic roadmap and short lines of support to R&D. Roughly half of the participants were recruited from these sources. This alternative approach predisposes some bias in the results. These developers are top contributors to the community, which could lead to some bias in favour of the SECO regarding the responses to the interview questions that regarded the influence of the developer community.

The final list of participants is shown in Table 13. Despite any efforts to gather participants that all fulfil the selection criteria, one included participant did not have the minimum required certificate. This is Developer D3. However, due to his prior knowledge of IT and experience in high-code, his experience was deemed sufficient. Hence, this does not affect the results.

Developer	Experience (years)	Certificate	Actor Type	Firm
D1	2	Advanced	Public Client	A
D2	3	Advanced	Public Client	B
D3	3	Rapid	Public Client	C
D4	9	Expert	Service	D
D5	6	Expert	Service	E
D6	5	Advanced	Service	F
D7	3	Intermediate	Service	G
D8	7	Advanced	Private Client	H
D9	4	Advanced	Private Client	I
D10	3	Intermediate	Private Client	J
D11	8	Expert	ISV	K
D12	10	Expert	ISV	L
D13	7	Expert	ISV	M

Lastly, some developers have switched jobs during their Mendix career, meaning that some have switched from clients to service providers, while others switched from client to consultant. This has been used to the advantage of the research to gain insight into why these developers switched careers to another type of actor and how this reflects on their motivation. In this list, D8 and D9 are former service providers. Also, D4 and D6 are former Mendix employees.

Furthermore, it was found during the interviews that many ISV actors simultaneously also do consultancy work. Developers D11 and D12 do both. D13 is the founder of both a consultancy firm and a SaaS firm.

6.2 DEVELOPER MOTIVATIONS TO START AND CONTINUE PARTICIPATION IN THE SECO

As described in the data analysis methods section (section 5.4), the interview data were analysed in two rounds. First, deductive coding was applied using the motivators found in the literature as themes. Then, a theory-driven inductive coding round using SDT concepts was conducted to allow new motivators and motivation-related themes to emerge. In the following section, motivators denoted by "_M#" refer to the initial tentative motivators found in the literature and displayed in Table 8 of section 2.2.5. Motivators denoted by "M#" refer to the motivators found in the case study for the PaaS SECO context.

Table 14 contains the motivations found in the case study of the PaaS SECO. Almost all tentative motivators that were found to be of importance in the MSECO were also found to be relevant in the PaaS SECO. However, slight differences were observed. The list of motivators counts two additional motivators: (M3) the ability to create apps rapidly and agile and (M8) recognition, esteem and status from the work. Additionally, _M14 was extended to (M14) the perceived presence, attitude and reputation of the orchestrator in the SECO towards the developer' as this captures the significance of the overall developers' perception of the orchestrator better. (_M15) Creating value for end-users has been extended to include the value created for all stakeholders with one's work, as this better captures the value for whom developers aim to create value. Finally, (_M13) influences of one's social circle (both local and

online) was separated from the list of motivators because it was found to be ineffective. Rather, it is closely related to the organisational context facilitating motivation. As the organisational context is highly separate from the SECO, this subject is addressed separately from the motivators.

Table 14: Definitive list of motivators for the PaaS SECO.

Item #	Motivational factors
M1	Fun experienced throughout development process.
M2	The work being intellectually stimulating.
M3	The ability to create apps rapidly and agile.
M4	The acquisition and improvement of skills.
M5	To contribute to something new.
M6	Financial gains and rewards from the work.
M7	To increase the position on the job market.
M8	Recognition, esteem and status from the work.
M9	Creating a reputation in the developer community.
M10	Identification and commitment to the development community.
M11	Attraction to the size and quality of the (online) developer community.
M12	The ability to exchange knowledge with other developers.
M13	Participating in social community events.
M14	The perceived presence, attitude and reputation of the orchestrator in the SECO towards the developer.
M15	Creating value with one's work.
M16	Satisfaction of the developer with their SECO.
M17	Gaining the ability to self-manage work.

The remainder of this section goes through each of the motivators in depth. Intrinsic motivators will be discussed first. A discussion of extrinsic motivators follows this. Lastly, the findings are briefly summarised.

6.2.1 Intrinsic Motivations

The intrinsic motivators relevant to the context of a PaaS SECO are (M1) fun experienced throughout the development process, (M2) the work being intellectually stimulating, (M3) the ability to create apps rapidly and agile, (M4) the acquisition and improvement of skills, (M5) contributing to something new, (M10) identification and commitment to the development community, (M13) participating in social community events, (M15) creating value with one's work and (M17) gaining the ability to self-manage work. Of this list of motivators, (M3) the ability to create apps rapidly and agile has been added based on the results of the data analysis.

These motivators are not fully stand-alone but overlap and influence each other. This is because most of these motivators come from satisfying the same basic psychological needs posited by SDT. Motivations such as M2, M3, M4 and M5 are all influenced by the level of competence and challenge posed by the platform to the developer. The satisfaction of the need for competence thus influences M2, M3, M4 and M5 influence (M1) the fun developers have throughout the development process.

M1 – Fun experienced throughout the development process
Fun is a classic example of an intrinsic motivator (Deci & Ryan, 2004). Many developers stated (M1) fun as one of the most important drivers to start and stay in the Mendix ecosystem. One developer stressed the importance of fun by stating:

"I would not be where I am if I didn't have the fun I am having" (D5)

Fun was, however, mentioned in a broader context than merely experiencing fun throughout the development process. This also becomes evident from the above quote by developer D5. Moreover, some developers question whether they experience fun throughout the development process. Several developers refer to the development process as 'solving the puzzle', in which fun is experienced upon completion of solving that 'puzzle'. Since a software application can be built instantaneously, the direct gratification of seeing one's finished work is seen as fun. At the same time, these developers feel doubtful whethr there is any fun they experience throughout the development.

"I don't think that developing itself is something that I would call very fun, but it's part of the process of reaching the fun. So, I always say, programming or creating a program [is like] solving puzzles, and I'm constantly solving puzzles, and the moment you have finished that puzzle is, of course, very satisfactory that you can be proud of the fact that you have solved your puzzle, but is the process of figuring out to pass all fun... ...It's not the main part of the fun. It's the trajectory to the fun." (D3)

Such an analogy of 'solving the puzzle' stresses the importance of the need for competence in a developer's work. Competence should be fulfilled so developers feel challenged in mastering their development skills. This is also reflected by how some developers mentioned not having 'fun' initially as learning the Mendix modelling language proved difficult, while others felt not challenged at all. The first group of developers deemed fun as not being present, while the latter group reported that fun was important. For the first group, the transition from theoretical learning material to practice is especially difficult. As confidence in one's abilities grows, the fun experienced also grows. This shows the importance of correctly fulfilling the need for competence since too much challenge will lead to a deprivation of competence, resulting in extrinsically motivated behaviour to continue in the SECO. At the same time, the right amount of challenge results in fun being present from the start. Furthermore, this also increases the importance of intrinsic motivators over time as confidence grows.

"And I think, well, how am I going to solve this? This is hard. But then at the end of the day, I'm like, oh, well, I did it. That's a good balance." (D2)

M2 – The work being intellectually stimulating
The reference to challenge through the analogy of 'solving the puzzle' suggests overlap with (M2) the work being intellectually stimulating. Therefore, this motivator strongly correlates with M1. Finding intellectual challenge in one's work best reflects how developers describe the importance of intellectual stimulation. Developers need this intellectual challenge to stay engaged and avoid a "bore-out" (D5). This also reflects the importance of continued challenge throughout their career to start and stay engaged.

"I would get bored otherwise in my daily work, and being bored with me.. [being bored] would mean I'm not motivated. And then work won't be fun. [Work] would be more like a chore. So, for me, it definitely needs to be, yeah, challenging." (D9)

This challenge manifests in two different ways. First is the emphasis on the analytical side of development, meaning that the functionality and requirements and making something the business demands were important. As this requires both development skills and knowledge and good communication with the involved stakeholders, this part of M2 fulfils competence and relatedness. Second is the technical implementation of what is demanded by the business. This refers to how the developer interacts with the platform and the intellectual stimulation a developer gains by thinking of and implementing a design. The fulfilment of competence solely facilitates this part of the intellectual challenge.

Intellectual challenge can be considered an intrinsic motivator as it is strongly related to the satisfaction of the need for competence and contributes to the fun.

M3 – The ability to create apps rapidly and agile
(M3) The ability to create apps rapidly and agile has been added to the list as many developers regarded it as stand-alone from the rest of the motivators, and it was mentioned often. This motivation is the effect of one of the specific advantages of low-code. Due to the easy-to-learn platform-specific modelling language that makes application development more rapid and seamless, being able to work quickly, having the flexibility to adapt rapidly to changing requirements and creating new apps fast is referred to as the most important motivator.

Due to these advantages of the modelling language, the feeling of competence and mastering skills is enhanced. Furthermore, an increased sense of autonomy might also be perceived since the platform accommodates changing requirements, giving the developer a sense of control.

"So when the business all of a sudden wants something different … with high code, it can be like a deal breaker, but with low-code, most of the time, you will be maybe a little bit frustrated because you've spent some time on it. But most of the time, you can also fix it quite fast." (D2)

At first glance, this seems contradictory to (M2) the work being intellectually stimulating as this would suggest that all form of challenge has been taken out of the hand of the developer by the ease with which applications are created. In turn, this leads to a deprivation of the fulfilment of competence. Contrarily, the emphasis on the analytical aspect of software development poses new challenges to the developer. M3 facilitates intellectual stimulation by removing the pitfalls of high-code, such as bug fixing and database design, that deprive competence and allow developers to focus on business-related aspects of app development. This also facilitates (M1) developers' fun throughout the development process.

As M3 enhances self-determined behviour as it contributes to competence and results in fun developers have, it can be considered an intrinsic motivator.

M4 – The acquisition and improvement of skills
(M4) To acquire new skills is also considered an important drive. Developers refer to this aspect in different senses. Developers starting their careers focus on finding an engaging and fun job and are motivated to grow personally. In SDT, this is referred to as intrinsic life goals and is more closely aligned with the satisfaction of the basic psychological needs. However, developers who are further in their careers see this as necessary for expanding their knowledge and skill set as part of their work or staying relevant in the job market. In the latter case, the acquisition of skills is rather extrinsically motivated. More specifically, as developers view this aspect as a part of their role as developers or out of fear of becoming irrelevant, this can be

related to as identified regulated or introjected regulated extrinsic motivators, respectively. Nonetheless, both views can be described as autonomous motivation.

To stay engaged in the SECO, developers rely heavily on the continuous developments within the platform. The innovations that progress the platform's capabilities to the developers are considered important several times and keep the developer engaged and motivated. This aspect of improving skills correlates with (M14) the perceived presence of the orchestrator since the delivery of such features and how it is presented plays a role in this as well. While some developers refer to keeping up to date as 'fun', others refer to it as part of the job and a necessity to stay relevant on the job market, meaning that this motivator can be perceived as intrinsically and extrinsically motivated.

"When Mendix is releasing these new features. They serve it to you like a party, for example, the Mendix version 10. And they announced it very big, and they had all these cool new things. So yeah, it's not.. how do I say it? It's just nice to follow along with what Mendix is doing, yeah." (D2)

"I think it's also good for for yourself to get the knowledge. And it's also fun. I mean, using a new module, getting to understand it, getting to implement it in the in the application. It's fun." (D5)

M5 – Contributing to something new
(M5) Contributing to something new was mostly interpreted as contributing to the creation of new apps. Building apps from scratch, using creativity and putting effort into creating a product or module is considered rewarding. Furthermore, developers find joy and a sense of accomplishment to deliver a finished product or a module. Having a creative outlet and building new apps is important to start and stay in the SECO.

"I do like, for example, starting a new app and something from scratch. It's really stimulating that first few weeks where you start from nothing, you start creating something." (D9)

"If that was just a maintenance job, if I'm very honest, then I don't think I will.. I can't imagine that I will do that for several years." (D12)

This also overlaps with previously mentioned motivators because competence is the important factor being fulfilled in this case, as creating something new requires (M2) intellectual stimulation, and both influence (M1) the fun developers have. As it stimulates self-determined behaviour and strongly results in sincere enjoyment, the motivator is considered intrinsic.

M10 - Identification and commitment to the development community
Concerning the motivators relating to the developer community of the SECO, most developers were unaware of the developer community activities and livelihood. Hence, to join a SECO, (M9) creating a reputation in the developer community, (M10) Identification and commitment to the development community, (M13) participating in social community events and (M14) the perceived presence, attitude and reputation of the orchestrator in the SECO towards the developer proved not important.

Few developers mentioned feeling a high (M10) identification and commitment to the development community. Whereas most said no when asked whether that form of relatedness was important for them to stay. Most developers consider it a "perk" or a "big plus" on top of

their work. On a personal level, it does add to their social life, especially for the developers who visit many meet-ups. Hence, some form of identification with co-developers in the community is relevant for sharing experiences and networking.

The starting point for developers who feel such identification and commitment to the community, the majority of whom are MVPs, is to attend physical events and get in touch with the orchestrator. The online surroundings of the community does not provide a foundation to create a strong connection to the community.

"So I went to this quick course at Mendix office and once I talked to people and I made this connection, I started to go to all the meet-ups and hackathons and stuff like that. And yeah, it's now it's so fun to go there because, you know, people know each other" (D6)

Identification contributes to intrinsic motivation as it is done purely for enjoyment. Also, this feeling of identification with the community increases their satisfaction with relatedness.

M13 – The influence of participating in community events
(M13) The influence of participating in community events is regarded as important for staying engaged and being involved for several reasons. First, talking to like-minded people with a common passion is considered fun. Second, developers can take inspiration and increase knowledge from the events, depending on the planned activities. Third is that events are a good opportunity to network. The inspiration developers take from these events, and the satisfaction of the need for relatedness will motivate developers to stay engaged in the SECO. In most cases, developers gain new knowledge and abilities and will increase their ties with the developers in the community.

