Tilburg University

IT Rebels

Driving citizen development through the enterprise-wide deployment of low-code development platforms.

A case study at IT Rebels

Harold de With

Driving citizen development through the enterprise-wide deployment of low-code development platforms.

A case study at IT Rebels

Thesis 2022 / 2023

H.C. de With
824041 | 2065618
h.c.dewith@tilburguniversity.edu

Submitted in partial fulfillment
of the requirements for the degree
- Master of Science -
in the Information Management domain

Supervised by
Assistant Professor C. Ma

Department of Information Management
Tilburg School of Economics and Management
Tilburg University
Tilburg, The Netherlands

January 2023

Abstract

Low-code development platforms (LCDP) are becoming increasingly popular among enterprises as significant cost savings can be realized. The popularity of LCDPs is increasing so quickly that by 2025 it is expected that 70% of all newly developed business applications are developed through low-code. IT consultancy firm IT Rebels acknowledges this trend and they want to turn the enterprise-wide deployment of LCDPs into a service offering to their clients.

This thesis aims to identify which IT governance-related arrangements have to be considered in an enterprise-wide LCDP deployment To do so, an extensive literature review was conducted. The literature review encompassed theories on LCDPs, enterprise-wide information systems, citizen development, and IT governance. The literature lacked a descriptive model to use as a foundation for the enterprise-wide deployment of an LCDP. The outcome of this study is based on qualitative research.

Data on the topic was collected through interviews with field professionals. The interviews aimed to gain a deep understanding of the context of an enterprise-wide LCDP deployment, and they zoomed in on a set of IT governance challenges. The outcomes of the interviews were analyzed to develop a theory following the grounded theory method

The result of this thesis is the identification of the key IT governance arrangements in relation to the enterprise-wide deployment of an LCDP. These arrangements are encompassed in the finalized research model. The most important finding of this thesis is the identification of three IT governance challenges: guidelines and principles, software ownership, and monitoring and evaluation. These IT governance challenges lead to the following critical design areas: digital development environments, user-assigned roles, and platform security. These critical design areas propose specific LCDP components that need configuration. Finally, it was concluded that citizen development on an LCDP is strived for, but the defined arrangements have to be in place to facilitate it.

Keywords: Low-code development platform, LCDP, Citizen development, IT governance, enterprise-wide,

Preface

I would like to express my deepest appreciation to my supervisor, assistant professor C. Ma, for her invaluable guidance, support, and encouragement throughout the course of this research. Her expertise, patience, and understanding have been instrumental in the successful completion of this thesis.

I would also like to extend my gratitude to IT Rebels for providing me with the opportunity to conduct this research as part of my master's degree in Information Management. Their support and encouragement have been invaluable in helping me to understand the importance of this field and the impact that it can have on education.

Finally, I would like to thank my family and friends for their unwavering support throughout this journey. Their encouragement and understanding have been a constant source of strength and motivation.

Contents

List of figures and tables

List of abbreviations

Abbreviation	Definition
BI	Business intelligence
DLP	Data-loss prevention
DTAP	Develop, testing, acceptance, production
IT	Information technology
LCDP	Low-code development platform
LCNC	Low-code/no-code
SAAS	Software-as-a-service

Chapter 1. Introduction

This chapter gives an introduction to the topic of this thesis. First, a general description of the topic is presented, followed by the problem statement. Subsequently, the research question is defined and the research approach is introduced. Finally, the relevance of this thesis is presented.

1.1 Research topic

Traditionally professional development was the norm for creating business applications but nowadays low-code software development is becoming increasingly popular. Wong et al. (2021) predict an increase in low-code application development; from less than 25% in 2020 to 70% in 2025 (given percentages are % of newly developed business applications). As a result of this increase, citizen development will become more popular given the accessible nature of low-code development platforms (LCDP) (Gartner, 2022). This new type of development proposes great benefits to enterprises, but there are governance-related impediments to beware of. Arrangements have to be made to determine who can engage in the development, which standards and guidelines to provide, and how to insure further development.

This thesis aims to determine the critical factors regarding governance in an enterprise-wide LCDP deployment and how to properly arrange governance to create a safe and robust development environment where citizen developers can tinker with applications. Microsoft Power Platform will be the main focus of this thesis concerning an LCDP, however, results generalize to other providers which are shown in Table 1.

Table 1

Leaders in Enterprise Low-Code Development Platforms

Provider	Pros	Cons
Microsoft	Integration with Office 365 and Azure, Innovation, Market understanding, and responsiveness	Pricing complexity, Marketing, Business logic and workflow
Mendix	Innovation, Product, Growth, and viability	Premium pricing, Marketing, Demographic
Outsystems	Product, Innovation, User Experience	Business logic, Industry strategy, Pricing
Salesforce	Industry strategy, Market responsive, Platform ecosystems	Innovation, Business logic, and workflow, pricing flexibility
ServiceNow	Innovation, Viability, Market responsiveness	Industry strategy, Business model, Frequent price changes

Note: Adapted from "Magic Quadrant for Enterprise Low-Code Application Platforms", by Wong et al., 2021 (https://www.gartner.com/doc/reprints?id=1-27IIPKYV&ct=210923&st=sb)

1.2 Context and problem statement

This chapter provides the problem indication, leading to the formulation of this thesis. Furthermore, the problem statement is defined and the research scope is presented. Finally, also a company description is given of IT-Rebels.

1.2.1 Problem indication

IT Rebels delivers low-code/no-code (LCNC) software solutions to their clients. This is executed through Microsoft Power Platform; a so-called LCDP. An LCDP in short is a platform on which applications can be developed through low-code techniques such as visual coding and model-driven application design. This results in scoped-down software artifacts that automate administrative processes within an enterprise (Wang et al., 2022). In addition to the delivery of these artifacts, IT Rebels also maintains and manages them and new features can be added on demand.

The problem comes from a governance viewpoint. An implemented LCDP without proper governance arrangements could lead to an environment where citizen development is the norm and every employee could possibly engage on the Microsoft Power Platform, leading to questionable quality, functionality, performance, and security (Gartner, 2022). This problem arises since IT Rebels now starts to receive client inquiries for enterprise-wide adoption of Microsoft Power Platform. As opposed to the scoped-down software artifacts. The aim is to have several databases and applications interconnected on a centralized platform i.e. Microsoft Power Platform. This should grant a user-friendly environment, which can be easily scaled and adapted to new situations while always providing up-to-date analytics, visualizations, applications, and integrations (Ahmad et al., 2022).

1.2.2 Problem statement

To properly deploy Microsoft Power Platform and stimulate the adoption of the software throughout an entire enterprise, IT governance is vital. IT governance offers guidance in aligning IT with the enterprise's strategy (Weill & Ross, 2004). IT Governance keeps increasing in importance as dedicated IT ecosystems (such as Microsoft Power Platform) are becoming increasingly popular and vital for businesses (Jansen, 2020). Furthermore, the guidance provided by IT governance can indicate the value derived from an IT investment in

the form of organizational performance (Wu et al., 2015). Regarding the enterprise-wide deployment of LCDPs and IT governance, the following sub-issues are formulated:

Lack of formalized guidelines and principles

There are not yet any formalized guidelines and principles regarding the establishment and granting of measures to artifacts (Moeller, 2013). This refers to the level of importance of a process and how to treat an artifact based on this level. For example, Moran (2020) defines three main levels of increasing process importance, respectively: productivity, important, and critical. Without defined guidelines and principles, the allocation of those levels will likely be too ambiguous.

Software ownership

This sub-issue relates to the way of arranging ownership and accountability over the delivered software artifact(s). The main consideration lies in determining whether ownership should be arranged on people- or service account level. The client has to decide if he delegates ownership to his employees or if a service account is responsible. However, in some instances, service accounts are legally not allowed since they do not direct to a specific person. Misalignment on this part could lead to IT anarchy (Ross, 2003).

Platform monitoring and evaluation

Once an LCDP is fully implemented and adopted, it has to be monitored and evaluated to solve possible errors in the governance arrangements and to stay compliant. Through monitoring data is gathered on relevant subjects, which can be analyzed. Based on these analyses, IT governance arrangements can be adjusted. Aspects that can be monitored rely on the extent of adoption of the system. Following Thong's (1999) research, the extent of adoption depends on several factors, starting with company size. A larger company tends to adopt to a further extent than smaller companies due to their available resources and business needs. On the contrary, a smaller company could possibly be more adaptable due to flatter organizational structures. Furthermore, employees' knowledge of the software proves to be a factor, where more knowledge leads to adoption to a further extent. Finally, the information intensity of a business's product or service makes a difference in the extent of adoption. From this perspective, information intensity might also relate to certain industries being more likely to adopt to a further extent.

1.3 Research question

For this thesis to be successful, it should be determined how an enterprise-wide deployment for Microsoft Power Platform can be executed with a focus on governance. Therefore, the following research question is formulated:

How to arrange IT governance in order to facilitate the enterprise-wide deployment of a low-code development platform at a medium to large enterprise?

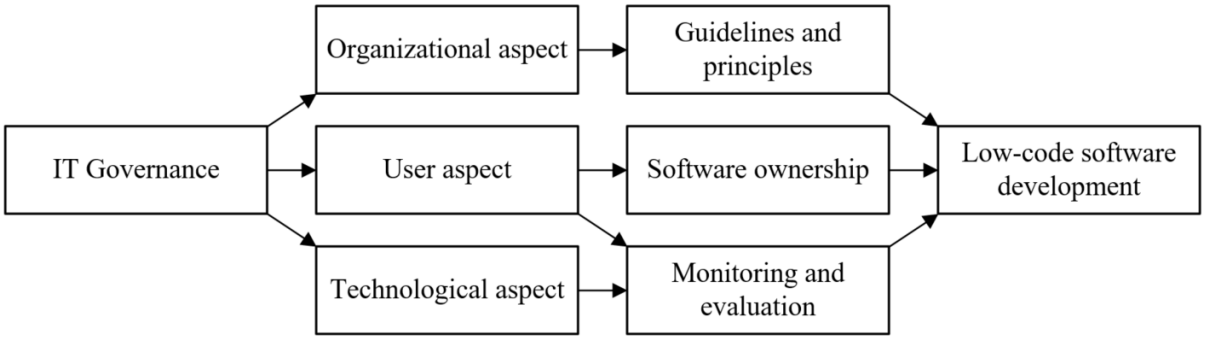
With regards to the previously defined sub-issues (Chapter 1.2.2), the following sub-questions are formulated in addition to the research question:

- *What are the vital guidelines and principles during enterprise-wide deployment of a low-code development platform, to create a safe and robust development environment?*
- *How does ownership of a software artifact have to be arranged once it is implemented within an enterprise?*
- *What has to be monitored and evaluated of a low-code development platform within an enterprise to keep it up-to-date?*

These three sub-questions can be linked to several governance-related aspects, which are depicted in figure 1. From an organizational aspect, guidelines and principles are of interest since they encompass high-level statements about the intended use of an LCDP. The user aspect relates mainly to who has ownership over a software artifact once it is developed. Moreover, the user aspect also relates partly to monitoring and evaluation since user data could be monitored and used for evaluation. The technological aspect revolves around monitoring and evaluation as monitoring is conducted through the technology itself. Evaluation plays a part in the technology as additions to the active policy are implemented within the technology of the platform.