"It doesn't, didn't really influence me to become a mendix developer, but it does help me or, or yeah, inspire me to stay a developer. ... Because it's, yeah, how you say that.. It's motivating to see how much energy is put into it and share it with everyone" (D8)

On the other hand, while other developers also stress the fun of these activities and meet-ups, the absence of such events wouldn't lead to leaving the SECO. Hence, this motivator is not considered as crucial for most.

"I really enjoy being active and read the forum being in the meetups and meeting other mendix developers. But for me, it's not like something that motivates me in keeping staying a Mendix developer, but I would miss it but it's not [crucial]" (D9)

As developers are self-motivated to go to events and are considered as contributing to the fun and enjoyment of being in the SECO, this motivator is intrinsic.

M15 – Creating value with one's work
(M15) Creating value is also one of the most important drivers for developers, as this is the ultimate goal of development. This motivator was changed from creating value for the end-users to creating value with one's work as different developers aim to create value for different stakeholders. This is elaborated on in detail in section 6.3.1.

In general, this motivator is considered crucial as it provides developers with feedback on their work that provides a sense of accomplishment and fulfilment. This increases their well-being, a central theme within SDT and a way to increase motivation, meaning that it is considered intrinsic. Most developers mention this as the most important driver to stay engaged in the

SECO. Of course, this motivator is not specific for this SECO but for software development or most IT projects. Regardless, the importance of this motivator makes it worth taking into account since it is tightly bound to the actual software development done on the platform.

"For me, it's good to see that I can deliver things that they are happy with." (D2)

To join the SECO, (M15) creating value is less important as starting developers are less aware of the importance of end-users and other stakeholders within the development process. Subsequently, this motivator becomes increasingly important to stay. Acknowledgement of what value their work adds is vital and essential for their well-being. This recognition differs from (M8) recognition, esteem and status from the work since it is not esteem or shame related but increases the feeling that their work is valuable and important. This type of recognition strongly reflects and enhances intrinsic motivation.

"The most satisfactory moment I have within my work is the moment that you have delivered a product which is being used by other people and that you can take pride in." (D3)

"I really love when we are giving a demo and you see the business talking to each other and saying, 'yeah! That was what we needed! They built it, yes!' And it's great. It's just great." (D5)

However, in some cases attempt to create value for other types of stakeholders, such as the business, which makes the motivator extrinsic as well. In what cases this motivator is intrinsic or extrinsic is also highlighted in section 6.3.1.

M17 – Gaining the ability to self-manage work
(M17) Gaining the ability to self-manage work was an important motivator to stay in the SECO, while not so much to join. Developers lack awareness of the self-management possibilities within the IT context. This motivator reflects the satisfaction of the need for autonomy and can therefore be considered intrinsic as it leads to self-determined behaviour. Developers referred to the methodology rather than the SECO that provided such self-management capabilities. As with most IT projects, agile methodologies such as Scrum are utilised. These agile project management methodologies give developers an increased sense of control, reflected in their intrinsic motivation, as this fulfils their basic psychological need for autonomy (Memeti et al., 2021).

"Nowadays, you work agile in your team or [at least] kind of agile. That means that you have freedom and I really like the freedom. Definitely. … later, I also learned that it really suits me too because I'm a bit chaotic in nature. So having this framework, having also the liberty to work as you want. I really like it. Yeah. No, for sure. For me, it's probably one of the best perks of working in software. Yeah." (D9)

It is somewhat questionable whether this is a motivator to join a PaaS SECO, since many developers agree that the SECO platform does not per se provide this, but rather the methodology or the organisational context the developer operates in. However, taking away this perk of being in the SECO would lead to decreased motivation to stay in the SECO, especially if competing SECOs integrate this aspect better. In support of this, one of the developers stressed how well Mendix integrates the agile way of working. Hence, it is adopted in this list.

6.2.2 Extrinsic Motivations

Apart from M4 and M15, the motivators identified as extrinsic are (M6) Financial gains and rewards from work, (M7) To increase the position on the job market, (M8) Recognition, esteem and status from work, (M9) Creating a reputation in the developer, (M11) Attraction to the size and quality of the (online) developer community, (M12) The ability to exchange knowledge with other developers, (M14) The perceived presence, attitude and reputation of the orchestrator in the SECO towards the developer, (M16) Satisfaction of the developer with their SECO.

M6 – Financial gains and rewards from the work
(M6) Financial gains and rewards from work and (M7) increasing one's position on the job market are classic examples of rewards that result in externally regulated extrinsic behaviour. Salary is not a high-weighting factor for developers to join or stay in a PaaS SECO. Contrarily, when asked if financial gains were important, almost none reported it as such.

While salary and financial types of rewards can be referred to as externally regulated extrinsic motivational factors, studies in SDT in the context of work organisations provide insight into the effect of financial rewards on the intrinsic motivations of developers. While much research is still needed on the effect of salary on intrinsic motivations, an increase in a base salary, i.e. salary not based on performance, will positively affect autonomous motivation as employees feel more valued by their employer. In short, on an organisational level, salary can positively impact the developer's motivation to join and stay, leading to increased performance and quality (Deci et al., 2017). From the SECO developer perspective, salary is not a motivator for most, given that a competitive salary is provided.

"The money is not the most important part of it. The fun being able to help people being able to learn those are the main things. Because I know that the financial side is already covered by anywhere." (D5)

However, there are two scenarios in which (M6) financial gains and rewards from the work do become important. The first is in terms of what the minimum should be. Developers who mentioned finding financial gains unimportant said that the salary should be sufficient to accommodate the living standards that the developer aspires to. Second, in case of a change in salary, for example, by getting an offer at another organisation or if the developer profession undergoes a devaluation, the importance of M6 increases. However, not for all developers, this is the case.

Furthermore, one could argue that the financial rewards are not attributed to the SECO. However, the value or health of the SECO and the height of the salary must be somewhat related: A fluctuation in the health of the SECO would result in a change in the value of the SECO developers, potentially impacting the wages. In this sense, financial gains do become an important motivator.

M7 – To increase the position on the job market
(M7) Increasing one's position on the job market was rarely mentioned as a driver to join the SECO, because most developers had not previously encountered the concept of low-code. Several developers spoke regarding their career switch to IT in general since this sector is perceived as "future-proof" (D6) regarding job security, yet this was before they decided on low-code. For one developer, low-code is perceived as the first logical step to pursue a career in high-code. In addition, two developers with a background in business mentioned that

'software developer' provides a clear job description compared to most business role titles. Though, followed up, stating that this was not a driver for becoming a developer.

On the other hand, one developer reported joining the SECO as somewhat risky due to low-code at that time still being niche. Furthermore, other developers mention that if demand for Mendix developers stagnates, there would be plenty of opportunities elsewhere in the IT sector. This guarantee of jobs within the IT sector also reflects on developers' attitude towards (M6) financial gains and rewards from work, as this could explain why finance is considered less important.

"So it's not really am I going to have work? I know I'm going to have work. I mean, there is work everywhere. Everyone is hiring." (D5)

M8 – Recognition, esteem and status from the work
The second motivator newly added to the list refers to (M8) the recognition, esteem, and status developers get from the work. As this motivation plays into the ego, esteem and shame-avoidance, this motivation falls into the introjected regulated extrinsic motivation category. This type of recognition differs from being valued by your peers and getting recognition for your work, as the latter fulfills one's need for relatedness, while the first is to satisfy one's ego. This overlaps with (M15) creating value, since the feedback on work positively affects one's ego.

"It's like an ego boost, right? You build something nice. People clap. People are happy when life and everything is good. Yeah, that's just like a really good feeling. So whenever you achieve that it's, it's definitely a strong motivator the next time." (D9)

"People are just really happy you're, you're like a magician for them, and that really makes, yeah, it makes me really happy." (D1)

One of the female participants expressed that due to her gender, she feels a need to prove herself and her technical knowledge and capabilities.

"So my technical side is not really taken seriously. So maybe that's why I'm more eager and motivated to prove myself and prove others. That I can be technical and I am technical and yeah, I can surprise them." (D6)

Another developer mentioned that social status within their own social circle also played a role to get into IT.

"It's a nice occupation in the sense that in your social circle, if you say, oh yeah, I'm a software developer. It's, it's, it's like a cool thing to say for sure. So, in that sense, I wouldn't say I did it because of that validation from people around me, but that helps to contribute. And if everybody thought that occupation sucks, I'm not sure I would have been a software developer. I might have been influenced to be something else." (D9)

However, most developers did not explicitly share the statement made above. This motivator increases over time as developers gain experience and expertise and contribute more to projects.

M9 – Creating a reputation in the developer community
As mentioned earlier, developers are not aware of the community. To stay in the SECO, the significance of these motivators remains relatively low. (M9) Gaining a reputation in the

developer community is unimportant since developers do not seek that type of acknowledgement from a community. Compared to M8, this motivator does not per se take place in one's direct surroundings, but more in the (online) developer community.

One developer reported that such a reputation results from self-determined social behaviour within the community. In such a case, it is arguable that creating a reputation is the underlying motivator.

"I don't think it was a drive to have a reputation. I think it's what I've always done. From a young age, I've always engaged with people I didn't necessarily know, and not especially with the idea of building a reputation. It's more something that I do automatically. ... Because I'm a social person." (D5)

Interestingly, only one of the MVP or expert-certified developers reported the reputation as important. For some, getting certificates is part of the job, meaning that it is a requirement for the job instead of a behaviour motivated to elevate one's esteem. Also none of the developers mentioned the reward system, in which developers gain points by contributing to the online forum, while mentioning reputation. Others say there is plenty of work, so there is no reason to stand out in the community.

As this motivator alligns with how developers see their jobs, this motivator can be referred to as an identified regulation type of extrinsic motivation.

M11 – Attraction to the size and quality of the (online) developer community
Compared to M9, M10 and M13, (M11) the size and quality of the developer community are considered relatively more important since the quality reflects the amount of support a developer can expect on, for example, an online forum. This is important during the learning phase the developer undergoes. The need to be part of a community is less present for this SECO. One of the reasons is that developers also have their own "community" within their own organisation.

Two developers mentioned that the community's growth is a better metric to join and stay since this indicates that developers aren't "investing their time and effort in something that might fail" (D5). In addition, two developers said that the attraction to the community would also be less attractive if the community were too big, and a small community increases the feeling of being connected.

"The community is, I would say, big, but not as huge as you would expect with, for example Java, or something else. So that makes it maybe even more easy to connect with other developers. ... Because the community isn't that big, you pretty much learn a lot of people pretty fast." (D2)

A surge in the size of the community also causes the perception of the quality of the community to decrease by the more experienced developers. The reason for this is that the new developers add low-quality contributions to the forum, which results in demotivation by experienced developers to partake actively in the community forum as the satisfaction of competence is decreasing. On the other hand, less experienced developers can identify with the difficulties and problems that the starting developers encounter. This increased relatedness, they feel, motivates them to be actively involved in answering these questions.

From an SDT perspective, becoming part of a sizeable, high-quality community is extrinsically oriented. As developers feel the need to be part of a growing community to avoid being part of a failing SECO, this can be referred to as an introjected regulated form of extrinsic motivation as it plays into the fear of failure.

M12 – The ability to exchange knowledge with other developers
Additionally, (M12) the ability to exchange knowledge with developers is highly associated with the (M11) quality of the developer community. Widgets and modules on the marketplace and knowledge on the forum are considered types of knowledge. As mentioned, knowledge exchange, in the form of knowledge consumption, is important in joining and starting up in the SECO. Also, for experienced developers, support from the forum stays important in some cases. However, this becomes less. In this case, it adds to the motivation to start.

In terms of contributing knowledge, results are scattered. Most developers value sharing and contributing knowledge mostly within their own organisations rather than online on the forum. As mentioned before, experienced developers feel over-qualified to answer easy questions. Additionally, several developers reported that sharing knowledge outside the company needs a commercial incentive.

"[Exchanging knowledge is] definitely important to me. … It's more like if colleagues got questions, then you can have a discussion about it. … It's not on a level of, like, sharing my knowledge online or being really actively contributing to anything like that" (D7)

"Why would the company that I work for pay me to create a solution [i.e. a widget or module] for somebody else? That's another company." (D3)

"But you work in a for-profit company, of course, right? So if you want to do something there needs to be a reason for that, to do that, or otherwise you have to do it in your own time or that sort of thing." (D11)

Since (M10) Identification and commitment to the development community was overall also perceived as less important, and developers overall do not identify to the community, the fulfilment of relatedness is somewhat low compared to the feeling of relatedness being fulfilled in the organisational context since developers have direct ties with their colleagues. The need for relatedness might yield a higher sense of fulfilment from the direct organisation than the developer community. Additionally, sharing knowledge internally in one's organisation is considered part of the job, meaning it is an identified regulation type of extrinsic motivator.

M14 – The perceived presence, attitude and reputation of the orchestrator in the SECO towards the developer
The community events are also highly associated with and related to (M14) the perceived presence of Mendix in the SECO. One important aspect is the perceived effort that the orchestrator puts into listening to developers and its reputation towards the developers. The presence and reputation are crucial as a positive perceived presence will lead to higher engagement and willingness to stay in the SECO, while a negative perceived presence will lead to lower willingness to stay in the SECO.

"So one thing that I find a bit frustrating about Mendix is that you can clearly see that it's a commercial company. So the main drive is not the best platform or the best tool for a developer, but the main drive is the fact that they need to turn a profit. And it's a very good tool for to sell to a manager, but it's not a very good tool to sell to a developer." (D3)

This quote from the interview of developer D3 demonstrates how a developer's motivation can be impacted by frustration caused by how they perceive the orchestrator. This is echoed by several developers as there needs to be some form of "respect" (D5) radiated by the orchestrator's reputation and effort that the orchestrator should put into contact with developers.