Figure 1

Governance aspects



1.4 Research approach

The main objective of this thesis was to determine how to arrange IT governance for an enterprise-wide LCDP deployment. To reach this objective, this thesis was conducted at a consultancy firm in the LCDP market. A literature review was conducted to develop a deep understanding of the relevant topics in this thesis, preliminary to the data collection. Through literature review the literature gap was defined, leading to the collection of data. Because of the qualitative nature of this thesis, in-depth interviews on a set of sub-issues were used to collect data. This data was coded and analyzed. Following the grounded theory methodology, the findings were interpreted and conceptualized into a guiding framework. Finally, the conclusion was drawn in order to reach the research objective.

1.5 Academic and managerial relevance

This thesis' relevance can be divided up into two categories: academic relevance and managerial relevance. Both of these are described below.

1.5.1 Academic relevance

The academic relevance of this thesis relates to three main topics. First, the existing literature on LCDPs is relatively young and superficial. This thesis covers new insights into its designated use and challenges. Second, citizen development is a relatively unexplored academic area. The existing literature is brief and only describes what is defined as citizen development and what the pros and cons are. This thesis viewed citizen development in the specific context of an LCDP, and how it can be executed as intended. The third topic is IT governance. IT governance is a well-explored area in general. For example, Weill & Ross (2004) define IT governance as: "Specifying the decision rights and accountability framework

to encourage desirable behavior in the use of IT". Furthermore, Weill & Broadbent (2002) constructed a set of five key IT decisions and relevant archetypes to arrange IT governance. However, combining IT governance with the field of LCDPs has not been done before and therefore no specific literature relates directly to the problem.

1.5.2 Managerial relevance

The intention of IT Rebels with the results of this thesis is to determine how they can deploy Microsoft Power Platform enterprise-wide at their clients, while properly arranging governance. This is with the aim of gaining a significant competitive advantage over other Microsoft Power Platform providers by being able to execute a deployment better with regard to quality, time, and budget (Eveleens & Verhoef, 2010). It is to be considered that both the clients' branch and size are not specified by IT Rebels and therefore should not limit this research. However, the main portion of clients is medium- to large-sized companies, situated in the Netherlands.

1.6 Research outline

This sub-chapter elaborates on the outline of this thesis, starting from chapter 2. Chapter 2 contains the literature review for this thesis. Existing literature on the key concepts in relation to this thesis was reviewed in this chapter and the literature gap was defined. Chapter 3 contains the methodology. This chapter covers the thesis's context, the data collection process, and the analysis process. Chapter 4 contains the results. The results that came forth from the data analysis are summed up and presented in this chapter. Chapter 5 contains the discussion. This chapter takes the results from the previous chapter and adds interpretation and meaning to them. Finally, chapter 6 contains the conclusion. The conclusion answers the research question. Furthermore, the limitations of this thesis are discussed and future research areas are suggested. In conclusion, the research implications are presented in this chapter.

Chapter 2. Literature review and theoretical background

This chapter provides an in-depth description and review of the literature on low-code development platforms, IT governance, Enterprise-wide IS, and Citizen development. Finally, the conceptual model is presented. Reliability is a key aspect throughout this literature review. To adhere to this, mostly objective, academic research was gathered and reviewed. Journals were looked up through the WorldCat Discovery database, as well as Google Scholar. When selecting suitable journals, the 'List of good IM journals' defined by TISEM and provided through the course 'MSc Thesis Information Management' was consulted.

2.1 Low-code development platform

The term 'low-code development platform' is derived from the term 'low-code application platform'. The introduction to the term 'low-code application platform' was in 2014 by Clay Richardson and John R. Rymer in their Forrester report on new development platforms. Richardson and Rymer define low-code application platforms as platforms that reduce the required amount of hand-coding, to increase application development speed. In this thesis, it is decided to use the term 'low-code development platform' rather than 'low-code application platform' since nowadays these platforms offer more than only application development, for example, the development of webpages, chatbots, workflows, and dashboards.

The basic idea of low-code software is to raise the level of abstraction during the development process. This is, however, not a new concept as 'model-driven engineering' (MDE) revolves around the same principle. This development paradigm was already introduced in the 1980s as the so-called computer-aided software engineering (CASE) tool (Case, 1985). In practice, MDE supports software development through the construction of models, in order to generate code (Whittle et al., 2014).

The way an LCDP works is through a combination of low-code techniques instead of relying on manual coding, with the purpose of enabling non-programmers to engage in development. These low-code techniques consist of MDE, drag-and-drop features, standard templates, basic logic, and visual interfaces (Waszkowski, 2019). Through these intuitive features, it is possible for end-users with a lack of programming experience to develop and test applications. In addition to the development of applications, also the development of databases, webpages, dashboards, workflows, and APIs are supported through LCDPs.

The benefits of using an LCDP are extensive. Given the rapid development of LCDPs, Gartner predicts that 70% of all newly developed business applications are constructed through

low-code development by 2025 (Wong et al., 2021). Deployment and adoption of an LCDP within an enterprise can lead to rapid application development, which leads to cost savings. A Forrester total economic impact report calculated an average cost saving of 74%. These savings are made up of three components: reduced application development costs, avoided third-party applications, and streamlined activities (Jonathan Lipsitz & Adrienne Capaldo, 2022). The validity of this work however is questionable since the research was commissioned by Microsoft. As Microsoft is an LCDP vendor, the findings of this research could be biased in favor of Microsoft.

Consultancy firm Gartner conducts annual market research on LCDPs. In 2021 they identified twelve significant LCDP vendors, under which five vendors are defined as leaders, three as challengers, and four as niche players in this market, based on their vision and ability to execute. The leading vendors in the LCDP market are: Salesforce, ServiceNow, Outsystems, Microsoft, and Mendix. What they have in common is an extensive array of capabilities for developing business solutions, combined with a clear market understanding and strategy.

2.1.1 Microsoft Power Platform

Microsoft is one of the leading vendors in the LCDP market with the offering of Microsoft Power Platform. Microsoft Power Platform is a suite of programs to build business solutions. The separate Power Platform modules are shown in Table 2 (Microsoft, n.d.).

Table 2

Microsoft Power Platform suite

Tool	**Description**
Power Apps	Make and deploy canvas applications through visual coding techniques.
Power Automate	Construct workflows and connect to applications, teams and people to automate organizational processes.
Power BI	Develop dashboards to showcase real-time business analytics and visuals.
Power Pages	Develop websites through low-code. An additional feature is professional development i.e. high-code.
Power Virtual Agents	Build chatbots across websites, applications, and Microsoft Teams and improve them over time.

Microsoft Power Platform runs on Microsoft Azure (Microsoft's cloud computing service) and therefore inherits its security and compliance features.

2.2 IT Governance for Enterprise-wide IS

Enterprise-wide IS is a term for systems that encompass processes and information across an entire enterprise. A good example of such a system is an Enterprise Resource Planning (ERP) system, as an ERP system aims to capture the administrative workload of an organization (Strong & Volkoff, 2010). ERP implementations have been studied extensively and it is concluded that several factors impact the success of an implementation. One of those factors is IT governance (Williamson, 1999). Moreover, the success rate of ERP implementation projects is significant and positively correlated with IT governance equilibrium (E. T. Wang & Chen, 2006). To get to this conclusion, survey research was conducted among a sample of 1120 enterprises, aimed at the IT managers. Of these surveys, 14,29% were filled in and returned in order to proceed to the analysis.

Following these findings, the logical question to ask is: how do you ensure IT governance equilibrium while executing an implementation? Given the generic nature of ERP systems, realizing a perfect fit is usually not realistic (Strong & Volkoff, 2010). This concept could possibly also apply to LDCPs as the enterprise-wide aim is the same. Six categories of misfits are described, being: functionality misfit, data misfit, usability misfit, role misfit, control misfit, and organizational culture misfit. These categories could be applied to the concept of Citizen development and LCDPs in order to determine if they are relevant or if some might be obsolete.

2.3 Citizen development

Citizen development is defined as software development by users without specific knowledge of coding or a background in software engineering (Gartner, 2022). The term 'citizen development' is widely used in a business context, but it is a relatively unexplored definition in academia. However, all literature leads to the same definition of citizen development (Ng'ambi, 2020). Literature on the term 'end user development' is more extensive and encompasses the same concepts as citizen development. For example, the fitness of end-user development for business applications was already assessed in 1979 as the demand for the extension of information systems was exceeding its capacity (McLean, 1979). Potential use was described however, resources were not sufficient yet to stimulate end-user development.

Over the years more research was conducted within this field, most of which was published between 2002 and 2017 in the United States and Europe (Barricelli et al., 2019). The domains these studies focused on most were: business and data management, web applications, smart objects and environments, games and entertainment, and education. To reach these findings, a literature review was conducted, initially encompassing a sample of 2717 papers. Through filtering these papers on criteria such as duplication and validity, a final sample of 165 papers was reviewed.