"It's already a big company, but they are still very accessible like in a meet-up or at Mendix World or in online gathering presentations. And there's also always room for asking questions and, and, and the people working at Mendix, they, they are really keen to know their customer. So, and because they ask those questions and I am allowed to give that input. I feel, yeah, there's a connection." (D8)

Furthermore, developers value the orchestrator's participation in community events, meet-ups, and the forum. Additionally, developers' perception of the orchestrator is also reflected by the innovations and perceived effort the orchestrator puts into developing the platform and provided tools. This demonstrates how the orchestrator's presence can increase the fulfilment of the needs for relatedness and perhaps autonomy, increasing motivation to stay in the SECO. Overall, this motivator is crucial for most developers.

As the orchestrators efforts and reputation should be in line with the values that the developers hold, this motivator is considered an identfied regulation type of extrinsic motivation.

M16 – Satisfaction of the developer with their SECO
(M16) Satisfaction with the SECO is not regarded as an important motivator to start, yet gains importance over time while the developer settles in the ecosystem. This motivator is mostly regarded as to the technical capabilities of the platform, how the platform is evolving and the support in terms of documents and tools that complement the platform. In this sense, M16 is also related to (M14) the perceived presence and reputation of the orchestrator since this also adds up to the general image a developer has of the SECO.

"A lot of time and effort is going into it. So seeing that happen is definitely a motivator and I mean, I wouldn't bet on a dead horse if I see that the day I see that Mendix is just milking it and not, not innovating anymore. For me as a platform, it's not interesting anymore to work for it because I'm investing time and effort and learning something that's not gonna grow anymore." (D9)

For most developers, this motivator, especially the technical dimension of the SECO, is crucial to stay as it is a technology they have to work with daily.

As satisfaction with the SECO has to align with the values of what developers find important in their work, this motivator is considered an identfied regulation type of extrinsic motivation.

6.2.3 (_M15) The Stimulation Through the Influences of One's Social Circle
The developer's direct social circle greatly influences what SECO the developer joins, as most of them learn about Mendix through their social network or organisation. However, this can not be regarded as a motivator, as their social circle serves as a channel to learn about the SECO, rather than a direct motivational reason to join a SECO. Interestingly, while it is not an answer on 'why' developers join a low-code PaaS SECO, it does answer the question 'which' SECO they join. Most developers opted for the SECO introduced to them by their organisation or their social circle. Although, none of the developers felt pressure from their social circle or organisation to join the SECO. Hence, it is removed as a motivator.

Also, while the direct social circle, in the form of their organisational context, greatly influences the developer's well-being in a later stage, this factor seemed less important in relation to the SECO. For example, the influence of a developer's social circle manifested into (M8) the status within their social circle that developers gain from the work by being a software developer. Furthermore, many developers regarded knowledge exchange as important within their organisation. However, while both items come from the developers' social circle and influence the developer, it has not emerged as a separate motivator.

This does not mean the organisational context is unimportant to the developer. In contrast, many developers report that the organisation they work for is important. Having supportive and fun colleagues was mentioned several times. For example, experienced colleagues have a big impact on the learning curve of the developers and can impact the need for competence and relatedness.

"We have a very experienced Mendix developer on our team. He's been working with Mendix for, I think, over 12 years now. ... So he's like a walking encyclopaedia with a lot of information about Mendix. He told me all the basic knowledge, but also a bit longer than it would normally take because I didn't really have a IT Background. So there was no pressure at all. Just, some guidance from the mendix learning paths and some actual information from our colleague" (D2)

Also, having good dynamics in the development team can influence the well-being of the developer. In turn, this positively impacts (M1) the fun that developers experience throughout the development process and developer's willingness to (M12) exchange knowledge with their colleagues.

"Yeah, for example, it's a really nice team. ... We are, like, working together, like, two or three days a week. We are also lots and lots of time in the office together, like, [it is easy to] just talk to someone and say, 'Can you help me?', and they're also always in for it. So that's one thing I am also happy with" (D1)

Additionally, the developer's satisfaction with their organisation impacts their overall well-being. This satisfaction will keep them in their organisations and indirectly in the SECO.

"I'm sure that at [employer name] I gain less than what I would gain if I went to a rival company, because I'm one of the most experienced Mendix developer in Belgium. I have my network. I have my certificates. I am an MVP. So, yeah, I'm sure I could have more, but I'm really happy at [employer name]. That's the thing." (D5)

Despite the importance, the social circle or organisation itself is not a direct motive to join a SECO or to stay in the SECO. Hence, this is separated from the list of motivators. Still, the social dynamics within the organisation positively impact the need for relatedness, and experienced colleagues surrounding the developer can also facilitate a better satisfaction of competence if developers are still learning.

6.2.4 Mapping the Motivators

Based on the list of motivators in Table 14, Figure 3 on the next page shows how the distinct motivators align with SDT. In this figure, the motivators are categorised along the autonomy-control continuum of SDT. Above the columns, the types of internalisation, the extrinsic versus intrinsic scale and the controlled versus autonomous are shown. These are based on the evaluations of the motivators as discussed above.

As can be observed, seven motivators are found to be intrinsic and eight motivators are extrinsic. As discussed for the motivators (M4) the acquisition and improvement of skills and (M15) creating value with one's work can be intrinsic or extrinsic depending on the developer's interpretation and context. In addition, five motivators belong to the controlled motivation category, whereas the majority are considered autonomous.
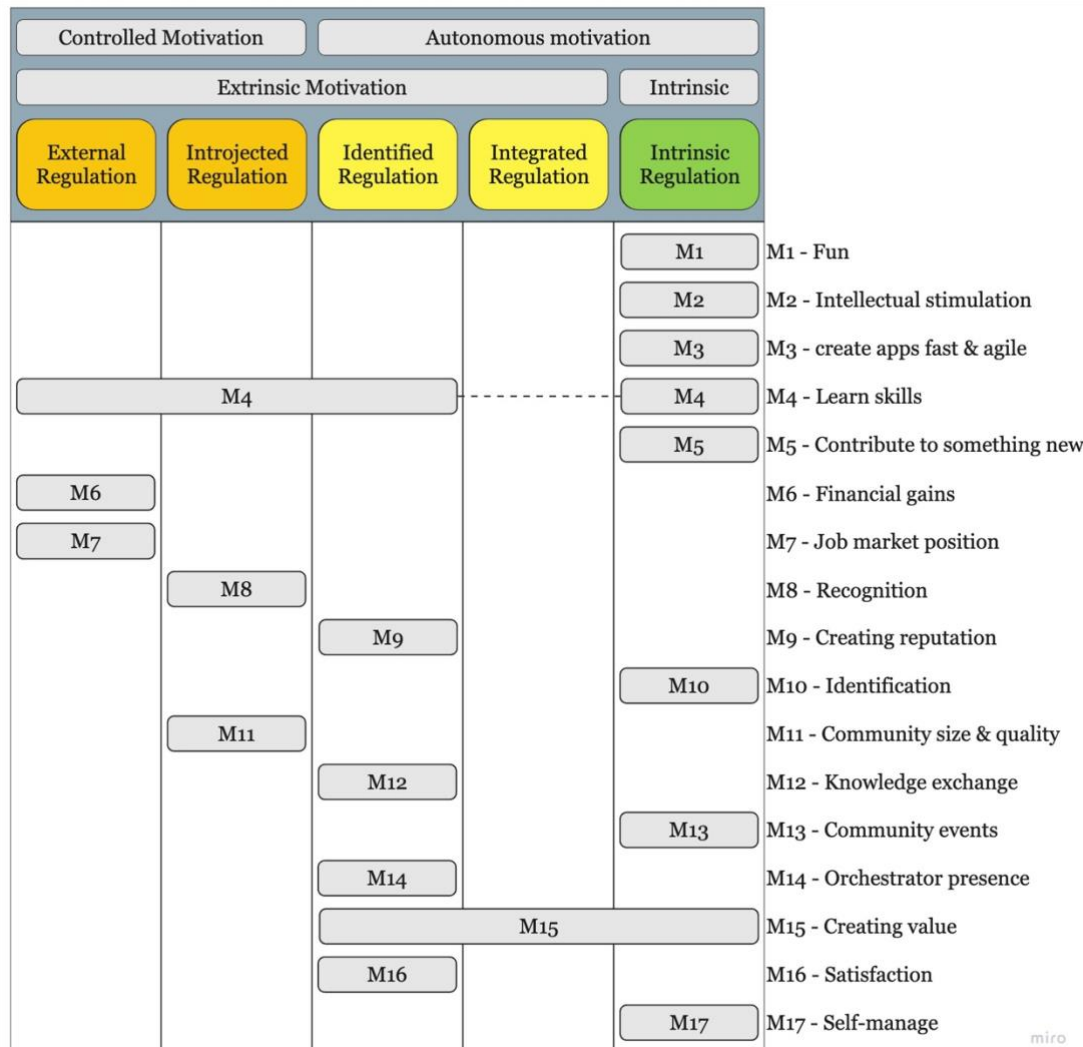

Figure 3: Motivators categorised along the SDT continuum of motivation.

6.2.5 In Summary
The first part of this chapter answers the case-study questions: '*What motivates developers to start participating in a PaaS SECO?*' and '*How do the motivators to continue in the SECO differ from those to start?*'.

Of the intrinsic motivators, (M1) fun throughout the development process, (M2) the work being intellectually stimulating, (M3) The ability to create apps rapidly and agile, (M4) The acquisition and improvement of skills and (M5) to contribute to something new proved to be relevant and important motivators to start. The feeling of being challenged, having a creative outlet and being able to craft apps that result in fun are critical in their work as developers. These motivators remain crucial over time. While most of these intrinsic motivators are related to the need for competence, two highly regarded motivators related to relatedness and autonomy, (M15) creating value with one's work and (M17) gaining the ability to self-manage work, respectively, are stressed as equally important. The latter two motivators are less

important in the start but become more important later when developers become more aware of end-users roles and the agile methodology.

The intrinsic motivators stressed as not significant are (M10) identification with the developer community, (M13) the influence of participating in community events, together with extrinsic motivators, (M9) creating a reputation in the developer, (M11) attraction to the size and quality of the (online) developer community, and (M12) the ability to exchange knowledge with other developers. Developers are unaware of the developer community before they participate in the SECO besides consuming knowledge from the platform, which translates to a relatively increased importance of M11 and M12 compared to the rest. M9 remains unimportant. Over time, M13 increases in importance, though it is still regarded as non-crucial as it is considered a perk of the job. Nonetheless, while not considered crucial, M13 has a positive effect since it can inspire developers and allow them to connect and share experiences with other developers.

Furthermore, (M13) social events hosted by the orchestrator partly increase (M14) the positive perception and attitude towards the orchestrator, together with platform innovations and efforts to create the best possible working environment for developers. While these social events are not considered a critical factor, the perception of the developers with the orchestrator somewhat is. Along with M14, (M16) developers' satisfaction with the ecosystem becomes an increasingly important critical motivational factor as developers want to be part of a successful SECO.

The remainder of the extrinsic motivators are considered less critical. Given that a (M6) competitive base salary is provided. Additionally, (M7) the developer's position on the job market was not considered an important motivator. Instead, some developers did pursue a career in IT in general. Overall, when developers believe there are many career opportunities in IT, this motivator becomes less significant with time. Lastly, (M8) the status and esteem developers gain from their job might impact some initially and increase in importance over time, but it is not critical.

The developer's social circle and organisation work as a channel to learn about the SECO. However, it does not directly form a motivation to start or continue. Furthermore, the organisational context is also important as it supports and recognises the developer's work, contributing to their need for competence, relatedness and autonomy. While separate from the motivators to start and continue in a SECO, the organisation also indirectly influences the motivators as well due to their influence on the basic psychological needs for competence, autonomy and relatedness.

6.3 DIFFERENCES IN MOTIVATION PER ACTOR CATEGORY

The main objective of this research is to understand how the motivations of developers differ per SECO actor category. As was discussed in the previous section, the importance and perceptions of motivators are rather nuanced, which makes it difficult to quantify the importance of a motivator, especially in relation to the rest of the motivators. Hence, the differences are covered qualitatively.

Furthermore, as discussed previously, overall motivators are alike for many developers since all aim to satisfy their needs for competence, relatedness and autonomy. For example, intrinsic motivators related to competence and autonomy were deemed crucial, while extrinsic

motivators were deemed less important. Some motivators were more nuanced as different developers interpreted them in different ways. A motivator such as the (M4) acquisition and improvement of skills was interpreted as an intrinsic and an extrinsic motivational factor. Consequently, these nuances form the differences between the motivators that developers have between these actors.

In terms of the actor activities in the SECO, another aspect that makes comparing categories difficult is that actor activities are different than anticipated, as they can overlap in some cases. Within the category of service providers, activities can differ significantly from secondment to consultancy. In the first, activities overlap with the actor category of customers since developers can be seconded to a single organisation for several years. In the latter, service providers solve IT problems through projects with their team of colleagues. Furthermore, many organisations that integrate an ISV business model also provide services, consequently leading to developers engaging in various types of activities ranging from working on SaaS, widgets or modules as well as client work. These unanticipated discrepancies impede the classification's usability, making it challenging to compare developer motivations.

Nonetheless, differences between the service providers, ISVs and customers were found. Several nuances were expressed for motivators M4, M15 and M16. Additionally, two themes arose from the analysis regarding the differences and the similarities of motivators between these groups. These are identification with the organisation and the pace of the work environment. None of these themes are tightly linked to direct attributes or characteristics of what the PaaS SECO offers. Contrarily, the themes are linked to the organisational context and the (commercial) activities the actors pursue.

6.3.1 Differences per Motivator

This overview provides how motivators were nuanced differently for the different actor types.

M4 – The acquisition and improvement of skills
Differences were found in the group of service providers for acquiring and improving skills. For this group, developers are more oriented around improving their skills, which is an important aspect of the job. Gaining certificates, for example, is more important as this increases the value of the developer for the consultancy organisation. In addition, the number of certificates they have is also reflected in their salary. Additionally, another consultant said it is important to differentiate himself from others by staying up to date and increasing his skills.