Over the years more research was conducted in this field, leading to the development of several techniques. The most frequently described techniques in academic literature are Component-based development, Rule-based development, and Programming by demonstration (Barricelli et al., 2019). Component-based development is, as its name suggests, based on the usage of defined components. These components are reusable and therefore decrease development time and increase reliability (Sommerville, 2011). Rule-based development takes certain variables and composes them into logical rules. These rules can then be triggered, e.g. by a person or a device, in order to execute the desired action (Ghiani et al., 2017). This study aimed to determine how end-users (without programming knowledge) could personalize applications by running several field experiments in which end-users saw different interfaces while developing. Programming by demonstration is a technique that records certain steps or actions performed by the user and translates this into a program within a system (Dey et al., 2004). Recording macro's in Microsoft Excel is an example of programming by demonstration, as it records the steps a user takes and composes a program out of this, all within one system i.e. Excel.

In addition to several techniques, also the type of application to be developed is an interesting research field since multiple types can be defined within an enterprise. Literature suggests that there are three distinct types of applications within an enterprise: Personal applications, Departmental applications, and Corporate applications (McLean, 1979). Each application type supports a certain type of usage, either for personal use, across departments, or even enterprise-wide. The complexity and criticality of an application increase as more users, data, and connectors are issued in its use. This proposes an area for discussion as there is no academic research on citizen development and the types of applications, with regard to complexity and criticality, they can develop.

Understanding a problem to define requirements is a vital part of software engineering (Anton, 2003). Since development is executed by the end user, one of the main impediments in software development i.e. requirements planning is mitigated. As the end users are the individuals experiencing the problem, they have extensive knowledge of the requirements and are able to implement this into the solution. This is opposed to an external consultant who first has to get an understanding of all processes and systems in place before being able to gather requirements. These benefits of citizen development, however, induce risks regarding application quality, security, redundancy, and performance (Oltrogge et al., 2018). Therefore, encouraging citizen development does not guarantee successful development and might even lead to new problems of a greater extent if not governed correctly.

2.4 IT Governance - background

IT governance is defined as "specifying the decision rights and accountability framework to encourage desirable behavior in the use of IT" (Weill & Ross, 2004). IT governance is an academically well-explored area, however, it is also relatively encompassing and complex. For clarity, it is to be noted that IT governance is not about the specific decisions that are made, nor is it about management. The emphasis is on who makes decisions, how they are made, and who is held accountable. The main goal and benefit of IT governance is deriving value from the use of IT, usually in the form of performance (Wu et al., 2015). Within the IT governance paradigm, there are three distinct dimensions. The first dimension is the focus of IT governance, relating to what is governed. In this case, the 'what' refers to either artifacts, data, or stakeholders. The second dimension is the scope of IT governance, which describes who is governed. The measure reaches from the project or application level all the way to the eco-system level. The third dimension is the patterns of IT governance, concerning the way how IT is governed. This dimension refers to decision rights, control mechanisms, and finally architecture (Tiwana et al., 2013) (Gregory et al., 2018). A widely used method of arranging compliant IT governance is the COBIT 2019 framework. COBIT helps to ensure the effective enterprise governance of IT (Ridley et al., 2004). COBIT does so by offering an extensive set of control objectives for an enterprise to base its governance upon.

In relation to this thesis, the framework on effective IT governance borrows itself specifically well due to the topics it addresses (Weill & Ross, 2004). Moreover, this framework presents the harmonization of business objectives with IT governance archetypes and business performance goals. The business objectives describe the results an enterprise hopes to achieve,

supported by IT. The IT governance archetypes summarize the input- and decision rights concerning divergent IT domains within the enterprise. The business performance goals are short-term, measurable objectives that ultimately relate back to the overarching business objectives. These three topics represent the desired achievements when aiming for effective IT governance. Each achievement has a measurement at its foundation, which also needs to be harmonized: business objectives are harmonized by desirable behavior, IT governance archetypes are harmonized by IT governance mechanisms and business performance goals are harmonized by metrics.

With this framework on effective IT governance, a matrix is provided for determining the input- and decision rights in relation to the different archetypes and the decision domains. This framework and matrix are both provided in Ross and Weill's book on IT governance (Weill & Ross, 2004). The development of this framework and matrix was described in a series of three papers. The initial paper was by Weill and Woodham on implementing effective IT governance (Weill & Woodham, 2002). The matrix was then further developed by Weill and Broadbent in their paper on describing and assessing IT governance (Weill & Broadbent, 2002). In addition to the development of this matrix itself, another paper focused on the most used approaches and the top-performer approaches in relation to this framework and matrix (Weill & Ross, 2004b).

IT decision domains

The IT governance arrangements matrix encompasses the different decision domains in relation to IT governance and plots them against several archetypes with regard to input- and decision rights. These archetypes are business groups or individuals, for example, a group of C-level executives or individual users. The first domain in this matrix is IT principles. This domain covers high-level statements about IT usage throughout an enterprise. The next domain is the IT infrastructure domain, which relates to the approach of building an enterprise's IT foundation. IT architecture is the following domain and encompasses the technical choices an enterprise has to take to satisfy its business needs. The next domain is the business application needs domain, which covers the business applications that are desired, so they can either be bought or built. The final domain is IT investments. This domain describes the decisions on IT investments in the form of money, time, and effort.

IT governance archetypes

For each of the decision domains, an archetype has input- and/or decision rights. The first defined archetype is the business monarchy. A business monarchy consists of the C-level executives either individually or in a group, excluding the CIO as an individual. The next archetype is the IT monarchy consisting of IT executives either operating as a group or individually. Next is the feudal archetype, involving the relevant business unit leaders or its delegates. The following archetype is federal, which is relatively indistinct as opposed to the other archetypes. This archetype consists of some combination of C-level executives and one or more other business groups, for example, end users and IT executives. Next is the duopoly archetype, involving the IT executives and one other business group. Finally, the anarchy archetype means the governance right rest upon each individual user.

2.4.1 IT governance on guidelines and principles

Within the IT governance domain, defined guidelines and principles offer guidance in determining how certain decisions are made. A relevant principle regarding this domain is policy-driven governance, as this secures an enterprise's continued compliance of an information system. An example of policy-driven governance is a data-loss prevention (DLP) policy. DLP is a method to track the information streams throughout an enterprise, to stop end-users from leaking sensitive data, and to safeguard existing data storage (Wuchner & Pretschner, 2012). A DLP policy describes the way data has to be handled throughout an enterprise to arrange the mitigation of data losses. Such policies and documents are fundamental in the configuration and implementation of new (and existing) information systems, as those systems have to conform to the enterprises' compliance level. In addition to security, providing guidelines also contributes to standardization within an LCDP.

In relation to the decision domains of the IT governance arrangements matrix, the subject 'guidelines and principles' fares under the IT principles domain. These guidelines and principles cover the desired usage of an LCDP within an enterprise. An enterprise could for example implement an LCDP and choose to enable citizen development for certain purposes, and mere professional development for business-critical purposes. These choices and their relation to the company's strategy are defined and documented within the IT principles domain. Following Weill and Ross's (2004) research, the most common archetype with input rights for this domain is federal. The most common archetype with decision rights is the IT duopoly. This implies that input rights rest upon a composition of C-level executives and one or more other

business groups. Concerning the decision rights, it shows that IT executives are always involved, complemented by one other business group.

2.4.2 IT governance on software ownership

Ownership in relation to an enterprise's information systems is a vital principle in IT governance (Grembergen, 2003). Software ownership determines the entity that formally owns a software artifact within an enterprise. The importance of unambiguous arrangements in relation to software ownership arises from a quality perspective. For example, research from Bird et al. presumed that a higher number of contributors results in a higher amount of failures in software development (2009). This finding could support the motivation of properly arranging software ownership to control the number of contributors. More research provided evidence for the hypothesis that a higher amount of contributors positively correlates with the number of failures within a software artifact (Bird et al., 2011). This proves that a lower, controlled number of developers make lesser mistakes. Moreover, this research mainly focused on the relationship between software ownership and the number of development failures. The results are that a higher level of ownership negatively correlates with the number of software failures. The level of ownership is measured as the percentage of total commits, made by the top contributor. So, a higher percentage of commits leads to a higher level of ownership. In addition to the owner, a distinction is made between major- and minor contributors. A major contributor is a developer that has made 5% or more of all commits whereas a minor contributor has made less than 5% of all commits.

This provides an interesting paradigm for approaching the business application needs and IT investment domains from the IT governance arrangements matrix. Literature suggests that the anarchy archetype is rarely used in general (Weill & Broadbent, 2002). However, when considering both ownership as described in this sub-chapter, as well as citizen development as described in chapter 2.4, the anarchy archetype proposes some joint properties. Within the context of this thesis, this provides an interesting perspective to further explore.

2.4.3 IT governance on monitoring and evaluation

Once IT governance is properly arranged, it has to stay up to date with new strategies, policies, and compliance obligations (Bowen et al., 2007). Therefore, it is an iterative domain. Monitoring and evaluation serves as a foundation for the further improvement of IT governance. For example, if over time too many unsupervised applications are developed, this

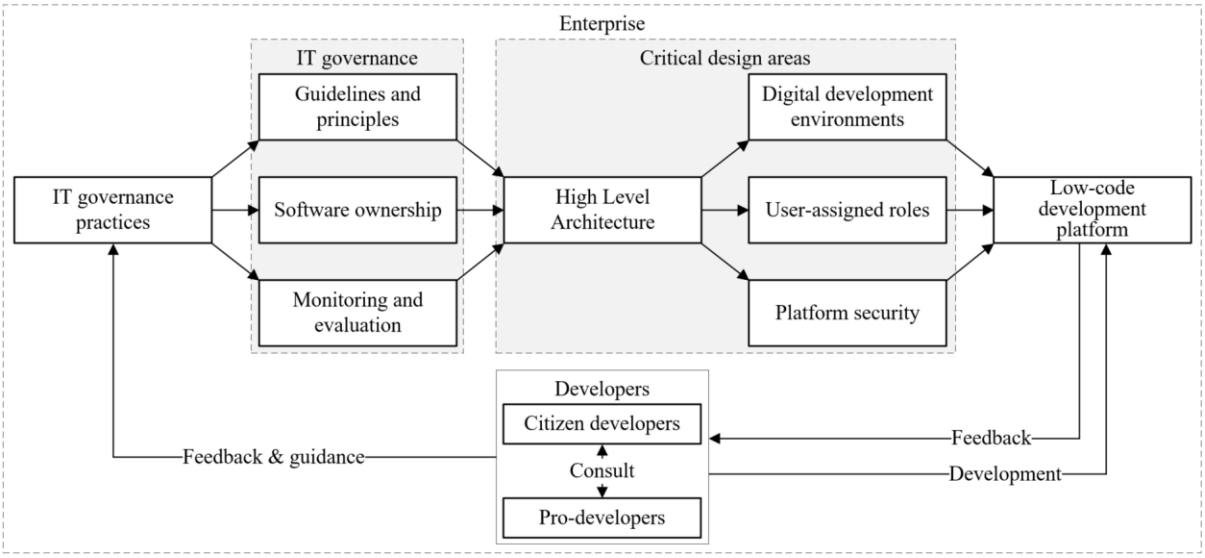
would be indicated through monitoring. This provides a basis to update the IT governance arrangements to prevent such unsupervised application development. Moreover the importance and relevance of monitoring and evaluation, research found that there is a significant, positive correlation between monitoring and IT governance performance (Simonsson et al., 2010).