"Let's be honest, if you have an extra knowledge, you have an extra certificate, and it means that you're a bit more expensive for the client. [My employer] gets more money, and normally it should be reflected on your own pay slip" (D5)

In contrast, developers from customers and ISVs generally emphasised improving skills as fun and doing it out of enjoyment. As stated in section 6.2.1 regarding the intrinsic motivators.

Hence, compared to developers who do not work for a client, acquisition and improvement are more of an extrinsic motivator as organisations can press and reward developers for gaining additional certificates. In contrast, the other SECO actor-type developers experience this as an intrinsic motivational factor.

M15 – Creating value with one's work

Clear distinctions could be made in developers' motivations in creating value. Where the original motivator extracted from the literature was to 'create value for the end-user', this motivator was changed to 'create value with one's work' since this captures the diversity better regarding whom developers desire to create value for. This diversity for whom developers seek to create value highly differs per type of SECO actor. These differences are discussed below.

The service providers generally have no contact with the end-user but depend on the contact with the product owner. The product owner is one of the roles within the scrum methodology and is tasked with maximising the value of what is being built. To do this, the product owner manages all the stakeholders and subsequently manages which features and functionalities are built and in what order. In short, the product owner represents the end-users or the customer.

"The client is the representative of the end-user, so the client should know, and the client should also have a strategy and a vision." (D6)

"When I look at my current job. Then I don't speak to the end-users because there's a product owner ... So in that sense, there is no direct contact with end-users so that I can say that there's no value in that" (D4)

From a professional perspective, creating value for the business or client is the main priority since it is part of their job, as one of the developers describes. This professional view on creating value for the clients is the main priority. Also, by creating value for the client, the service provider aims to generate more sales. Hence, the consultant is always looking for more business opportunities.

On a more personal level, the developers also report that they gain satisfaction and enjoyment from the value they create with what they deliver to the business and feel highly motivated to help the customer further by solving their problems. This feedback offers affirmation, which, when transformed into enjoyment, boosts wellbeing and increases intrinsic motivation.

Along with adding value for the customer through development, increasing one's value as a consultant by developing one's skills was cited as crucial, as mentioned in the previous paragraph. Another developer who mentions that the customer comes first in some ways reflects this. This demonstrates the extrinsic nature of providing service centred on the client. More specifically, in this sense, this motivation is externally regulated. As a result, creating value for service providers tends to be driven more by extrinsic motivation.

Some similarities could be found with the service providers from the ISV type of actor. On the one hand, one developer reported creating value for the product owner as (s)he led the project. While creating value remains the objective, there is less to no contact with the end-users or product owners throughout the development process. On the other hand, one developer currently in charge of the SaaS development said that it can be rather vague as he wears the hats of the developer and the end-user at the same time. In this latter case, the absence of end-user feedback negatively impacts intrinsic motivation. Creating widgets or SaaS requires more technical skill and, consequently, more focus. Nonetheless, creating value with the delivered work remains highly important.

On the other hand, developers from the customer type of actor reported having direct contact with the actual end-users of the apps. Several of these developers reported that the

interactions with the application's end-user added to their satisfaction with their finished product. This increases well-being, meaning basic psychological needs would be fulfilled, leading to higher intrinsic motivation.

"For example, when you're testing a new application or, testing a new feature, you have a lot of contact with the end-users. And I just enjoy that part" (D2)

This relates to the basic psychological need of relatedness as it is highly focussed on the feedback developers get from whomever they develop apps for. Arguably, increased interaction with the actual end-user throughout the development process will lead to higher satisfaction with the need for relatedness. From an SDT perspective, this results in increased well-being and, consequently, more intrinsically motivated behaviour. In this aspect, developers from customers are more intrinsically motivated. In contrast, service providers and ISV developers are more extrinsically motivated.

M16 – Satisfaction of the developer with their SECO
While overall being reported as highly important by most developers, the satisfaction of the developers from the ISV and the customer type of actors report a business-driven consequence to this motivator. As the satisfaction of the SECO by the developer is partly expressed by the platform's strategic direction, these developers report this to be additionally relevant since SaaS products, for ISV developers, or apps crucial to business-as-usual, for customers, are fully dependent on the existence and longevity of the SECO. In this sense, the satisfaction with the SECO for these developers has an additional layer that makes satisfaction with the SECO more important.

6.3.2 Identification with the Organisation
In addition to the motivators that were found to differ between developers from different SECO actor groups, a theme that arose is for developers to see the lasting impact of the value they create when they develop an app or suite of apps. One key aspect is the organisational context and how they feel committed to the organisation they develop for, as depending on the relationship to the organisation they develop the apps for can increase their motivation.

For developers of customers in both the public and private sectors, one key aspect influencing their motivation is their relationship to their organisation. Developers report feeling more connected to the organisation, which leads to increased fulfilment of the need for relatedness.

Developers that develop internally, i.e. the customer developers from both the private and public sector, feel a deeper connection to their organisation, which is reflected in behaviour and development. This is reflected in their development as they feel more part of the 'whole' solution.

"You're making your own company better and that really gives some extra drive to, to put more energy into it, but also to think outside-of-the-box. So you're not only building this, this one application, but you know, this one application is part of probably other applications and then a part of the company that will be helped by creating this app." (D8)

This feeling of being more connected with the organisation increases the intrinsic motivations of these developers. In turn, this form of self-determined behaviour potentially affects one's productivity and creativity.

"You have to think about.. It is more than Mendix here. So basically, you have a whole data lake, you have plenty of other apps, you have to connect to them, you have to think about them. You have to be careful with the data that you publish. ... " (D9)

Again, this theme is highly related to the need for relatedness. In this case, the developers of the customer organisations are more likely to feel higher forms of intrinsic motivation due to their connectedness to their organisations. However, this is not to say that service providers do not feel this commitment to their own organisation. Contrarily, as mentioned before, overall, the organisational context influences all developers' motivation. Hence, the increased intrinsic motivations of the developers from clients do not per se reflect their commitment and willingness to stay in the SECO, but could increase the quality of what they produce.

6.3.3 Pace of the Work and Environment

Another theme that arose is the pace of the work and the environment, which reflects most upon the need for competence and autonomy. This theme mostly highlights the dynamic work of the service providers. In this theme, secondment as a type of service is excluded as during the respondent validation phase, service providers mentioned this not being relevant for that type of service.

Developers from the service provider type of actor group reported being in a more "exciting" work environment due to the diversity and dynamics of the projects they execute. As they do multiple projects for various clients, these developers are exposed to more types of industries, work environments and projects. This results in faster skill development. In this sense, M4 is experienced as more important as well since service providers aim to build up a skillset faster. It would also be expected to see apparent differences in M1 and M2. However, no differences were observed.

Additionally, due to the diverse client-oriented work by the service providers, it was established that motivation also differs per project. The motivators most impacted by this are (M1) the fun experienced throughout the development process, (M2) the work being intellectually stimulating, and (M15) the creation of value with one's work. In projects where M1 and M2 are experienced as low or non-present, developers will heavily rely on M15 as an extrinsic motivator since creating value for the client is seen as part of their work.

On the other hand, developers from clients can also report fluctuations in M1 and M2 and (M5) contributing to something new as projects finish and an organisation transitions from app development to maintenance and feature updates of existing apps. In this case, M1, M2 and M5 decrease, and developers also heavily depend on the connection to their organisation to stay motivated.

6.3.4 In Summary

To conclude this section, this summary answers the third case-study question: '*What are the differences in motivation between the developers of the SECO actors in a PaaS SECO?*'.

It is difficult to strictly separate the activities of the actors according to the developed classification, as the activities of some actors overlap. The customer-type activities overlap with the second type of consultancy. Furthermore, many ISV organisations also provide consultancy-like services, meaning they integrate two business models, and their developers perform various activities.

The activity of the actor does influence the motivation of developers. This becomes visible in motivators (M4) the acquisition and improvement of skills, (M15) creating value with one's work and (M16) satisfaction with the SECO. Service providers' work for external clients affects their motivation to be more extrinsic. In contrast, customer developers from public and private organisations who develop only internally for their own organisation show higher intrinsic motivation. M4 and M15 were interpreted more extrinsically by the service providers, whereas customer developers were more intrinsically motivated. For ISV developers, M4 is intrinsic, whereas M15 is slightly more extrinsic.

Differences related to the organisation and pace of the work environment also play a role. Developers from customers express a deeper connection to the organisation they work for, which translates to increased intrinsic motivation. Additionally, the work of the consultants is more diverse and exciting. Also, due to the dynamic work environment, developers of service providers learn skills more and faster. On the downside, the varied work also causes the service providers' motivation to fluctuate depending on how much they enjoy their projects.

6.4 DEVELOPERS WITH AND WITHOUT BACKGROUND IN IT

This section presents an alternative, interfering and rivalling exploration of how motivations can differ within a low-code PaaS SECO. According to Yin (2009), presenting potential rivalling theories increases the validity of the research of case studies. Since clear patterns could be established throughout the data collection and analysis phase, some interference in the presented results must be mentioned. Differences in motivation or in which motivation is expressed were found between developers with (professional) high-code experience and developers without prior high-code or IT experience. Five of the thirteen developers, scattered somewhat evenly over the actor categories, have prior high-code experience: D1, D3, D4, D8 and D11. The biggest differences were found in the need for competence and autonomy, which are elaborated on below.

Interestingly, what draws the high-code developers to the SECO are the advantages of low-code, which are competence-related, mainly (M3) the agility with which applications can be made, as they are not opposed by drawbacks normally occurring in high-code. For example, activities experienced as bottlenecks in high-code, such as bug-fixing, database design and setting up security, result in the deprivation of competence as it adds complexities not associated with (M1) fun (M2) intellectual challenges. Also, being impressed by the low-code platform's capabilities, performance, and stability plays a significant role as high-code developers initially have a sceptical attitude towards low-code. This results in curiosity and intrinsic motivators to discover low-code possibilities.

"It was fun also to see how easy you can build something. Especially like starting an application where it takes a lot more time in high code, right? Things like security setting that up in a good secure way or maintaining your database, all this kind of thing that used to be much more of a headache." (D11)

Another clear distinction that can be found in the need for competence is mostly reflected by (M2) the intellectually stimulating work and (M4) skills acquisition and improvement. First, it can be observed that developers who have a background in high-code do not feel intellectually stimulated by the technical implementation of app development. In one case, a developer even said that the platform aids in decreased (M2) intellectual stimulation. While another developer reported only being intellectually stimulated by the analytical aspect of the

development in low-code, there seems to be no technical challenge. Subsequently, these developers do not perceive they are (M4) acquiring new skills by learning low-code.

"I'm not sure if I can say that I gained a lot of new skills. It's maybe more deepening [of] existing skills. Let's say Mendix itself doesn't really help me to get new skills ... Mendix itself is not challenging enough to acquire new skills." (D4)

While both groups of developers regard competence-related motivational factors as important, for high-code developers, these motivational factors wear off as the urge and need for (M2) intellectual stimulation in the form of technical challenges arise. Hence, purely low-code fails to provide high-code developers with enough challenge in the long-term. To stay in the SECO, these developers require high-code possibilities.

"One thing I was looking for when I was starting in Mendix is if there were also any high code possibilities. I was a high code developer, as I said earlier and I still have the love for high code" (D1)

Additionally, one of the high-code developers felt overqualified for low-code, frustrating his need for competence and decreasing motivation. This feeling was amplified by the number of non-high-code developers the developer was surrounded by. High-code developers strive for technical intellectual challenges and control over the back-end processes. For the latter, in one case, this undermined the sense of autonomy that the developer has over the development process, frustrating motivation as well. Another developer agreed with this decreased sense of control over the back-end processes.

"I would have preferred to see a hybrid solution. So nowadays, I really enjoy the fact that I'm modelling my program, because it takes the problem-solving of created errors in your own code away. But I still would like to see sort of the possibility to write high-code. ... I would still like to see what is the code being generated in the back-end, because sometimes programs can be written more efficiently in high code than a low-code." (D3)

Developers who do not have a background in high-code have a somewhat opposite experience. In comparison, most report struggling to learn application development in low-code, especially while transitioning from theory to practice. In a few cases, this would lead to the absence of (M1) fun at the very start. However, the prospect of the enjoyment and fun that low-code software development would offer motivated these developers to continue.

"And then we got into a team where the lead was expecting too much from me. Not from my colleague, who was already IT trained and was technically more adapted. But for me, it was really difficult. I didn't have a burnout, but it was close." (D5)

After overcoming this first obstacle, these developers do not experience that the (M2) intellectual stimulation from the work becomes less throughout their time in the SECO. When mentioning intellectual stimulation, the developers enjoy the analytical and business side the most, as this is considered the "social side" (D9) of software development. This group of developers also strive for analytical challenges. However, compared to high-code developers, non-high-code developers also enjoy and experience fun in the challenges in the technical implementation of low-code software.

6.4.1 In Summary

This last section of the results chapter attempted to answer the last case-study question: *'How do the motivations of developers differ along the dimensions of competence, autonomy and relatedness?'*.

Developers transitioning from high-code to low-code have similar experiences within the SECO that differ from developers without high-code experience. A distinct difference can be observed based on the competence need. Especially in motivators (M1) fun experienced throughout the development process, (M2) the work being intellectually stimulating and (M3) the ability to create apps rapidly and agile. High-code developers are initially drawn to low-code due to its agility and reduced complexities in bug-fixing, database design, and security setup. However, they may find low-code lacking challenges in the continuance, a crucial competence-related factor. One high-code developer even feels overqualified and frustrated in low-code deu to the lack of control over back-end processes, leading to a loss of autonomy fulfilment. High-code developers desire a hybrid solution that allows them to work with both low-code and high-code to address their need for competence and autonomy.

On the other hand, developers without a high-code background can struggle initially when transitioning from theory to practice in low-code but do not experience a decrease in intellectual stimulation over time. They find satisfaction in the analytical and business aspects of development, even though they may face difficulties and stress during the learning process. In the long term, low-code poses enough technical challenges to stay engaged and challenged. Only a few non-high-code developers are looking for technical challenges that take them beyond the low-code realm and into the high-code realm.

Hence, in answering the case study question, along the competence and autonomy dimensions, a clear distinction can be made in intrinsic motivation between developers with and without high-code background. This manifests mainly in the possibilities and transparancy of the platform and the challenges it is able to pose to developers.

7 DISCUSSION

This section discusses the findings of the previous chapter. First, the main case study findings are presented in this chapter, along with an interpretation based on the results of the literature review and SDT. The outcomes of the sub-questions are then discussed. The generalisability of the findings is then covered. The findings' implications for theory and practice are then discussed. The chapter concludes with a discussion of the study's limitations and a suggestion for future research.

7.1 CASE STUDY FINDINGS AND RELATION TO THEORY

This section covers the case study's findings per the case study question and relates the findings to the literature used in the literature review and SDT.

7.1.1 Developer Motivations within a PaaS SECO

From the analysis, it has become evident that intrinsic motivators are more important than extrinsic ones. This aligns with studies from other proprietary SECOs and reconfirms similar findings of Koch & Kerschbaum (2014) and Steglich et al. (2019). While joining the PaaS SECO, developers have little experience or no knowledge of the SECO. The expectations they have of the platform's abilities and what it can provide, in the case of low-code PaaS SECOs, the ability (M3) to create apps rapidly and agile, (M4) learn new skills and (M5) to contribute to something new, is found to be most important. Also, (M2) intellectual stimulation and (M1) fun throughout the development process are important. On the other hand, intrinsic motivators that were less important to join and gained increasing importance over time are the ability (M15) to self-manage and (M17) to create value with one's work. However, these are low due to unawareness of the possibilities.

In comparison, the data shows that extrinsic motivators were considered relatively unimportant to join the SECO. The same is observed by Koch & Kerschbaum (2014) and Steglich et al. (2019) for MSECOs. In some cases, an extrinsic motivator is simply not that important to the developer, which is true for (M6) financial gains, (M7) position on the job market and (M16) satisfaction with the SECO. While these are considered relatively unimportant, developers deem it important to be part of a growing and successful SECO and strive for a competitive salary that accommodates their lifestyle. This means that the importance of these motivators will also vary depending on the health of the SECO.

In other cases, developers lack awareness, mainly for the motivators regarding the developer community and (M14) the orchestrator's perceived presence, attitude and reputation in the SECO. The motivators related to the developer community (M9) creating a reputation in the developer, (M11) attraction to the size and quality of the (online) developer community, and (M12) the ability to exchange knowledge with other developers, together with the intrinsic motivators (M10) identification and commitment to the developer community and (M13) participating in social community events ranked low for joining the SECO. Most of the developers are not aware of the community, its size and its activities. From these motivators, M11 and M12 were relatively more important as developers rely on the forum and help from peers. The importance of these motivators increases over time. The same is observed by de Souza et al. (2016) for the MSECO.

For the sustainment of developers, intrinsic motivational factors remain similarly crucial. For some, (M1) fun increases as developers become more confident in their abilities. In particular, extrinsic motivational factors become more important. Especially (M14) the presence of the orchestrator becomes crucial, and (M16) the satisfaction with the SECO. M14 is also deemed important by de Souza et al. (2016). The latter, M16, is viewed as less crucial in Steglich et al. (2019). The maturity of the Mendix platform could explain this, as it's less mature than the Android and iOS platform maturity. Mendix developers emphasise the platform's innovations as more crucial than the developers in studies of platforms that have reached higher maturities, such as iOS and Android.

The community also plays a more important role over time, though not as important as in the MSECO. (M12) Knowledge exchange, for example, is regarded as most crucial in Steglich et al. (2019) and de Souza et al. (2016), whereas in this study, it is not. This is unexpected as both studies include employed developers in their studies. One possible explanation is that the total developer communities of MSECOs include many private developers and hobbyists who are more willing and intrinsically motivated to share more knowledge, which could influence the overall knowledge-sharing culture of the community. Future research could follow this up.

Furthermore, the role of (M13) the community events as motivation to stay is questionable, given that many developers experience it as a perk rather than a must. However, given the inspiration, knowledge and newly established connections with peers, the events serve an important purpose nonetheless. For the developers who strongly identify with the community, the identification with the community starts by attending physical events. Hence, these events serve purposes that positively impact the SECO's developer community, for example, regarding knowledge and density of the social network. Even if only a small percentage of developers develop a strong identification with the community and start actively contributing, events can be called a purposeful success. Just as Constantinou & Mens (2017), a minority of core developers play a pivotal role in the contributions to the SECO. Lastly, if any actual employees or developers from Mendix company are present at the events, this significantly boosts motivation. Taking all into account, this research cannot conclude confidently on the community's specific role in the motivation of developers, especially since it plays a more prominent role in the MSECO as well (de Souza et al., 2016; Koch & Guceri-Ucar, 2017; Steglich et al., 2019). Similarly, this could be attributed to the difference in culture. This would be a fruitful future research avenue.

The newly added motivator M3 is a characteristic of low-code, though it reflects a crucial need of developers. Given that this is a characteristic specific to low-code, it is logical that this motivator was not found in the literature for other SECOs. Nevertheless, a developer's need for rapid and agile development goes beyond low-code. It could also be important to other developers in other SECOs as this facilitates competence, i.e. it allows the developer to focus on the intellectual challenges that they find interesting, which are the analytical and technical challenges of software development. As shown by Miranda et al. (2014), the complexity of the platform and SECO is not the same as the enjoyable challenges developers pursue and can hinder developers' participation in the SECO.

The same also counts for (M8) Recognition, esteem and status. This goes beyond low-code and could be of influence to any developer. The use of SDT throughout the data analysis could be why this motivator emerged in this study and not in other studies. Indeed, recognition and esteem are classic SDT examples of the introjected regulation type of extrinsic motivators. In

comparison, de Souza et al. (2016) slightly hints at the existence of this motivator for MSECOs, as developers in their study mention the large user base and potential to reach many users with their apps as factors to join the SECO.

To conclude, in this study, the most important motivators to join are (M1) fun throughout the development process, (M2) intellectual stimulation, (M3) the ability to create apps rapidly and agile, (M4) the acquisition and improvement of skills, (M5) the ability to contribute to something new, (M12) the ability to exchange knowledge with other developers and (M11) the attraction to the size and quality of the developer community. The choice of SECO is mainly influenced by their direct surroundings, such as managers or personal social networks. Furthermore, the continuance of developers is stimulated by the same motivators but also includes (M13) influence of the community events, (M15) the ability to self-manage and (M17) to create value with one's work from the intrinsic motivators. While most extrinsic motivators increase some, the increase of (M14) the presence of the orchestrator and (M16) the satisfaction with the SECO from the extrinsic motivators are most evidently visible.

7.1.2 Differences in Motivation

Based on the results of the types of actors that were researched, a validation of the classification could not be established. While the classification proved successful, as all actors were present in the case SECO, the actors are more fluid in terms of SECO activities. This confirms the vagueness in the literature regarding the classification of SECO actors and implies that a classification must either be formed per SECO or a new approach should be established. Suggestions to follow-up on this matter are discussed in section 7.3. This made it more challenging to distinguish between the differences in motivation of the different SECO actors' developers.

Nonetheless, differences could be observed between these developer groups. There were no clear distinctions on how motivators were highly present for one category of developers and not for other categories. Contrarily, the motivators were overall equally relevant between the groups. The interpretation of some of a few motivators to continue differed, resulting in a difference in overall motivation. This is true for (M4) the acquisition of skills, (M15) to create value with one's work and (M16) the satisfaction of the developer with the SECO. Overall, the category of service providers interpreted the motivators of M4 and M15 as more extrinsic than the other developer categories. At the same time, the ISV developers stress the importance of M16 since their product is fully dependent on the SECOs. In one of the respondent validation interviews of the customer developers, they stressed that this is also crucial for them since crucial operations for business as usual can depend on the apps.

Given these differences, this implies that the developers in the group of service providers are relatively more extrinsically motivated than the developers in the other categories of developers. Nonetheless, all developers stressed the intrinsic motivators as the most important, making it hard to compare the actual difference in motivation. Additionally, no significant differences could be found between developers from commercial and public organisations. ISV developers showed signs of higher extrinsic motivations as well. However, they also have their role as consultants, making it hard to draw a valid conclusion on ISV as an independent actor group. Hence, they are to be grouped with service providers as niche players.

Furthermore, many developers switch roles within their careers within the SECO. Within this study, four developers had switched between different types of actor organisations, and one is

considering switching. Given that developers switch roles throughout their careers, but motivators such as the intrinsic ones remain the same, it supports that there might be no significant differences in motivation. It is, for example, hard to believe that developers will switch to an actor that provokes more extrinsically motivated behaviour since SDT posits that self-determined behaviour is the natural tendency for people to seek and developers are more intrinsically motivated. Also, to what extent this influences continuation in the SECO is unclear. These developers have in common that they still enjoy working in the SECO, but do not enjoy their specific activities. This leads to their decision to stop their continuance at a specific actor, but to remain in the SECO. This emphasises the importance of intrinsic motivators in the continuance of developers.

The lack of clear differences can also mean that existing differences were not measured. It would mean that the sample and research method did not suit the research objective, motivation was mismeasured during the data collection process, or the motivators were an insufficient choice of measurement. This could be the case as the sample size was limited, and semi-structured interviews are not the best for comparison. A second reason could be that the interview protocol is aimed directly at the developer's role. No questions such as "Did you choose specifically for [ISV/a service provider/customer]? And, if yes, why?". This could have resulted in missing clear differences. However, the developers who switched from organisations were asked why, and this yielded only minor differences.

It is also probable that the motivators are insufficient for measuring the motivation. Many motivators overlap, there are many nuances to the motivators, and some motivators are found to be both intrinsic and extrinsic. If there is so much overlap between the motivators, it is highly possible that the current list of motivators does not properly represent the motivation of developers. This is not surprising given the subjectivity of the motivation construct and the motivators found in this study. For many text extracts, there was no clear single-fit, making coding difficult as to what motivator a statement should be assigned to. Also, the broad interpretation of the motivators made it difficult to assign weights to the importance of motivators. As a result, creating a unitary scale to determine whether and how important a motivator was, and subsequently ranking the importance proved difficult.

Another possibility is that there are rarely any true differences between the developers, or the differences lie elsewhere. Many factors that define the context of the developer have not been taken into account, such as the size of the organisation and the specific role of the developer in the team, since this also reflects how the developer sees their role and values what is important. Many developers already work at an organisation where Mendix is introduced; hence, to what extent do developers influence the choice to join the SECO if it is an executive decision of their organisation? Nonetheless, no controlled motivation was found to be of influence in the developer's decision to join the SECO.

Since developers interpret the motivators broadly and differently, it implies that some other variables might influence their motivation. Many factors that define the developer's context have not been considered, such as the size of the organisation, the career stage in the developer's life and the developer's specific developer role in the team. As suggested by de Souza et al. (2016), the prior experiences of developers influences their decision as well. This has been followed up and resulted in the differences between developers with high-code and no high-code experience.

Throughout the data collection, it became clear that distinct differences exist between the motivators of developers with and without prior experience in high-code software development. These are related to intrinsic motivations. Interestingly, developers join the SECO for the same reason of having (M3) the ability to create apps fast and agile. However, developers without high-code deem this step into low-code as in their reach of skills. In contrast, developers with high-code experience avoid the complexities that hinder the development in high-code. These same types of complexities align with those of Miranda et al. (2014).

In retaining high-code developers, the possibilities for high-code challenges become more present. This implies that while relatively all developers feel the need for competence through intellectual stimulation, it is expressed through different needs from the SECO. The SECO must propose ways to offer high-code possibilities to retain high-code developers. In turn, fostering the knowledge of high-code developers in the low-code can pose an advantage to the SECO.

7.2 GENERALISABILITY

This study adopted the exploratory single case-study approach; thus, there is little room for generalisation (Yin, 2009). Nevertheless, this subject must be carefully considered, given the layering of SECO types covered in this study. The levels of SECO discussed throughout this thesis were the proprietary SECO, the PaaS SECO and the low-code PaaS SECO, in which every latter is a subset of the previously named SECO. This section discusses the extent to which the findings from this case study can be generalised per level of SECOs.

This study reconfirms and expands the knowledge of the overall nature of software developer motivation in proprietary SECOs. It bridges the gap between the MSECO and other proprietary SECOs, finding that intrinsic motivations are more important than extrinsic ones. Due to the limited sample size and adapting the single case study approach in this research, this study can't make any claims for the entire population of PaaS SECO developers or low-code PaaS SECO developers. Furthermore, since low-code is a PaaS niche, it is not possible to confidently generalise these results to PaaS SECOs in general. Nonetheless, since this is the first study to extend the knowledge of SECO developer motivation outside the realm of MSECO, the findings in this study matter since they add evidence to the claims made in MSECO literature (Koch & Kerschbaum, 2014; Steglich et al., 2019).

As this is the first study investigating the motivators of developers in the field of low-code, other studies are needed to validate the findings of this study and expand the knowledge of developer motivations in the context of low-code PaaS. This will enrich and validate the understanding of motivators found in this study. Other low-code SECOs may have other characteristics that will result in different prioritisation of developer motivators. For example, the differences in characteristics between Android and iOS cause a distinct difference between the motivations of these developers, as iOS developers are more extrinsically motivated than Android developers (Koch & Kerschbaum, 2014).