2.5 Literature gap and conceptual model

This sub-chapter describes the literature gap that was identified through the literature review. The literature gap covers three main topics. The first topic is LCDPs. Literature on LCDPs is brief as this is a relatively new domain, but the basic concepts are documented in several academic journals. In addition, extensive documentation on enterprise-wide IS can be generalized toward LCDPs to complement the existing literature. The second topic is citizen development. As well as LCDPs, also citizen development is a relatively unexplored area in academia, although the literature on related terms such as end-user development can be used to a certain extent. The third topic is IT governance. IT governance is an extensively researched subject with lots of academic documentation on its constructs, also including frameworks. In the case of this thesis, the framework on effective IT governance (as described in chapter 2.4) provides several relevant constructs that can be applied to the governance arrangements with regard to LCDPs. However, this framework mainly focuses on traditional IT processes and development and does not account for the unexplored domains of citizen development and LCDPs.

Figure 2

Conceptual model



Regarding this thesis, the developed conceptual model is presented in Figure 2. This model shows that IT governance proposes three main challenges on the topics: guidelines and principles, software ownership, and monitoring and evaluation. These challenges are countered by focusing on the critical design areas, captured in a high-level architecture. These critical design areas aim to create a robust and safe LCDP within an enterprise. Both citizen developers and pro-developers develop software artifacts on this LCDP. In addition, they gather feedback from working on the LCDP. This feedback can then be used to provide guidance for encountering and optimizing the IT governance practices.

This conceptual model distinguishes itself in several aspects. The main difference is that the literature focuses on professional development and not on citizen development. However, pro-developers have experience and education in their field of work, whereas citizen developers are without any experience or education. Pro-developers, therefore, know how to act and behave when handling data and connectors for example. For citizen developers, extra measures are needed to guarantee security and quality throughout their developed software artifacts. Moreover, traditional literature assesses mainly traditional development. This thesis revolves around low-code and no-code development. This type of development is based on several techniques that are not used in traditional development. Finally, the current literature does not cover IT governance in the specific fields of citizen development and LCDPs.

Chapter 3. Methodology and data

This chapter describes the used methodology throughout this thesis and what data was gathered and analyzed. First, the research context is described. Then, the data collection method is explained and subsequently, the data analysis is presented. Finally, some remarks on the validity and reliability of the thesis are given.

3.1 Research context

This thesis was written in collaboration with IT Rebels. IT Rebels is a consultancy firm, whose core business revolves around the development, implementation, and maintenance of divergent business solutions for their clients. All of these solutions are developed with Microsoft Power Platform. IT Rebels' goal is to develop fitting business solutions for their clients. The first step in this process is gathering requirements followed by process analysis and setting up a data infrastructure. Subsequently, a prototype is developed and validated. If the solution is satisfactory, deployment and implementation are arranged. Finally, IT Rebels manages and maintains its delivered solutions and adds features on demand (Rozendaal, 2022). Throughout their services, IT Rebels aims to reduce clients' workload while integrating and streamlining processes and teams (IT-Rebels, 2021).

Following IT-Rebels as a participating company, the main LCDP that was discussed throughout the course of this thesis was Microsoft Power Platform. Furthermore, this thesis focuses on the development and deployment stages, not primarily on maintenance. The development of Microsoft Power Platform itself goes at a rapid pace. Features that are now unheard of, might be available in the future. Therefore, the thesis will focus on the existing components and possibilities within Microsoft Power Platform. Considering the client's company size, the scope will be on medium- to large-sized companies, since this is the main portion of IT Rebels' clients.

One of the main topics in this thesis is the facilitation of citizen development throughout an enterprise. This development approach differs from traditional, professional development. However, although the focus is on citizen development, professional development is also mentioned in this thesis and the interviews. The reason for this is to be able to compare fundamental differences between professional development and citizen development and to determine to what level they complement each other.

3.2 Data collection

This sub-chapter describes the data collection process that was executed during the course of this thesis.

Methodology

This thesis follows a qualitative approach, mainly based on interviews. The reason for this is that although there are relatively many academic publications on the topics of IT governance and enterprise-wide IS, academic knowledge on LCDPs and citizen development only reaches a certain extent. In addition, the phenomenon of combining IT governance with enterprise-wide LCDPs, to ultimately facilitate citizen development is a new and academically unexplored area.

The primary method to gather data was conducting interviews. More particularly, these were in-depth, open-ended, semi-structured interviews. In-depth interviews are the best fit for this thesis as its nature is qualitative (Fontana & Frey, 2000). Furthermore, a deep understanding of the topic is essential to answer the research question and its sub-questions. In addition, the interviews were semi-structured. This is the most fitting interview structure as the topic and the research objectives are clear, but detailed answers are needed for analysis. The data was collected between September 2022 and January 2023.

The aim of this thesis is to determine the crucial IT governance arrangements when deploying an LCDP at an enterprise-wide level. To do so, the grounded theory method was preceded to construct a framework that provides the foundation for deploying an LCDP (Birks et al., 2013). Grounded theory proposed the best fitting methodology as the objective of this thesis is to develop a new theory rather than testing an existing one. These arrangements are overarching, qualitative definitions that are not necessarily expressed in numbers.

Interview selection criteria

The interviews were conducted with interviewees that adhered to predefined selection criteria. The interviews were conducted in two waves, where both waves had different criteria. The main criteria for the first wave of interviews were that the interviewees had to be involved at a policy level with the enterprise-wide deployment of an LCDP at the enterprise they work for, and had decision rights during the process. As they have experience on the topic of this thesis, they provided essential input to shaping this thesis. Furthermore, the interviewees had

to have experience in the field of IT as this thesis focuses on technical aspects. Moreover, a minimum of five years of working experience was required.

The selection criteria for the second wave of interviews focused on the specialization of the interviewees. They were selected based on their extensive knowledge of- and experience with IT and LCDPs, but on a practical level. Through their practical insights, the findings from the previous wave could be assessed. The interviewee's profiles are shown in Table 3.

Table 3

Interviewee profiles

Wave	Function	Company	Branche	Years of experience	Date
1	Solution architect	A	Public transportation	8	7/11/2022
	BI consultant	B	Government	15	11/11/2022
	ICT architect	C	Government	14	14/11/2022
2	Technical consultant	D	Managed services	11	22/12/2022
	Developer	E	Consultancy	2	29/12/2022

Interviewing process

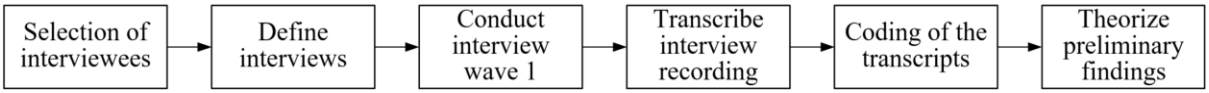
This paragraph describes the exact process that was followed for conducting the interviews for this thesis. As previously stated, the interviews were conducted in two separate waves.

Interview wave 1

The process for the first wave of interviews is shown in Figure 3.

Figure 3

First wave of interviews



The first step for the first wave of interviews was selecting the interviewees based on the previously described criteria. The next step was taking into account their expertise and defining the interview protocol and questions. The interview questions were designed around the sub-issues that are described in chapter 1.2.2 and also the interview structure followed these issues as separate sections. The next step was conducting the interviews. In the first wave, three interviews of 45-60 minutes each were conducted through Microsoft Teams. The interviews

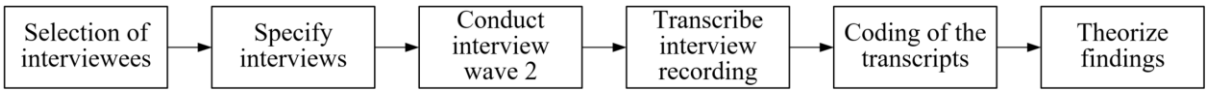
started with an introduction round and an explanation of the research. Then the structure of the interview was explained and confidentiality was assured. It was explicitly asked by the interviewer if the interview could be recorded for transcription purposes. To record the interviews, Microsoft Teams was used. This first wave took place in November 2022, the entire interview protocol can be found in appendix 1. Subsequently, the interview recordings were transcribed using transcription software from Microsoft Stream. The rough transcripts were then manually checked and corrected. The next step was coding the transcripts based on keywords. First, all quotes from the interviews were separated from each other. Based on keywords, the quotes were given one or more labels to identify them. After this first round of coding, a second round was executed. In this second round, the quotes were filtered by their preliminary code and they got assigned several overarching labels. The final step in the coding process was linking the overarching labels to the different topics from the conceptual model. Finally, the preliminary findings from the first wave of interviews were theorized and used as further input for the second wave of interviews.

Interview wave 2

The process for the second wave of interviews is shown in Figure 4.

Figure 4

Second wave of interviews



The second wave of interviews started with selecting the interviewees. The preliminary findings from the first wave of interviews provided input for the selection criteria that were used. The second step was taking the interviews of the first wave and specifying them based on the preliminary findings and interviewees. The second wave of interviews consisted of two interviews that lasted 60-80 minutes each. One of them was conducted and recorded through Microsoft Teams and one was conducted in person and recorded using the voice recording function of an iPhone 12 in an enclosed room. These interviews were conducted in December 2022. The next step was transcribing the recordings. One of the recordings was transcribed using Microsoft stream and the other one was transcribed using Amberscript. The rough transcripts were manually checked and corrected. Subsequently, the transcripts were coded based on keywords. The first step was separating all quotes from the interviews and labeling them based on keywords. The next round of coding considered the preliminary codes and

overarching labels were assigned. The overarching labels were then linked to the topics from the conceptual model. Finally, all results, also those from the first wave of interviews, were theorized.

All interviews were conducted in Dutch as this was the native language of the interviewees. The transcripts are available on request. Relevant quotes were translated into English and presented in chapter 4.

3.3 Data analysis

The data gathered from the interviews were used as input for analysis and drawing conclusions. In order to do so, the interviews were recorded, transcribed, and coded as described in sub-chapter 3.2. The quotes from the interviews were sorted based on the assigned codes. This led to several shortlists of quotes containing comparable statements and information. Sets of comparable quotes were abstracted based on their essence and assigned to the topics from the conceptual model. The data could then be filtered based on the assigned topic from the conceptual model. This produced a list of several sets of quotes, covering different aspects of the chosen topic. These findings are presented per topic in chapter 4. Chapter 5 further interprets these results and theorizes them based on their collective message in relation to the topics.

External validity

To constitute the external validity of this thesis the concept of LCDPs was the focus, rather than a specific LCDP vendor. However, all interviewees had experience with one LCDP in particular, being Microsoft Power Platform. This could be explained by the fact that Microsoft Power Platform is the main development tool for the collaborating consultancy firm of this thesis. Regarding the external validity of this thesis, this could mean that the results are applicable to Microsoft Power Platform, but that they possibly do not generalize toward other LCDPs.

Chapter 4. Results

This chapter describes the gathered results throughout the course of this thesis. The results were derived from a set of interviews with professionals. The first results that are presented cover a set of governance challenges for an LCDP deployment. The second section revolves around a set of design areas when configuring the LCDP. Finally, the results on citizen development are presented.

4.1 IT Governance challenges

Through several conversations and interviews, a set of three main governance challenges were identified for an enterprise-wide LCDP deployment. These challenges are described in the following sub-chapters, starting with guidelines and principles, followed by software ownership, and finally monitoring and evaluation.

4.1.1 Guidelines and principles

Guidelines and principles cover internal compliance to an enterprise's defined set of rules. There is no best practice among these guidelines in relation to LCDPs. Therefore, the development process is open for each individual's interpretation, whereas unambiguity would contribute towards compliance. A broad idea of the rules should be in place, as everyone understands that a software artifact needs to be secure. But without guidelines, everybody can interpret security in their own way. Guidelines and principles give consensus about the rules and what they mean when participating on the LCDP.

The main principle that has to be applied within an LCDP is the classification of software artifacts based on their business criticality. Business criticality relates to the importance of a software artifact for an enterprise. There should be a division between software artifacts for personal productivity and business-critical software artifacts. Personal productivity is the development of software artifacts strictly for personal use and excluded from any sensitive data. Business critical software artifacts contain sensitive data and/or span across departments. Concerning business criticality, the following was stated:

"The division should be: If you want to automate something for your own productivity, that is fine. If you want to share it with a colleague, you have to consult your manager. And if it becomes business critical, it has to be handed to professional development." - (company D)

"*Data is regularly connected and you want to be in control of what is exactly connected"* - (company B).

Moreover, collaboration between developers and the IT department should always occur in order to contribute to the quality and security of an artifact. Regarding collaboration with the IT department, the following was said:

"[…] when constructing that mailing list for example, that should always be done in collaboration with IT, that is mandatory." - (company D).

An enterprise should initially use the LCDP for personal productivity and gradually expand towards more business-critical development. Statements on the gradual expansion of an LCDP were:

"Our intention is to keep it small and tight at the start, and expand step by step. […] Automation of processes should not be done by everyone at the beginning." - (company B)

"At first, it is mainly for personal and individual use. So solutions spanning further than personal productivity should not be developed yet." - (company A).

The key takeaways with regard to guidelines and principles are: make a division between personal productivity and business-critical software artifacts. Development should always be executed in collaboration with the IT department. Finally, an enterprise should gradually increase the intensity of its LCDP usage, shifting from personal productivity to business-critical software artifacts.

4.1.2 Software ownership

Software ownership relates to the user aspect of IT governance. A method for arranging ownership has to be in place when an application is being developed and deployed and who has ownership over the larger parts of the LCDP altogether. History has taught that in similar situations, ownership was for the developer of an artifact until something went wrong. The moment an artifact proved to be insecure or faulty, the IT department would be seen as the owner. This then led to IT departments being held responsible for the uncontrolled behavior of other employees. Moreover, there is also the risk of the initial developer leaving the enterprise without providing sufficient documentation on the artifact.

In order to stay in control, decisions about software ownership of an artifact have to be made upfront. In response to the interview question as to why ownership is an important aspect with regard to IT governance, one response was:

"You want to avoid a similar situation to Access, where everyone has ownership over their creations, but the moment it goes wrong the IT department is held accountable." - (company A).

Ultimately, ownership of an artifact should be retained by the lead developer. The lead developer in general is the top contributor, making most of the commits to an artifact. In addition, the lead developer should have sufficient knowledge of the technology to guarantee quality and security in order to get ownership. in response to the question of who should have ownership over a software artifact, it was stated:

"It is the application lead-developers responsibility to guarantee the application's security and access to data. […] So ownership of an application has to be for the lead developer, the project lead." - (company E).

Moreover, if the software artifact reaches a certain point of enterprise criticality, ownership should always be given to the IT department. Enterprise criticality is further covered in chapter

4.2.3 Platform security. On the contrary, if an artifact is strictly for the personal use of an employee, the consequences of failure are not detrimental to the enterprise. In such a case, it would be correct to assign ownership of the software artifact to the lead developer, even if this is a non-IT employee. It was stated that:

"IT was responsible in the end when things did not work. They did not like it, but therefore IT became the owner and they had to restrict development by other employees." and: *"As long as failure does not hurt the enterprise, it is okay. But if it covers a critical process, ownership should be with someone with sufficient technical skill."* - (company D).

The key takeaways regarding software ownership are: Software ownership is one of the main governance challenges in the area of LCDPs. The lead developer should have ownership of the software artifact. If a software artifact covers a critical process, the owner should have sufficient technical skills to guarantee security. If a software artifact is not detrimental to the enterprise, the owner does not necessarily need substantial technical skills.

4.1.3 Monitoring and evaluation

Monitoring is the practice of collecting feedback data from the LCDP and evaluation refers to using this data to adjust governance arrangements where needed. Adjustment of governance arrangements can help to stay compliant and in control. For example, if the amount of newly developed software artifacts is not monitored, capacity can run out.

Monitoring is essential for an enterprise to stay in control of its LCDP. During the interviews, it was stated that:

"We have the need for a tool that can monitor the platform as a whole, so we can track our whizzkids, who develops solutions, how many applications are running […]" - (company B)

"And platform-wide you want to see if the database can take it since you have different capacities, one for the true database information, one for file information, and one more." - (company E)

The usage of connectors within and between software artifacts is an aspect that should always be monitored. This keeps the managers up to date on which external sources are connected to the enterprise's database. A quote specifically on the topic of connectors was:

"Look, often you are going to make connections and we need to have a grip on what is being connected." - (company B)

Complementary to connectors, also an overview of the running software artifacts should be present. In addition, also capacity has to be monitored since an LCDP can run out of digital capacity i.e. storage space. This means that no new data can be saved and the development of artifacts comes to hold. Furthermore, the number of developers, licensing model, role assignment, device usage, and sign-ins are aspects that should be monitored on an LCDP. More in-depth, it was stated that:

"You want to monitor the path people take when developing and if there are enough licenses. How many users are there, which roles they have, which devices are used, those are all relevant" - (company E)

Finally, depending on the in-house IT capabilities, monitoring should be partially outsourced to a party that can deliver support in case of calamities. On the topic of outsourcing the following was said:

"Basic monitoring should be outsourced, we should arrange managed services so there is control over new situations that need addressing." - (company C).

The key takeaways with regard to monitoring and evaluation are: you need monitoring and evaluation in order to stay in control. Important aspects to monitor are connectors with external sources. Furthermore, capacity has to be monitored, as well as developers, licenses, roles, devices, and sign-ins. Finally, if IT capabilities are not sufficient, monitoring should be outsourced to a third party for support.

4.2 Critical design areas

Through the conducted interviews, three main critical design areas were identified. In addition to the governance challenges, these design areas propose specific components within an LCDP. These critical design areas are described in the following sub-chapters, starting with digital development environments. Subsequently, user-assigned roles are covered and finally, the findings on platform security are presented.

4.2.1 Digital development environments

Within an LCDP, the development of software artifacts should not occur all in one place. If all artifacts were to be developed in the same digital space, the platform becomes cluttered and the same measures would apply to every developer. To overcome this, digital development environments are used. An environment is a digital space in which development is executed. Different environments can facilitate different purposes. Environments can be divided based on the department for example. Each environment can be configured individually in order to tailor them to their intent. On an LCDP, several environments exist and within these environments, roles are assigned to users and artifacts are developed.

Concerning the environments within an LCDP, the main principle to adhere to is to disable every function except what is minimally needed. During the interviews a quote on the environment's functions was:

"Yes, we believe that everything should be closed in the basis, and that it can be opened on request […]." - (company C).

If a new function is needed, developers can request to enable it. In that case, an admin decides whether the function should indeed be enabled. A quote on this was:

"The philosophy is that we close down as much as possible by default. And functions can be opened up when requested." - (company B)

There is no set way for dividing environments, however, an example would be designated environments for sales, management, and finance. Coherent to this division, certain groups have access to the designated environments. Whatever the division of an enterprise is, the environment structure always needs a development environment, a testing environment, and a production environment. The development environment is the place where the software artifact is developed. It is then exported to the testing environment to test the artifact. Finally, if the software artifact is approved, it is exported to the production environment so it can be used. Concerning the different types of environments, the following was said:

"We always use DTP (development, testing, production). We scrapped the A (acceptance) as that is already part of the LCDP" - (company E).

4.2.2 User-assigned roles

User-assigned roles are roles given to a user to grant them certain rights within the LCDP. In addition to the previous sub-chapter on environments, a user could have different roles in each environment he has access to.