In addition, new motivators emerged as relevant for developers in the low-code PaaS SECO in this study. While rapid and agile development is closely linked to one of the profound characteristics of low-code, I believe this might also reflect the needs of developers in other SECOs. For example, this very motivator is what draws high-code developers from other type of SECOs to low-code. Additional findings are interesting on the level of proprietary SECOs

that could prove relevant in other proprietary SECOs. While the findings concerning the found motivators can't be generalised from this study to other SECOs, it is worth mentioning that such motivators could be present in other proprietary SECOs as well since the status developers gain from work is not something that can only be present in a low-code PaaS SECO.

Regarding the differences between motivators, findings need support from future studies to confirm what was found here, preferably using research methods that are better suited for comparison.

7.3 IMPLICATIONS

Based on the interpretations of these results, the following implications have been established. The theoretical implications are discussed first, followed by the implications for practitioners.

7.3.1 Scientific Contribution and Theoretical Implications

This study offers valuable insights into the motivators of developers within the PaaS SECO, a domain that has remained relatively unexplored until now. In doing so, it contributes to an expanding body of knowledge on the social dimension of proprietary SECOs. The theoretical contributions are elaborated on below.

The results, revolving around the importance of motivators, build on existing evidence of the dominant role of intrinsic motivators over extrinsic ones in proprietary SECOs by exploring the topic of developer motivation in an undiscovered context, that of PaaS and low-code PaaS. Similarly to the MSECO, intrinsic motivators remain crucial over time, while the relevance of extrinsic motivators increases. The findings of this study are relevant and enrich the current understanding of developer motivation as the architecture configuration, in terms of actor types, and underlying technology of PaaS and low-code PaaS significantly differ from the MSECO or FOSS SECO.

Furthermore, by adopting the lens of Self-Determination Theory to compass the construct of motivation, this research underscores the complex and multifaceted nature of motivation, offering a more nuanced understanding than previously emphasised by scholars. While some scholars find similar nuances, this study effectively applies SDT to underscore a revision on the motivational nature of these nuances. As a result, this study challenges the existing categorisation of motivators that scholars have previously used. According to this study, motivational factors blindly assumed to be intrinsic or extrinsic are found to be categorised wrongly or are broader in scope than previously thought. Learning skills were previously considered intrinsic, and creating value was considered extrinsic, whereas, in this study, they exist in both intrinsic and extrinsic forms. This is a crucial finding as it betters the understanding of motivators and grounds them in theory. By revising the motivators along the SDT continuum, this study confirms the prior findings of the dominant role of intrinsic motivators over extrinsic motivators and grounds them in theory.

SDT also reveals that the majority of extrinsic motivators are autonomous forms. This nuance adds substance to the findings of prior scholars since it provides a more detailed distribution of the motivators. This insight matters since intrinsic and autonomous motivation are linked to improved learning and performance, particularly in a professional work environment (Deci et al., 2017). Scholars benefit from this insight as it provides more precision and explanation in future research on causal relationships or motivation enhancement strategies, as this finding enables more precise measurement of motivation-related variables. It also raises the

question of whether the intrinsic-extrinsic binary scale still applies in the case of extrinsic motivation, which is a more complex construct, as suggested by earlier research.

Additionally, these findings offer a foundation upon which subsequent studies can build, utilising motivation theory to grasp motivation's underlying nature and driving forces. Potential causal relationships can be speculated on more theoretically grounded, such as how the motivators overlap and influence each other. In addition, a profound understanding of motivators grounded in a single theory contributes to developing better and more accurate strategies to enhance motivation. From a scholarly perspective, the lens of motivation theory provides more meaning and accuracy to the findings and provides a better foundation to expand knowledge on developer motivation.

Lastly, recognising differences in motivation between various categories of developers can assist in bridging the gap in the perception of developers within a SECO as a homogenous group. This study thus offers a fresh perspective on motivation within SECOs, exploring any differences along the roles of developers and their previous experiences. Based on preliminary findings, it lays the groundwork for further research and understanding of this crucial aspect of developer behaviour.

7.3.2 Practical Implications

In continuation of the previous paragraph, several managerial implications have also been identified. The orchestrator attempts to keep the PaaS SECO healthy by attracting and sustaining developers (Lucassen et al., 2013). For PaaS providers, the following implications have been defined. Specific recommendations for the orchestrator of this case study are discussed in the conclusion.

This first study regarding the motivators and their importance helps PaaS providers understand better what developers deem important and when it becomes important. To start participation, mainly intrinsic motivators are identified that are related to the competence need. Based on this finding, PaaS providers gain insight into their platform's role in a developer's decision to start, as using it should be perceived as fun, challenging and facilitate creativity. It should also mitigate complexities of development to make development fast and agile.

The use of SDT as a lens to perceive motivation can guide practitioners to refine their strategies for attracting and sustaining developers. These findings should be taken into account when considering what strategies should be employed to enhance motivations. For example, no extrinsic rewards should be offered to attract developers that are autonomously motivated as this potentially decreases the developer's initial motivation. In contrast, where motivation is low or controlled, extrinsic rewards could be deployed. Specific types of enhancement strategies should be followed up in future research.

The presence of the orchestrator, both online and in physical events, has a positive impact on the motivation of developers. These results should be considered when considering how to interact with the developer community. Based on this finding, PaaS providers should invest in opportunities to blend with the developer community. Simultaneously, orchestrators can take advantage of their presence in the developer community by gaining insights and information on what improvements the SECO developers want to see. In addition, how ecosystem innovations and new platform features are delivered to the developer community matters. The

study could not establish to what extent this influences the adoption of new features and increased platform activity.

Regarding the differences in motivation, specific features of the SECO do not attract developers from one actor more than another. For orchestrators, this data contribute a clearer understanding of Based on the differences, the key takeaway is that the developers are not motivated by their. Since this could have resulted from the sample size and methods used, orchestrators are suggested to do quantitative research. If no differences exist, it has become evident that the organisational context does matter and influences the developer's decision.

7.4 LIMITATIONS AND FUTURE RESEARCH

This section aims to reflect on some of the limitations of this study. These limitations result in interesting future research avenues.

First, any limitations regarding the generalisability of this research are discussed. For this study, a sample was selected limited to developers in the Netherlands to eliminate any cultural differences that could interfere with the results. This means that the results of this research will be generalisable to the Netherlands, while generalising across cultures is less reliable. Future research could carry out similar studies in different cultures to understand how the importance of motivations differs over culture.

Additionally to the generalisability, the sample in this study was mainly obtained through social events and from a program of top-tier developers. This has potentially resulted in some bias in favour of the case SECO, leading to skewed results regarding some of the motivators. For example, some developers mentioned that they enjoy going to the meet-ups and consider it fun to stay updated with the platform innovations, while most of their colleagues don't. This paves the way for interesting future research to understand the different layers of involvement and engagement of developers and what drives developers who are less involved.

The selected methodology has also resulted in some constraints and limitations in this study. First, within the sample, saturation was not reached. Hence, there could be additional takes on motivation, and new motivators could potentially arise, provided that a larger sample size is considered in the study. Second, the most important limitation for comparing the differences is the small sample size per category and the use of semi-structured interviews for a comparative analysis between developer groups. While the study could confidently nuance the differences of motivators, a large sample and quantitatively deployable research methods, such as questionnaires or structured interviews, would be better suited to highlight any differences with more confidence.

Finally, the results showed that it is very hard to separate the developers of SECO actors based on their roles and the actor's activity. This interfered with the results and might have potentially skewed the results. While it is proposed to investigate further the differences in motivation between actor categories, a new categorisation must be proposed to separate the roles of the developers and get a clearer view of the differences. It is advised to ask a developer what role he identifies the most with and for whom he develops (end-users internal to his organisation or external). Having a clear understanding of how the developers identify and who they develop for may yield better results.

To properly quantify and rank the importance of the motivators of developers, the next step should be to conduct a survey. To ensure that survey questions measure the motivators by the appropriate motivation type, SDT should be used in their formulation. To separate the developers, it is necessary to not only categorise them based on actors but also on what role they identify most with.

8 CONCLUSION

After thoroughly discussing the results and acknowledging the limitations of this study, this thesis concludes by answering the main research question. Thereafter, concise and practical recommendations specific to the orchestrater of the case study are provided.

8.1 ANSWERING THE MAIN RESEARCH QUESTION

To conclude, this thesis sheds light on the motivators of developers to start and continue participation in a PaaS SECO and uncovered the differences in motivation between different developer groups.

The central research question for this thesis was as follows:

"What motivates individual developers in diverse actor types to start and continue participation in a PaaS SECO?"

In an exploratory case study, through semi-structured interviews, this study analysed the developers of ISVs, service providers and customers from the public and private sectors in the PaaS SECO context and found that developers from all actors show strong intrinsic motivation to start in the SECO. Developers seek intellectual challenge, a creative outlet and to what extent the SECO can fulfil their need for competence while lowering complexities and bottlenecks encountered in software development, which, in turn, leads to fun. The intrinsic motivators reflect this to start participation in the SECO, which are fun throughout the development process, intellectual stimulation, the ability to create apps rapidly and agile, the acquisition and improvement of skills, and the ability to contribute to something new. Extrinsic motivators are less important to start participation. The identified extrinsic motivators are the attraction to the size and quality of the developer community and the ability to exchange knowledge and link strongly to the support that the SECO and its developer community provide during the start, as these mitigate potential complexities that may deprive the fulfilment of competency needs and decreased motivation.

In the continuation of developers in the SECO, developers have gained a deeper understanding of all that plays in the SECO. The fun increases as their skills progressed and developers gain more confidence. Creating value with their work becomes crucial and can both extrinsically or intrinsically motivate them. The fulfilment of autonomy needs by agile methodologies increases their intrinsic motivations. Furthermore, developers become more aware and sensitive to the platform's strategic direction, because they want to be part of a successful SECO. The relationship between the developer, the community and the orchestrator starts to affect a developer's motivation for the continuance as they participate in community events and develop a perception of the orchestrator's role and attitude in the SECO, leading to increased motivation for continuance or separation from the SECO. Intrinsic motivators remain important and include the ability to self-manage and to create value with one's work and the influence of community events. The significance of extrinsic motivators, of which the majority is an autonomous form of motivation, is more important in continuance. These include the presence of the orchestrator and the satisfaction with the SECO, together with the extrinsic motivators important to start participation.

Differences in the motivations of the developers of the diverse actors start to become more evident in continued participation, mostly evident in what drives developers to learn skills and

for whom they create value. While intrinsic motivators remain crucial for all, niche players are more extrinsically motivated in due to their work for external clients than customers, who develop apps for internal use.

Due to the qualitative methods deployed, this study emphasised the nuances within motivators and, in turn, paves the way for future research to focus on quantifying the importance of motivators. Additionally, the exploratory approach and inclusion of mainly active and community-involved developers located in one country leaves room for new research to validate the findings of this study by including other PaaS SECOs and less community-active developers to strengthen the grounds for generalisability.

8.2 RECOMMENDATIONS FOR MENDIX

Apart from the general implications for practitioners in section 7.3.2, recommendations and potential avenues for future research are proposed specifically towards Mendix. This is done separately as this section aims to provide recommendations that align with the specific goals and strategies currently implemented by Mendix. The 4 C's guide the learning and programs team: grow the developer community's *Capacity*, increase developers' *Capabilities*, increase the *Connections* within the community and increase the number of *Contributions* to the community. In light of this strategy, the following recommendations are proposed:

1. To grow the **capabilities** of developers, especially developers without a background in high-code mentioned struggling with the transition from theory (e.g. learning paths) to practice (e.g. software practices), this effect can be mitigated by introducing realistic software project cases in the formal and online training material. Additionally, this addresses the lack of awareness regarding the motivators regarding self-management and creating value.

2. Furthermore, to increase **contributions** and **capabilities**, an increase in developers goes hand-in-hand with the perception of decreasing overall quality of the developer community by the more experienced developers. I would suggest creating dedicated spaces for onboarding developers to post questions so that these developers can help each other and experienced developers can enter these spaces at their own will. This would improve autonomy and yield higher response rates to questions from onboarders. In addition, expert-certified developers could be given a curator role within these forums to separate the good from the bad answers. This could potentially soothe their feeling of being over-qualified.

3. In continuation of improving the forum for increased **contributions**, besides the peer-to-peer help, merging internal developers from Mendix more with the developer community can have two benefits: first is that developers learn what the pain points are of developers. Second, the presence and curation of internal Mendix employees and developers on the platform can boost motivation to contribute more.

4. This is also relevant for the physical community spaces such as developer meet-ups and events. Increasing the presence of Mendix employees increases motivation and is possibly linked to higher identification. Currently, communities are self-emerging in other countries such as Lisbon and Brazil. For the emergence of these regional communities, I would suggest promoting physical events and having representatives attend these meet-ups and learn from these cultures and be aware of how motivation might differ. This can potentially increase **connections**.

For **capacity** growth, developers in this study did not compare Mendix with competitors before joining. While their choice for low-code is influenced by intrinsic motivation, their specific choice for Mendix was mostly based on what PaaS was introduced to them first. While this research included developers with a wide range of SECO joining dates, the market, competition between low-code PaaS providers, and increasing familiarity with low-code might have changed by now. An important future research avenue is to research if and how developers make a thorough choice between Mendix and other low-code platforms and why they opt for Mendix.

In addition, this research provided reasons why developers opt for low-code. There can't be any claims made on why developers do not opt for low-code. Some speculations can be made regarding high-code developers, as one developer said feeling over-qualified and lacking insight into code being generated in the platform's back-end. For non-high-code developers, no such claims can be made. Future research should look into these differences. A fruitful starting point for such research would be to conduct research on starting developers in university programs and investigate why developers would and would not consider low-code app development as a starting point for their career. Additionally, research on the decision-making process of enterprises to adopt Mendix and how they include (potential) developers in the decision-making process can provide insight on the selling points.

Lastly, this research mainly included developers that are active in the community, thereby excluding the developers that could be more interesting. Getting in contact with non-active developers has proven to be difficult in past Mendix research. However, most developers included in this study have colleagues that are not or less active in the developer community. As a final suggestion to source these developers, I can recommend Mendix to approach active developers through the meet-ups and through them source their non-active colleagues. This can provide crucial insights into the non-active developers and can allow interesting comparative analysis between active and non-active developers within the same organisation.