The main criterion with user-assigned roles is that they should be defined in groups. This means that there are no individual roles, but there are standardized groups. A user can then be added to a group to inherit the role. This role gives the user certain rights and limitations on the LCDP. In the interviews, the following was said about roles:

"I think users should never have individual rights, so it has to be arranged in groups. I suggest three or four standard groups" and: *"Only if it is absolutely necessary, extra roles can be defined."* - (company C).

Furthermore, the lead developer should be the person to decide which role is given to each user, as he is aware of the appropriate rights and limitations. On the allocation of roles, the following was stated:

"Access to an application and its data should be arranged by the lead developer." and: *"He (the lead developer) has to think about it because he is the person that knows what types of users there are […]"* - (company E).

4.2.3 Platform security

The main topic regarding platform security is data-loss prevention (DLP) policy, in addition to several topics discussed above, such as environments and roles. DLP refers to the measures that are taken in order to prevent data from leaving the enterprise's boundaries.

The DLP policy should include measures with regard to data and software access, connector usage, and auditing. During the interviews, the following about platform security was mentioned:

"Regarding security, it is of importance that there are DLP policies defined." and: *"DLP policies are enterprise-wide and focus on the management and configuration of the LCDP."* - (company E).

Access to data and software determines which user can see which data and applications. A user should only be able to access data designated to him, and nothing more in order to minimize the chance of data loss. On the accessibility of important data, the following was said:

"Sensitive data is dangerous, you should actively close access if it is not needed." - (company C).

To verify if security is up to standard, security should be periodically audited by a specialized party. With regard to auditing it was quoted:

"So if an app collects data, we have to let it be audited by a third party to see if we did all right." - (company C).

4.3 Citizen development

Citizen development is the development of software artifacts by users without specific knowledge of IT and coding. As IT departments might be working at capacity, citizen development proposes benefits as the development of software artifacts can be executed without IT being in the lead. However, IT not being in the lead of development also induces risks as citizen developers are not specialized in development. Therefore it is important to arrange an LCDP in such a way that citizen development can be executed both efficiently by the developers, but also securely for the enterprise.

The main aspects with regard to successful citizen development within an enterprise are threefold: first, citizen development in collaboration with the IT department, second, the

classification of data and software artifacts, and third, gradually increasing the intensity of the LCDP usage. Although citizen development aims to relieve the workload of the IT department, citizen development should never be conducted without consulting IT. An LCDP is very accessible, but it remains an IT domain and faulty development can hurt the enterprise. Therefore, independent of the importance of a software artifact, citizen developers should always collaborate with the IT department. With regard to the collaboration with IT, the following was stated during the interviews:

"They have to work at or with the IT department. […] Then at least some higher thought is put into development, just imagine if someone without IT knowledge starts developing." and: *"Yes, so citizen developers had to always be consulting the IT department"* - (company D).

Regarding the importance of a software artifact, an enterprise should have a well-thought data classification model in place. Based on the data and the function of an artifact, it should be determined if the artifact is either for personal productivity or if it is business critical. Citizen developers should be able to develop applications for personal productivity, but when an artifact becomes business-critical it should be developed by professional developers. Regarding the importance of artifacts, the following was said:

"It is twofold. We are going to develop applications with our own community […] But when an application is for external use it should be developed by a professional." - (company C). and:

"The division should be: If you want to automate something for your own productivity, that is fine. If you want to share it with a colleague, you have to consult your manager. And if it becomes business critical, it has to be handed to professional development." - (company D)

Furthermore, an enterprise should always gradually increase the usability of its LCDP. This is to get the developers familiar with the platform in a controlled manner, with less risk of hurting the enterprise if something is faulty. On gradually increasing platform usage, the following was stated:

"In the beginning, they will recognize a flow and stuff, but they will not be able to properly work with it. So you have to gradually increase it, in a structured manner." - (company B)

"At first, it is mainly for personal and individual use. So solutions spanning further than personal productivity should not be developed yet." - (company A).

The key takeaways regarding citizen development are: although citizen development happens outside the IT department, IT should always be consulted so they can manage citizen development. Furthermore, citizen development should only occur for data and software artifacts used for personal productivity. When it becomes business critical, the IT department should take over development. Finally, citizen development should be gradually increased, following a clear structure so developers can get used to the LCDP.

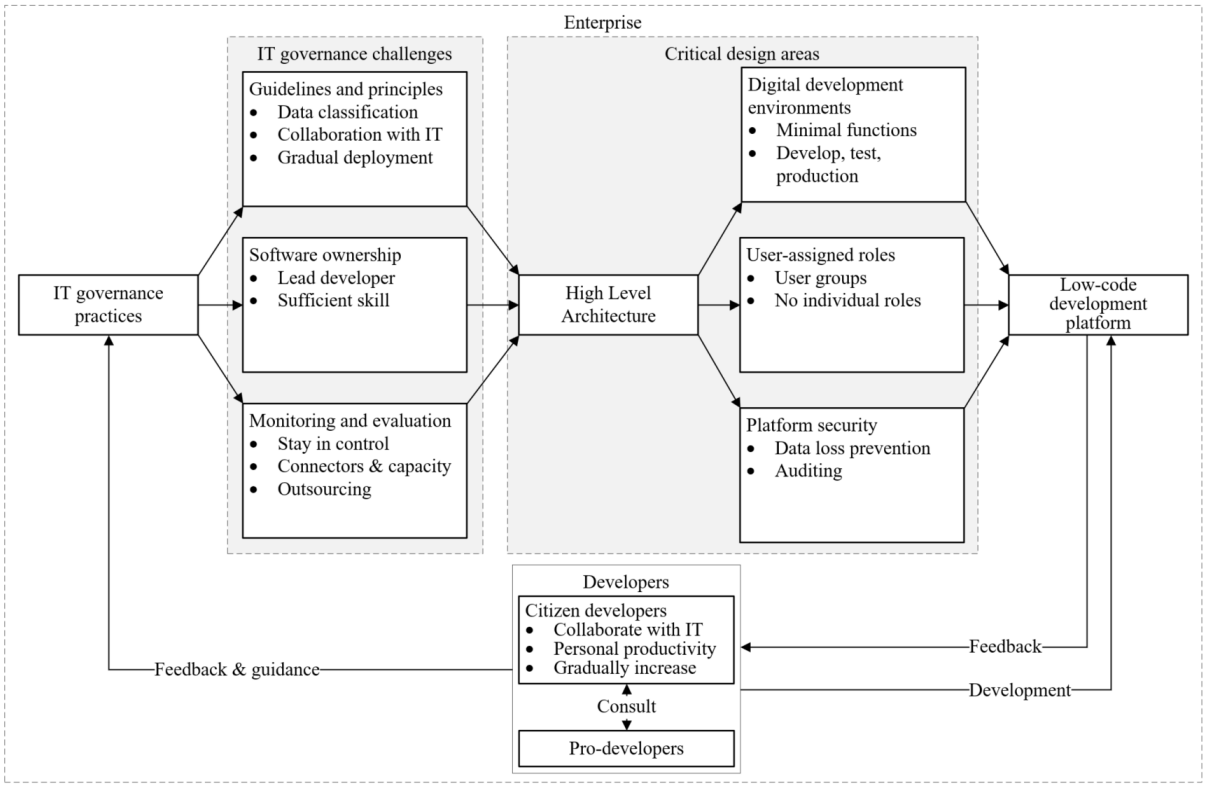
Chapter 5. Discussion

Throughout the course of this thesis, the research topic was scoped down, research questions were formulated, existing literature was gathered and a methodology was constructed. This led to the collection and analysis of data that will ultimately be used to answer the research questions. This chapter focuses on the meaning and interpretation of the results in relation to the enterprise-wide deployment of an LCDP.

The results of this thesis proposed several insights on the topic of IT governance, citizen development, and LCDPs. The cohesion between the gained insights proved to be valuable in defining the answer to the research question. An overview of the results in relation to the conceptual model is shown in figure 5.

Figure 5

Research model



5.1 IT governance challenges

The IT governance challenges were identified in an early stage of this thesis and they were finetuned during and after the conducted interviews. For each of these three challenges, several decision domains are formulated based on the results of this thesis. These decision

domains are topics an enterprise has to consider when deploying an LCDP enterprise-wide. They do not specifically determine the way an enterprise should arrange its IT governance as that is not what IT governance is about. IT governance revolves around the decision domains and who makes these decisions rather than the answer to these decisions.

In relation to the theoretical background of IT governance, these findings suggest that an IT duopoly should have decision rights regarding this challenge (Weill & Ross, 2004). The reason for this is that these guidelines and principles are defined and managed by IT, in collaboration with the developers. Software ownership can take on either the IT monarchy or the anarchy archetype, based on data criticality. If the data has low criticality and the software artifact is used for personal productivity, the anarchy archetype would fit best since the lead developer should be the owner. However, if a software artifact has a high business criticality, the IT monarchy is the best fit since IT should be the owner. Concerning monitoring and evaluation, IT duopoly would propose the best fit. The duopoly would again consist of both IT and the developers as together they collaborate on the LCDP.

5.2 Critical design areas

The concepts from the IT governance challenges are captured into the high-level architecture of an enterprise, translating them into three critical design areas. Each of these critical design areas proposes specific decisions that are in line with the governance challenges. These design areas were defined during the explorative interviews and they make up the LCDP. The main focus was on limiting access and rights to developers as much as possible. This can be explained by the fact that a citizen developer does not have IT experience and therefore is more prone to make mistakes. By limiting their rights, the chances of data loss are mitigated. On the contrary, it is interesting that training or education of citizen developers was not explicitly mentioned. This implies that an enterprise rather limits the possibilities a citizen developer has on the LCDP, than providing training and loosening the limitations.

Another interesting finding was that no overarching structure for the division of digital development environments was proposed, other than development, testing, and production. An intuitive way of dividing environments would be based on business units. This means that each department (e.g. sales, finance, marketing, IT) has its own development, testing, and production environments. However, the phenomenon of LCDPs could still be too premature for a lot of enterprises to define a best practice on environment division.