REFERENCES

Arango, P. (2018). Motivation: Introduction to the Theory, Concepts, and Research. In P. Orellana García & P. Baldwin Lind (Eds.), *Reading Achievement and Motivation in Boys and Girls* (Vol. 15, pp. 1–28). Springer International Publishing. https://doi.org/10.1007/978-3-319-75948-7_1

Atkinson, J. W. (1964). *An introduction to motivation*. Van Nostrand.

Bandura, A., Freeman, W. H., & Lightsey, R. (1999). Self-Efficacy: The Exercise of Control. *Journal of Cognitive Psychotherapy*, *13*(2), 158–166. https://doi.org/10.1891/0889-8391.13.2.158

Barbosa, O., Dos Santos, R. P., Alves, C., Werner, C., & Jansen, S. (2013). A Systematic Mapping Study on Software Ecosystems through a Three-dimensional Perspective. In S. Jansen, S. Brinkkemper, & M. A. Cusumano (Eds.), *Software Ecosystems: Analyzing and Managing Business Networks in Software Industry* (pp. 59–84). Edward Elgar.

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, *3*(2), 77–101. https://doi.org/10.1191/1478088706qp063oa

Cain, M., & Severson, L. (2023). *Hype Cycle for Digital Workplace Applications, 2023* (ID G00791003). Gartner. https://www.gartner.com/interactive/hc/4592699?ref=solrctp&refval=377130464&ipDocCd=793411

Ceccagnoli, Forman, Huang, & Wu. (2012). Cocreation of Value in a Platform Ecosystem! The Case of Enterprise Software. *MIS Quarterly*, *36*(1), 263. https://doi.org/10.2307/41410417

Constantinou, E., & Mens, T. (2017). An empirical comparison of developer retention in the RubyGems and npm software ecosystems. *INNOVATIONS IN SYSTEMS AND SOFTWARE ENGINEERING*, *13*(2–3), 101–115. https://doi.org/10.1007/s11334-017-0303-4

Corbin, J., & Strauss, A. (2008). *Basics of Qualitative Research (3rd ed.): Techniques and Procedures for Developing Grounded Theory*. SAGE Publications, Inc. https://doi.org/10.4135/9781452230153

Crowe, S., Cresswell, K., Robertson, A., Huby, G., Avery, A., & Sheikh, A. (2011). The case study approach. *BMC Medical Research Methodology*, *11*, 100. https://doi.org/10.1186/1471-2288-11-100

Damasio, A. R. (1996). The somatic marker hypothesis and the possible functions of the prefrontal cortex. *Philosophical Transactions of the Royal Society of London. Series B, Biological Sciences*, *351*(1346), 1413–1420. https://doi.org/10.1098/rstb.1996.0125

de Souza, C., Figueira, F., Miranda, M., Ferreira, R., Treude, C., Singer, L., & ACM. (2016). The Social Side of Software Platform Ecosystems. *Instituto Tecnologico Vale Desenvolvimento Sustentavel*, 3204–3214. https://doi.org/10.1145/2858036.2858431

Deci, E. L., Olafsen, A. H., & Ryan, R. M. (2017). Self-Determination Theory in Work Organizations: The State of a Science. *Annual Review of Organizational Psychology and Organizational Behavior*, *4*(1), 19–43. https://doi.org/10.1146/annurev-orgpsych-032516-113108

Deci, E. L., & Ryan, R. M. (2000). The 'What' and 'Why' of Goal Pursuits: Human Needs and the Self-Determination of Behavior. *Psychological Inquiry*, *11*(4), 227–268. https://doi.org/10.1207/S15327965PLI1104_01

Deci, E. L., & Ryan, R. M. (Eds.). (2004). *Handbook of self-determination research* (Softcover edition). University of Rochester Press.

Eccles, J. S., & Wigfield, A. (2002). Motivational Beliefs, Values, and Goals. *Annual Review of Psychology*, *53*(1), 109–132. https://doi.org/10.1146/annurev.psych.53.100901.135153

Ernst, M., Pine, D. S., & Hardin, M. (2006). Triadic model of the neurobiology of motivated behavior in adolescence. *Psychological Medicine*, *36*(3), 299–312. https://doi.org/10.1017/S0033291705005891

Fontao, A. D. L., Dos Santos, R. P., & Dias-Neto, A. C. (2015). Mobile Software Ecosystem (MSECO): A Systematic Mapping Study. *2015 IEEE 39th Annual Computer Software and Applications Conference*, 653–658. https://doi.org/10.1109/COMPSAC.2015.121

Fontao, A., Dias-Neto, A., Viana, D., & IEEE. (2017). Investigating factors that influence developers' experience in mobile software ecosystems. *Universidade Federal de Amazonas*, 55–58. https://doi.org/10.1109/JSOS.2017.10

Gartner. (2022). *Top Strategic Technology Trends for 2023: Industry Cloud Platforms*. Gartner.

https://www.gartner.com/document/4020027?ref=solrAll&refval=369626369

George, T. (2022, November 30). *Semi-Structured Interview | Definition, Guide & Examples*. Scribbr. https://www.scribbr.com/methodology/semi-structured-interview/

Ghosh, R. A. (2007). Understanding Free Software Developers: Findings from the FLOSS Study. In J. Feller, B. Fitzgerald, S. A. Hissam, & K. R. huff (Eds.), *Perspectives on Free and Open Source Software* (pp. 23–45). MIT Press. https://ieeexplore.ieee.org/document/6277079

Goldbach, T., & Benlian, A. (2015). *Understanding informal control modes on software platforms -The mediating role of third-party developers' intrinsic motivation*. 2015 International Conference on Information Systems: Exploring the Information Frontier, ICIS 2015. Scopus. https://www.scopus.com/inward/record.uri?eid=2-s2.0-85126608000&partnerID=40&md5=79f7830c16b303950d0d04053a8b30df

Goldbach, T., & Kemper, V. (2014). Should I stay or should I go? The effects of control mechanisms on app developers´ intention to stick with a platform. *European Conference on Information Systems*.

Handoyo, E., Jansen, S., & Brinkkemper, S. (2013). Software Ecosystem Roles Classification. In G. Herzwurm & T. Margaria (Eds.), *Software Business. From Physical Products to Software Services and Solutions* (Vol. 150, pp. 212–216). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-39336-5_21

Hartigh, E. den, Hartigh, E. den, Tol, M., & Visscher, W. (2006). *The Health Measurement of a Business Ecosystem*. 1–7.

Hertel, G., Niedner, S., & Herrmann, S. (2003). Motivation of software developers in Open Source projects: An Internet-based survey of contributors to the Linux kernel. *Research Policy*, *32*(7), 1159–1177. https://doi.org/10.1016/S0048-7333(03)00047-7

Hu, N., Tian, G., Liu, L., Liang, B., Gao, Y., Dong, M., Jenab, K., Shafiei-Monfared, S., Luis Daniel Otero, Otero, L., Centeno, G., Centeno, G., Otero, C. E., DEA-Tobit, A., Reeves, K., Levesque, M., Lee, I. H., Lee, I. H., Minniti, M., … Heinzl, A. (2011). Why Do Complementors Participate? An Analysis of Partnership Networks in the Enterprise Software Industry Y. Kwak, J. Park, B. Chung, and S. Ghosh Understanding End Users' Acceptance of Enterprise Resource Planning (ERP) System in Project-Based Sectors. *IEEE Transactions on Engineering Management*. https://doi.org/10.1109/tem.2011.2111421

Hyrynsalmi, S., Hyrynsalmi, S., Seppänen, M., Nokkala, T., Suominen, A., & Järvi, A. (2015). *Wealthy, Healthy and/or Happy—What does 'Ecosystem Health' Stand for?* 272–287. https://doi.org/10.1007/978-3-319-19593-3_24

Iansiti, M., & Levien, R. (2004). Strategy as ecology. *Harvard Business Review*, *82*(3), 68–126.

Jang, H., Reeve, J., & Deci, E. L. (2010). Engaging students in learning activities: It is not autonomy support or structure but autonomy support and structure. *Journal of Educational Psychology*, *102*(3), 588–600. https://doi.org/10.1037/a0019682

Jansen, S., Brinkkemper, S., & Finkelstein, A. (2009). Business Network Management as a Survival Strategy: A Tale of Two Software Ecosystems. *IWSECO@ICSR*.

Jansen, S., Brinkkemper, S., & Finkelstein, A. (2008). *Component Assembly Mechanisms and Relationship Intimacy in a Software Supply Network YR 2008*. EurOMA 2008, tradition and innovation in operations management: connecting past and future, Groningen, Netherlands.

Jansen, S., Brinkkemper, S., Souer, J., & Luinenburg, L. (2012). Shades of gray: Opening up a software producing organization with the open software enterprise model. *Journal of Systems and Software*, *85*(7), 1495–1510. https://doi.org/10.1016/j.jss.2011.12.007

Jansen, S., Finkelstein, A., & Brinkkemper, S. (2009). A sense of community: A research agenda for software ecosystems. *International Conference on Software Engineering*, 187–190. https://doi.org/10.1109/icse-companion.2009.5070978

Kleinginna, P. R., & Kleinginna, A. M. (1981). A categorized list of motivation definitions, with a suggestion for a consensual definition. *Motivation and Emotion*, *5*(3), 263–291. https://doi.org/10.1007/BF00993889

Knodel, J., & Manikas, K. (2015). Towards a Typification of Software Ecosystems. In J. M. Fernandes, R. J. Machado, & K. Wnuk (Eds.), *Software Business* (Vol. 210, pp. 60–65). Springer International Publishing. https://doi.org/10.1007/978-3-319-19593-3_5

Koch, S., & Guceri-Ucar, G. (2017). Motivations of application developers: Innovation, business model choice, release policy, and success. *Journal of Organizational Computing and Electronic Commerce*, *27*(3), 218–238. https://doi.org/10.1080/10919392.2017.1331673

Koch, S., & Kerschbaum, M. (2014). Joining a smartphone ecosystem: Application developers' motivations and decision criteria. *Information & Software Technology*, *56*(11), 1423–1435. https://doi.org/10.1016/j.infsof.2014.03.010

Lakhani, K., & Wolf, R. G. (2007). Why Hackers Do What They Do: Understanding Motivation and Effort in Free/Open Source Software Projects. In J. Feller, B. Fitzgerald, S. A. Hissam, & K. R. huff (Eds.), *Perspectives on Free and Open Source Software* (pp. 3–21). MIT Press. https://ieeexplore.ieee.org/document/6277090

Lucassen, G., van Rooij, K., & Jansen, S. (2013). Ecosystem Health of Cloud PaaS Providers. *International Conference on Software Business*, 183–194. https://doi.org/10.1007/978-3-642-39336-5_18

Manikas, K., & Hansen, K. M. (2013). Software ecosystems—A systematic literature review. *Journal of Systems and Software*, *86*(5), 1294–1306. https://doi.org/10.1016/j.jss.2012.12.026

Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., & Ghalsasi, A. (2011). Cloud computing—The business perspective. *Decision Support Systems*, *51*(1), 176–189. https://doi.org/10.1016/j.dss.2010.12.006

Maslow, A. H. (1943). A theory of human motivation. *Psychological Review*, *50*(4), 370–396. https://doi.org/10.1037/h0054346

Mell, P. M., & Grance, T. (2011). *The NIST definition of cloud computing* (NIST SP 800-145; 0 ed., p. NIST SP 800-145). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-145

Memeti, A., Huck-Fries, V., Wiesche, M., Thatcher, J. B., & Krcmar, H. (2021). *Motivation in IT Projects: Investigating the Effect of Agile Practices on Team Members' Intrinsic Motivation*. 161.

Mendix. (n.d.). *Low-code Application Development Platform*. Mendix. https://www.mendix.com/

Miranda, M., Ferreira, R., de Souza, C., Figueira, F., & Singer, L. (2014). *An Exploratory Study of the Adoption of Mobile Development Platforms by Software Engineers* (T. Catarci, A. Abadi, & A. Wasserman, Eds.; WOS:000380927500012; pp. 50–53). https://doi.org/10.1145/2593902.2593915

Moore, J. F. (1999). Predators and prey: A new ecology of competition. *Harvard Business Review*, 121–141.

Ng, J. Y. Y., Ntoumanis, N., Thøgersen-Ntoumani, C., Deci, E. L., Ryan, R. M., Duda, J. L., & Williams, G. C. (2012). Self-Determination Theory Applied to Health Contexts: A Meta-Analysis. *Perspectives on Psychological Science*, *7*(4), 325–340. https://doi.org/10.1177/1745691612447309

Petri, H. L., & Govern, J. M. (2004). *Motivation: Theory, research, and applications* (5th ed). Wadsworth/Thomson.

Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical Software Engineering*, *14*(2), 131–164. https://doi.org/10.1007/s10664-008-9102-8

Ryan, R. M., & Deci, E. L. (2000a). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, *55*(1), 68–78. https://doi.org/10.1037/0003-066X.55.1.68

Ryan, R. M., & Deci, E. L. (2000b). When rewards compete with nature. In *Intrinsic and Extrinsic Motivation* (pp. 13–54). Elsevier. https://doi.org/10.1016/B978-012619070-0/50024-6

Ryan, R. M., & Deci, E. L. (2018). *Self-determination theory: Basic psychological needs in motivation, development, and wellness* (Paperback edition). The Guilford Press.

Sekaran, U., & Bougie, R. (2016). *Research methods for business: A skill building approach*. John Wiley & Sons.