5.3 Citizen development

The expectation of citizen development was that enterprises would see it as a major asset because LCDP vendors usually present citizen development as a great benefit. However, the results of this thesis show that, although the interviewees were positive about citizen development, they were hesitant as well. All of the interviewees mentioned that citizen development could propose benefits to the enterprise, but only if it was introduced gradually. This hesitation could come forth from the fact that citizen developers do not have IT experience. Despite the accessible nature of an LCDP, it is still seen as a significant enough risk by IT managers to limit citizen development. Furthermore, the fact that the IT department should always be concerned with development also suggests that IT managers are hesitant about citizen developers acting independently.

Chapter 6. Conclusion, limitations, and implications

This chapter presents the final part of the thesis. First, the conclusions are drawn and the research question is answered. Subsequently, limitations to this thesis are discussed and future research directions are suggested. Finally, both the academic and managerial implications are presented.

6.1 Conclusion

Following the results and discussion of this thesis the conclusion was drawn, representing the definitive findings of this study. First, the methods are presented once more and substantiation for the research questions is given.

As the market of LCDPs is growing each year, so is the intensity of its usage by enterprises. For example, by 2025 it is expected that 70% of all newly developed applications are developed through LCDPs. As the popularity is increasing, enterprises aim to use their LCDP more and more, leading them to the enterprise-wide deployment of the platform. However, the development on an LCDP is partly executed by citizen developers, which asks for robust governance arrangements. The aim of this thesis is to identify which arrangements are relevant for the enterprise-wide deployment of an LCDP. To do so, interviews were conducted and analyzed. This has led to the development of a framework, guiding the deployment of an LCDP from an IT governance perspective.

At the foundation of this research, one main research question was formulated, encompassing three sub-questions:

Research question:

How to arrange IT governance in order to facilitate the enterprise-wide deployment of a low-code development platform at a medium to large enterprise?

Sub-questions:

1) *What are the vital guidelines and principles during enterprise-wide deployment of a low-code development platform, to create a safe and robust development environment?*
2) *How does ownership of a software artifact have to be arranged once it is implemented within an enterprise?*
3) *What has to be monitored and evaluated of a low-code development platform within an enterprise to keep it up-to-date?*

The first sub-question revolves around the identification of guidelines and principles an enterprise should consider when deploying an LCDP. When considering guidelines and principles, the first arrangement to make is to classify data as business-critical and non-business-critical information. In addition, application development should never occur independently from the IT department. Finally, the deployment on an LCDP should occur gradually so developers can get used to it.

The second sub-question aims to identify the way software ownership within an LCDP should be arranged. The lead developer of a software artifact should also have ownership of the artifact. However, it is important to also consider the data classification concerned with the artifact. If the owner is not within the IT department and the software artifact uses business-critical data, ownership should be transferred to the IT department. If the owner is not within the IT department but the software artifact does not cover any business critical aspects, collaboration with the IT department is sufficient.

The third sub-question revolves around the monitoring and evaluation of the LCDP. The starting point concerning monitoring is to collect data in order to stay in control of the LCDP. The most important aspects that should always be monitored are the connectors that are used on the LCDP and the capacity of the platform itself. Furthermore, an enterprise should consider its IT capabilities and decide if these are sufficient to conduct monitoring in-house, or if it should be outsourced.

These sub-questions describe the IT governance challenges an enterprise has to take into account when aiming for an enterprise-wide LCDP deployment. Accounting for these challenges proposes a high-level architecture that encompasses several specific design areas an enterprise has to configure to successfully deploy an LCDP. The specific topics that need configuring are the digital development environments, user-assigned role definition, and platform security. Having these design areas in line with the governance arrangements should support the enterprise-wide deployment of an LCDP, where both pro-developers, as well as citizen developers, can collaborate.

6.2 Research implications

This sub-chapter describes the research implications of this thesis. The research implications are divided into theoretical and managerial implications.

Theoretical implications

Previous research identified the usability of LCDPs, its pros and cons, and its techniques. It was found that LCDPs propose significant benefits for an organization as the development time of software artifacts is vastly reduced. To substantiate this decrease in development time, several low-code techniques were identified. This reduction in development time positively influences the workload of the IT department. The workload of the IT department can be further relieved as an LCDP offers the possibility for citizen developers to develop their own software artifacts instead of relying on the IT department. Subsequently, the increased development speed was linked with significant cost savings.

This thesis takes into account the previous literature on LCDPs and extends it toward the domain of IT governance. Previous literature elaborated on what an LCDP is and why it is relevant. This thesis complements existing work by identifying to which arrangements an LCDP should conform in order to properly facilitate citizen development. This resulted in several topics that were abstracted to IT governance challenges and to more specific critical design areas.

Considering what previous studies identified as LCDPs, this study expanded the literature by defining three major IT governance challenges. These challenges are concerned with the enterprise-wide deployment of an LCDP. The first IT governance challenge relates to guidelines and principles that should be in place when executing such a deployment. The second IT governance challenge relates to who has to have software ownership on an LCDP. The third IT governance challenge concerns the monitoring and evaluation of the LCDP in order to stay in control.

In addition to the finding of three main IT governance challenges, three critical design areas were identified in this thesis. These critical design areas propose specific decision domains that need configuration when setting up an LCDP. The first critical design area covers digital development environments. This is a main component within an LCDP and the setup should conform to a predefined structure. The second critical design area revolves around user-assigned roles. This implies the role definition and allocation to the users that engage on the LCDP. The third critical design area is platform security. Security is another foundational concept when deploying an LCDP, aimed at preventing data loss.

Managerial implications

From a managerial viewpoint, this thesis expands on the current services IT Rebels offers to its clients. IT Rebels currently delivers process automation solutions in the form of business applications to its clients. This thesis developed a foundational theory for the development of a new service offering. This service offering would be guiding clients in the enterprise-wide deployment of an LCDP. The definition of the three IT governance challenges helps to guide the client in determining their high-level architecture, in collaboration with IT Rebels. Within this high-level architecture, the more specific critical design areas can be arranged and configured by IT Rebels. Ultimately, the deliverable is an enterprise-wide deployment of an LCDP.

6.3 Limitations and future research

Although the results contribute to the existing literature on the shared domain of IT governance, citizen development, and LCDPs, limitations to this thesis should be mentioned. Mentioning these limitations allows for reflection on the establishment of this thesis and the researcher's potential impact. There are three identified limitations to this thesis:

The first limitation recognizes that the sample size for interviews might not have been optimal. Initially, the plan was to conduct around seven to ten interviews with companies using an LCDP on an enterprise-wide level. However, the number of companies willing to participate turned out to be substantially lower. The reason for their hesitance has not been identified, but it might come forth from a lack of time or seeing their situation as a competitive advantage. Although a deep understanding of the domain was gathered, it can not be ruled out that more data might provide other insights.

The second limitation relates to researcher bias. As grounded theory aims to stay close to theoretical findings researcher bias is addressed. However, the interpretive nature of this thesis might lead to researcher bias. For example, another researcher might interpret the results differently from this study, although this is not assessed. Given this explanation, researcher bias could be a limitation of this thesis. It can not be ruled out, neither can it be proven.

The third limitation refers to the statistical substantiation of the proposed model. As this thesis follows a qualitative and explorative approach, the results can not be statistically substantiated as of right now. This leads to a future research recommendation, aimed at statistically assessing the proposed IT governance measures. Through quantitative survey

research data could be gathered on these arrangements, focusing on a defined sample. Moreover, also empirical data produced by an LCDP could be analyzed over time, in order to assess the effects of these arrangements in practice.

References

Ahmad, M. O., Ahmad, I., Rana, N. P., & Khan, I. S. (2022). An Empirical Investigation on Business Analytics in Software and Systems Development Projects. *Information Systems Frontiers*. https://doi.org/10.1007/s10796-022-10253-w

Al Alamin, M. A., Malakar, S., Uddin, G., Afroz, S., Haider, T. B., & Iqbal, A. (2021). An Empirical Study of Developer Discussions on Low-Code Software Development Challenges. *2021 IEEE/ACM 18th International Conference on Mining Software Repositories (MSR)*. https://doi.org/10.1109/msr52588.2021.00018

Anton, A. (2003). Successful software projects need requirements planning. *IEEE Software*, *20*(3), 44–47. https://doi.org/10.1109/ms.2003.1196319

Aral, S., Brynjolfsson, E., & Wu, L. (2012). Three-Way Complementarities: Performance Pay, Human Resource Analytics, and Information Technology. *Management Science*, *58*(5), 913–931. https://doi.org/10.1287/mnsc.1110.1460

Barricelli, B. R., Cassano, F., Fogli, D., & Piccinno, A. (2019). End-user development, end-user programming and end-user software engineering: A systematic mapping study. *Journal of Systems and Software*, *149*, 101–137. https://doi.org/10.1016/j.jss.2018.11.041

Bird, C., Nagappan, N., Devanbu, P., Gall, H., & Murphy, B. (2009). Does distributed development affect software quality? An empirical case study of Windows Vista. *2009 IEEE 31st International Conference on Software Engineering*. https://doi.org/10.1109/icse.2009.5070550

Bird, C., Nagappan, N., Murphy, B., Gall, H., & Devanbu, P. (2011). Don't touch my code! *Proceedings of the 19th ACM SIGSOFT Symposium and the 13th European Conference on Foundations of Software Engineering - SIGSOFT/FSE '11*. https://doi.org/10.1145/2025113.2025119

Birks, D. F., Fernandez, W., Levina, N., & Nasirin, S. (2013). Grounded theory method in information systems research: its nature, diversity and opportunities. *European Journal of Information Systems*, *22*(1), 1–8. https://doi.org/10.1057/ejis.2012.48

Bock, A. C., & Frank, U. (2021). Low-Code Platform. *Business &Amp; Information Systems Engineering*, *63*(6), 733–740. https://doi.org/10.1007/s12599-021-00726-8

Bowen, P. L., Cheung, M. Y. D., & Rohde, F. H. (2007). Enhancing IT governance practices: A model and case study of an organization's efforts. *International Journal of*

Accounting Information Systems, *8*(3), 191–221. https://doi.org/10.1016/j.accinf.2007.07.002

Case, A. F. (1985). Computer-aided software engineering (CASE). *ACM SIGMIS Database: The DATABASE for Advances in Information Systems*, *17*(1), 35–43. https://doi.org/10.1145/1040694.1040698

Clay Richardson & John R. Rymer. (2014). *New Development Platforms Emerge For Customer-Facing Applications*. Forrester. Retrieved September 26, 2022, from https://www.forrester.com/report/New-Development-Platforms-Emerge-For-CustomerFacing-Applications/RES113411

Costello, T. (2012). RACI - Getting projects "unstuck." *IT Professional*, *14*(2). https://doi.org/10.1109/MITP.2012.41

Denzin, N. K. (2017). *Sociological Methods: A Sourcebook* (1st ed.). Routledge.

Dey, A. K., Hamid, R., Beckmann, C., Li, I., & Hsu, D. (2004). a CAPpella. *Proceedings of the 2004 Conference on Human Factors in Computing Systems - CHI '04*. https://doi.org/10.1145/985692.985697

Eveleens, J., & Verhoef, C. (2010). The rise and fall of the Chaos report figures. *IEEE Software*, *27*(1), 30–36. https://doi.org/10.1109/ms.2009.154

Fontana, A., & Frey, J. H. (2000). The interview: From structured questions to negotiated text. In *N.K. Denzin & Y.S. Lincoln (Eds.), Handbook of qualitative research* (2nd ed.). Thousand Oaks, CA: Sage.

Gartner. (2022, March 3). *Definition of Citizen Developer - Gartner Information Technology Glossary*. Retrieved August 29, 2022, from https://www.gartner.com/en/information-technology/glossary/citizen-developer

Ghiani, G., Manca, M., Paternò, F., & Santoro, C. (2017). Personalization of Context-Dependent Applications Through Trigger-Action Rules. *ACM Transactions on Computer-Human Interaction*, *24*(2), 1–33. https://doi.org/10.1145/3057861

Gregory, R. W., Kaganer, E., Henfridsson, O., & Ruch, T. J. (2018). IT Consumerization and the transformation of IT governance. *MIS Quarterly*, *42*(4), 1225–1253. https://doi.org/10.25300/MISQ/2018/13703

Grembergen, W. van. (2003). *Strategies for Information Technology Governance*. Idea Group Publishing.

IT-Rebels. (2021, December 15). 90North. Retrieved July 19, 2022, from https://90north.nl/our-brands/it-rebels/

Jansen, S. (2020). A focus area maturity model for software ecosystem governance. *Information and Software Technology*, *118*, 106219. https://doi.org/10.1016/j.infsof.2019.106219

Jonathan Lipsitz & Adrienne Capaldo. (2022). The total economic impact of Power Apps. In *Forrester*. Forrester Total Economic Impact. Retrieved October 3, 2022, from https://azure.microsoft.com/en-us/solutions/low-code-application-development/#overview

McLean, E. R. (1979). End Users as Application Developers. *MIS Quarterly*, *3*(4), 37. https://doi.org/10.2307/249047

Microsoft. (n.d.). *Business Application Platform | Microsoft Power Platform*. Retrieved October 5, 2022, from https://powerplatform.microsoft.com/nl-nl/

Moeller, R. R. (2013). *Executive's Guide to IT Governance: Improving Systems Processes with Service Management, COBIT, and ITIL*. Wiley.

Moran, D. (2020, September 17). *Establishing an Environment Strategy for Microsoft Power Platform*. Microsoft Power Apps. Retrieved August 3, 2022, from https://powerapps.microsoft.com/de-ch/blog/establishing-an-environment-strategy-for-microsoft-power-platform/

Ng'ambi, D. (2020). Empathy-driven mobile app development (MAD) without coding. *Critical Mobile Pedagogy*, *1*. https://www.taylorfrancis.com/chapters/edit/10.4324/9780429261572-10/empathy-driven-mobile-app-development-mad-without-coding-dick-ng-ambi

Oltrogge, M., Derr, E., Stransky, C., Acar, Y., Fahl, S., Rossow, C., Pellegrino, G., Bugiel, S., & Backes, M. (2018). The Rise of the Citizen Developer: Assessing the Security Impact of Online App Generators. *2018 IEEE Symposium on Security and Privacy (SP)*. https://doi.org/10.1109/sp.2018.00005

Ridley, G., Young, J., & Carroll, P. (2004). COBIT and its utilization: a framework from the literature. *37th Annual Hawaii International Conference on System Sciences, 2004. Proceedings of The*. https://doi.org/10.1109/hicss.2004.1265566

Ross, J. W. (2003). Center for Information Systems Research - Research Briefings 2002. *SSRN Electronic Journal*. https://doi.org/10.2139/ssrn.407541

Rozendaal, S. (2022, August 30). *The Rebel Approach*. Azure DevOps. Retrieved September 5, 2022, from https://dev.azure.com/IT-Rebels/Operations/_wiki/wikis/Operations.wiki/1/The-Rebel-Approach

Simonsson, M., Johnson, P., & Ekstedt, M. (2010). The Effect of IT Governance Maturity on IT Governance Performance. *Information Systems Management*, *27*(1), 10–24. https://doi.org/10.1080/10580530903455106

Sommerville, I. (2011). *Software Engineering*. Pearson.

Strong, & Volkoff. (2010). Understanding Organization—Enterprise System Fit: A Path to Theorizing the Information Technology Artifact. *MIS Quarterly*, *34*(4), 731. https://doi.org/10.2307/25750703

Thong, J. Y. L. (1999). An Integrated Model of Information Systems Adoption in Small Businesses. *Journal of Management Information Systems*, *15*(4), 187–214. https://doi.org/10.1080/07421222.1999.11518227

Tiwana, A., Konsynski, B., & Venkatraman, N. (2013). Special Issue: Information Technology and Organizational Governance: The IT Governance Cube. *Journal of Management Information Systems*, *30*(3), 7–12. https://doi.org/10.2753/mis0742-1222300301

Wang, E. T., & Chen, J. H. (2006). The influence of governance equilibrium on ERP project success. *Decision Support Systems*, *41*(4), 708–727. https://doi.org/10.1016/j.dss.2004.10.005

Wang, W., Chen, T., Indulska, M., Sadiq, S., & Weber, B. (2022). Business process and rule integration approaches—An empirical analysis of model understanding. *Information Systems*, *104*, 101901. https://doi.org/10.1016/j.is.2021.101901

Waszkowski, R. (2019). Low-code platform for automating business processes in manufacturing. *IFAC-PapersOnLine*, *52*(10), 376–381. https://doi.org/10.1016/j.ifacol.2019.10.060

Weill, P., & Broadbent, M. (2002). Describing and assessing IT governance - The Governance Arrangements Matrix. *MIT Sloan School of Management Working Paper Series*, *2*(3E), 40–42. https://doi.org/10.2139/ssrn.407541

Weill, P., & Ross, J. (2004a). *IT Governance: How Top Performers Manage IT Decision Rights for Superior Results* (1st ed.). Harvard Business Review Press.

Weill, P., & Ross, J. W. (2004b). IT Governance on One Page. *MIT Sloan School of Management Working Paper Series*, *349*. https://doi.org/10.2139/ssrn.664612

Weill, P., & Woodham, R. (2002). Don't Just Lead, Govern: Implementing Effective IT Governance. *MIT Sloan School of Management Working Paper Series*, *326*. https://doi.org/10.2139/ssrn.317319

Whittle, J., Hutchinson, J., & Rouncefield, M. (2014). The State of Practice in Model-Driven Engineering. *IEEE Software*, *31*(3), 79–85. https://doi.org/10.1109/ms.2013.65

Williamson, O. E. (1999). *The Mechanisms of Governance*. Oxford University Press.

Wong, J., Lijima, K., Leow, A., Jain, A., & Vincent, P. (2021, September 20). *Magic Quadrant for Enterprise Low-Code Application Platforms*. Gartner. Retrieved September 5, 2022, from https://www.gartner.com/doc/reprints?id=1-27IIPKYV&ct=210923&st=sb

Wu, S. P. J., Straub, D. W., & Liang, T. P. (2015). How Information Technology Governance Mechanisms and Strategic Alignment Influence Organizational Performance: Insights from a Matched Survey of Business and IT Managers. *MIS Quarterly*, *39*(2), 497–518. https://misq.umn.edu/how-information-technology-governance-mechanisms-and-strategic-alignment-influence-organizational-performance-insights-from-a-matched-survey-of-business-and-it-managers.html

Wuchner, T., & Pretschner, A. (2012). Data Loss Prevention Based on Data-Driven Usage Control. *2012 IEEE 23rd International Symposium on Software Reliability Engineering*. https://doi.org/10.1109/issre.2012.10

Appendix I, Interviewing protocol

Equipment:

1. Voice recorder / Microsoft Teams
2. Pen and paper
3. Location
4. Interview protocol
5. Contact information respondent

Interview protocol:

- Personal introduction
- Introduce research
- Introduce the aim of this interview
- Structure of this interview
    - 16 questions, both open and closed questions
    - 3 subjects, short introduction per subject
    - Time: 1 hour.
- Assure confidentiality
    - Interview is only used for this thesis
    - Interview will be anonymized
    - Ask for interest in receiving transcribed interview afterward
- Ask for permission to (voice) record the interview
- Ask for interest in receiving research findings/results
- Conduct interview
- Thank interviewee for their time and effort

Question type: Open questions, semi-structured

Question order: Per sub-issue, first objective questions, followed by subjective questions and finally how- and why questions.

Q1	What is your job title at %company?
A	
Q2	For how long have you been employed in this title?
A	_ years
Q3	To what extent has %company enterprise-wide adopted the LCDP?
A	
Q4	To what extent do you aim to stimulate citizen / end-user development?
A	
Q5	Does %company provide standards, guidelines, and principles with regard to the participation in / development on the LCDP?
A	YES / NO
Q6	What are crucial standards, guidelines, and principles that %company provides?
A	
Q7	To what extent do these standards, guidelines, and principles facilitate desired usage of the LCDP?
A	
Q8	Does %company use distinctive roles (RBAC) within the LCDP?
A	YES / NO
Q9	Which roles are there?
A	
Q10	What are the technical authorizations coupled with these roles?
A	
Q11	What are the benefits of this role division?
A	
Q12	What are the disadvantages of this role division?
A	
Q13	Does %company monitor the usage of the LCDP?

A	YES / NO
Q14	How does %company monitor this?
A	
Q15	What monitored aspects and subjects are essential to %company?
A	
Q16	How do you use this data for optimizing the LCDP?
A	