Steglich, C., Marczak, S., de Souza, C., Guerra, L., Mosmann, L., Figueira, F., Perin, M., Assoc Comp Machinery, & IEEE Comp Soc. (2019). Social Aspects and How They Influence MSECO Developers. *Pontificia Universidade Catolica Do Rio Grande Do Sul*, 99–106. https://doi.org/10.1109/CHASE.2019.00032

Tiwana, A. (2015). Platform Desertion by App Developers. *Journal of Management Information Systems*, *32*(4), 40–77. https://doi.org/10.1080/07421222.2015.1138365

Valença, G., Alves, C., & Jansen, S. (2018). Strategies for managing power relationships in software ecosystems. *Journal of Systems and Software*, *144*, 478–500. https://doi.org/10.1016/j.jss.2018.07.036

van Angeren, J., Alves, C., & Jansen, S. (2016). Can we ask you to collaborate? Analyzing app developer relationships in commercial platform ecosystems. *Journal of Systems and Software*, *113*(113), 430–445. https://doi.org/10.1016/j.jss.2015.11.025

Van Den Broeck, A., Ferris, D. L., Chang, C.-H., & Rosen, C. C. (2016). A Review of Self-Determination Theory's Basic Psychological Needs at Work. *Journal of Management*, *42*(5), 1195–1229. https://doi.org/10.1177/0149206316632058

Vincent, P. (2022). *Follow the Buying Journey for Enterprise Low-Code Application Platforms* (ID G00765841). Gartner. https://www.gartner.com/document/4016176?ref=solrAll&refval=377130631&

Vincent, P., Lijima, K., Leow, A., West, M., & Matvitskyy, O. (2022). *Magic Quadrant for Enterprise Low-Code Application Platforms* (ID G00759450). Gartner. https://www.gartner.com/document/4022825?toggle=1

Weiner, B. (1985). An attributional theory of achievement motivation and emotion. *Psychological Review*, *92*(4), 548–573. https://doi.org/10.1037/0033-295X.92.4.548

Wigfield, A., & Eccles, J. S. (1992). The development of achievement task values: A theoretical analysis. *Developmental Review*, *12*(3), 265–310. https://doi.org/10.1016/0273-2297(92)90011-P

Yin, R. K. (2009). *Case study research: Design and methods* (4th ed). Sage Publications.

Appendix A

Interview Protocol

Background and warm-up questions

1. What was your role before you became a Mendix developer?

2. How or through whom were you introduced to Mendix?

3. What kind of projects do you do?

4. How do you see your role as a [ISV / Consultant / Client] developer?

In-depth questions

5. What were the most important drivers for you to become a Mendix developer?

6. Did you get personal satisfaction from Mendix development, and how important was this in becoming a developer?
   a. Followed by questions regarding _M1 to _M4

7. How did your career prospects play a role into your decision to join Mendix?
   a. Followed by questions regarding _M5 and _M6

8. Were you familiar with the Mendix developer community, and their activities, and did this play a role when you became a developer?
   a. Followed by questions regarding _M7 to _M12

9. How does your direct social circle influence your drive to become a Mendix developer?

10. How did the end-users you created products for play into your motivation when you became a developer?

11. Taking all into account, do you feel satisfied with what the ecosystem offers and to what extent is that important for your motivation?

12. Is there still a driver or some form of motivation that has influenced you which has not come up during our conversation?

Checklist to guide the interview
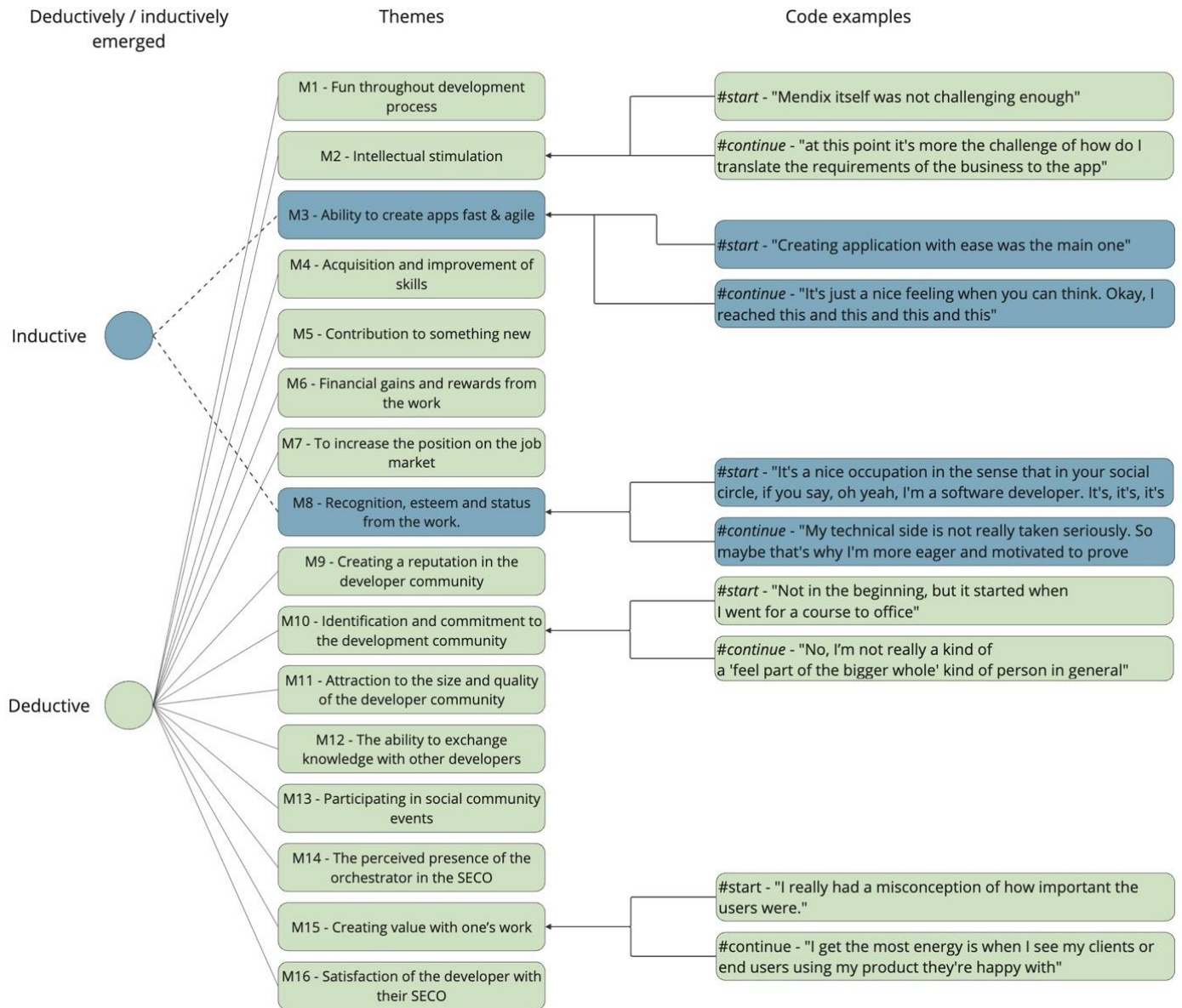
Item #	Motivational factors	Check box
_M1	Fun experienced throughout the development process.	
_M2	The work being intellectually stimulating.	
_M3	The acquisition & improvement of skills.	

_M4	Contributing to something new.	
_M5	Financial gains and rewards from the work.	
_M6	Increasing my position on the job market.	
_M7	Creating/Gaining a reputation in the developer community.	
_M8	Identification and commitment to the development community and becoming part of a community.	
_M9	Feeling attracted by the size and quality of the (online) developer community.	
_M10	The ability to exchange knowledge with other developers.	
_M11	Participating in social community events.	
_M12	The orchestrator's presence in the SECO.	
_M13	Stimulation through the influences of one's social circle (both local and online).	
_M14	Creating value for the end-user(s).	
_M15	Satisfaction of the developer with their SECO.	
_M16	Gaining the ability to self-manage work.	

Appendix B

Coding Trees and Examples

Figure B1: Coding tree 1.

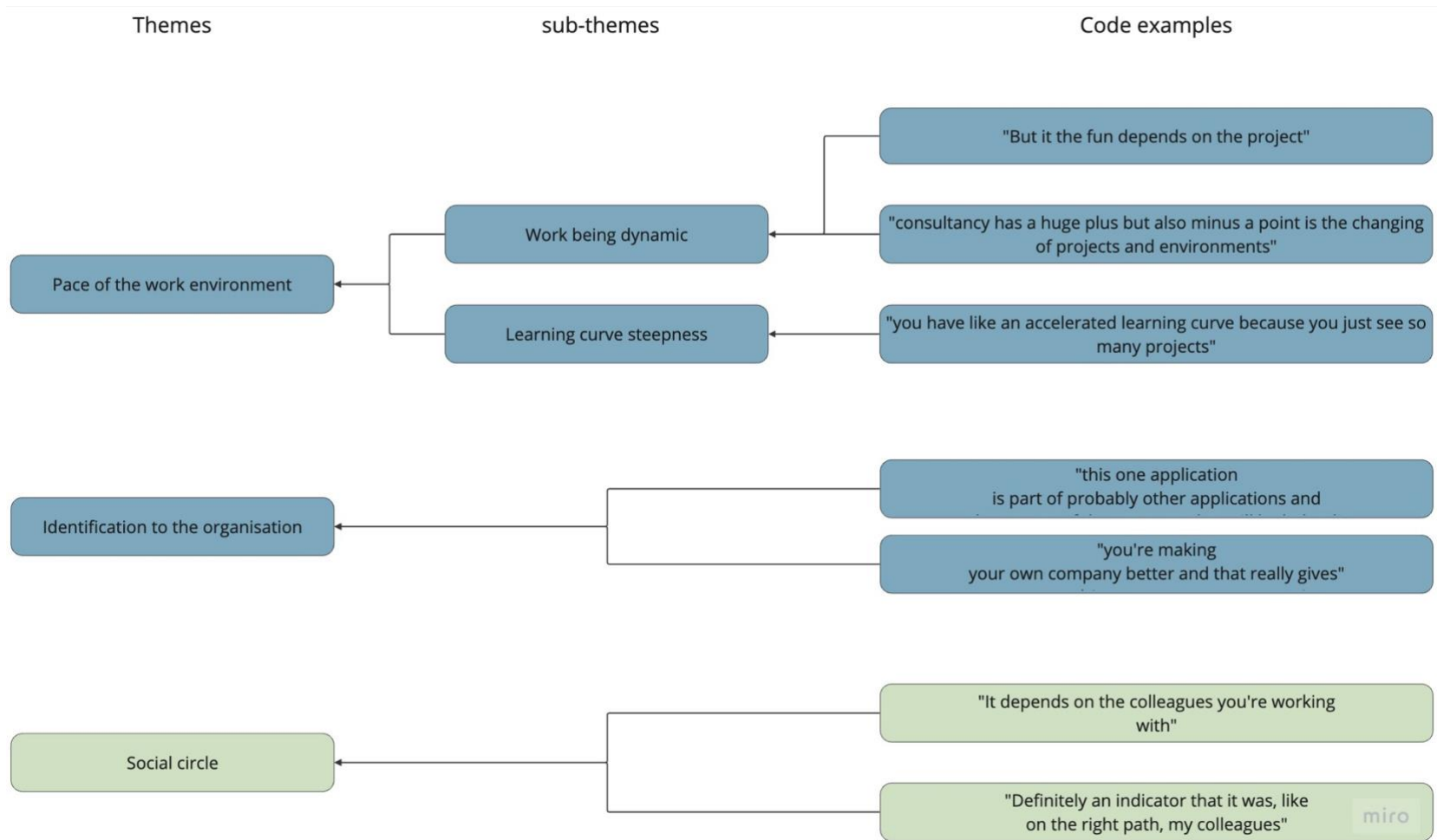| Themes	sub-themes	Code examples
Pace of the work environment	Work being dynamic	"But it the fun depends on the project"
		"consultancy has a huge plus but also minus a point is the changing of projects and environments"
	Learning curve steepness	"you have like an accelerated learning curve because you just see so many projects"
Identification to the organisation		"this one application is part of probably other applications and
		"you're making your own company better and that really gives"
Social circle		"It depends on the colleagues you're working with"
		"Definitely an indicator that it was, like on the right path, my colleagues"

Figure B2: Coding tree 2.

Appendix C

HREC Letter of Approval

Date 15-Sep-2023
Contact person Grace van Arkel, Policy Advisor
Academic Integrity
E-mail E.G.vanArkel@tudelft.nl

TUDelft Delft University of Technology

Human Research Ethics
Committee TU Delft
(http://hrec.tudelft.nl)

Visiting address
Jaffalaan 5 (building 31)
2628 BX Delft

Postal address
P.O. Box 5015 2600 GA Delft
The Netherlands

Ethics Approval Application: the differences in developer motivations to join a PaaS SECO
Applicant: Verver, Caspar

Dear Caspar Verver,

It is a pleasure to inform you that your application mentioned above has been approved.

Thanks very much for your submission to the HREC which has been conditionally approved. Please note that this approval is subject to your ensuring that the following condition/s is/are fulfilled:

Please assure that in case the need arises, you do not share personal data or more information than strictly required to obtain the help of the company.

In addition to any specific conditions or notes, the HREC provides the following standard advice to all applicants:
• In light of recent tax changes, we advise that you confirm any proposed remuneration of research subjects with your faculty contract manager before going ahead.
• Please make sure when you carry out your research that you confirm contemporary covid protocols with your faculty HSE advisor, and that ongoing covid risks and precautions are flagged in the informed consent - with particular attention to this where there are physically vulnerable (eg: elderly or with underlying conditions) participants involved.
• Our default advice is not to publish transcripts or transcript summaries, but to retain these privately for specific purposes/checking; and if they are to be made public then only if fully anonymised and the transcript/summary itself approved by participants for specific purpose.
• Where there are collaborating (including funding) partners, appropriate formal agreements including clarity on responsibilities, including data ownership, responsibilities and access, should be in place and that relevant aspects of such agreements (such as access to raw or other data) are clear in the Informed Consent.

Sincerely,

Dr. Ir. U. Pesch
Chair HREC
Faculty of Technology, Policy and Management