# Universiteit Leiden

## ICT in Business and the Public Sector

Quantifying the effectiveness of low-code development platforms in the Dutch public sector

Name: Jelle Sijtstra
Student-no: 1680471

Date: 28/05/2022

1st supervisor: Prof.dr.ir. A.J. Klievink
2nd supervisor: Dr. G.J. Ramackers
3rd supervisor: Ir. M.R. Krouwel

MASTER'S THESIS

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

# Disclaimer

This thesis was supported by an internship at Capgemini. The interpretations, opinions, and findings in this thesis do not represent Capgemini's position on these topics.

Some knowledge in this thesis was acquired with anonymous respondents, especially in the exploratory stages of this research. Several conversations were had via informal talks or digital correspondence based on anonymity. These conversations were held with no more than 5 people with minor significance on the overall research project.

This research is not meant as an objective and complete overview of the pro's and cons of low-code development platforms. This thesis looks at the technologies current performance in the public sector from a differentiating point of view. While this document can help decide if low-code development platforms could be useful for an individual organization or project, it is not advised to base that decision solely on this thesis.

**Abstract**

Low-code development platforms (LCDPs) can be described as software development platforms utilizing prebuilt and prefigured modules to develop applications with minimal or no procedural code via advanced graphical user interfaces. They are designed to allow users who have little developer experience to make business applications or increase the productivity of existing developers, or both. LCDPs are often advertised as a silver bullet, solving many of the great challenges large enterprises deal with today to stay flexible, scalable and competitive. Market analysts suggest that by 2023 over half of all medium or larger companies will have adopted a LCDP as a strategic development platform. These promises and predictions sound great, yet there is a severe lack of research looking into the performance of LCDPs in practice. Currently, researchers use many 'grey' sources to evaluate and analyze this new technology, limiting the accuracy and objectivity of the current scientific literature. The focus of this research will be on the public sector, as preliminary research showed LCDPs may be particularly effective in the context of the sectors unique dynamics.

An exploratory sequential mixed study was performed to assess whether LCDPs contribute to a more effective use of IT in the public sector. For this research, a conceptual framework was developed that links LCDPs advantages to public sector organizational capabilities to public value categories. This framework was iteratively built, checked, and improved via interviews with domain experts. Based on this framework, a survey was developed and conducted among 16 public officials from 14 organizations in the public sector. Based on the outcomes of the guiding questions, the expert interviews, and the survey responses, we can conclude that LCDPs contribute to a more effective use of IT in the Dutch public sector for the sample researched. This is achieved through a combination of advantages of LCDPs over traditional programming methods which improve the organizational capabilities important for digitalization. These may in turn lead to improvements in duty oriented and service oriented public values, but the evidence on this from this sample is inadequate. The research outcomes can help public organizations better understand which of the potential advantages of LCDPs are actually realized in practice within the context of public sector organizations. The conceptual framework also provides a basis for future research into the application of LCDPs.

# Acknowledgments

This thesis was made under the great supervision of Prof.dr.ir. A.J. Klievink and Dr. G.J. Ramackers who I would like to thank for their invaluable guidance throughout this thesis. This thesis was supported by an internship at Capgemini which I am very thankful for. I would like to thank the experts at Capgemini who were willing to share their knowledge and experience in the interviews: Prof. dr. Martin Op 't Land, Maurice Driessen, Wiger Levering, Jelle Roo, Max Hikke, and Mariska Dessaur-Terwel. I would further like to thank the 16 participants of the survey. Their openness and passionate comments provided an additional level of insight that I would not have dared to expect and helped this study greatly.

I would also like to thank Ilse Sytstra-Boek, Dr. A.R. Ingrams, Prof.dr. J. Cabot, Hanneke van der Horst, Erik Hoorweg, Kees van den Broek, Dr. L. Zavolokina, Dirk van Egmond, and my other Capgemini colleagues who have advised this project. A final thanks goes out to Capgemini supervisor Marien Krouwel for his support and seemingly infinite wisdom in the low-code development domain.

# Contents

# 1 Introduction

## 1.1 Background

Ever-changing market conditions and a constantly changing IT landscape call for fast and cheap ways to meet software demands. Furthermore, recruiting developers has become increasingly difficult and more expensive [31]. In order to tackle these problems, low-code development platforms (LCDPs) have emerged. These platforms are designed with the idea to limit recurring traditional hand-coding and programming [83]. LCDPs can be described as ecosystems, which use prebuilt and prefigured modules to develop applications. The platforms use more visual interfaces to allow users who have little to no developer experience to make business applications and to increase the productivity of existing developers.

The term low-code was introduced in a Forrester report in 2014, stating that enterprises like the fast, continuous, and test-and-learn delivery it enables [74]. A selection of the benefits of low-code software development that are stated in whitepapers and on LCDPs websites are privacy, rapidity, cost reduction, complexity reduction, easy maintenance, increased responsiveness to business, escape legacy debt, and enabling citizen developers (developers that have no or little software engineering background) [78].

Many bold claims are made about the future of LCDPs. In a 2019 report, Gartner predicted that LCDPs will be responsible for 65% of all app development activity in 2024 [88]. A year later, they predicted that by 2023 over half of all medium or larger companies will have adopted an LCDPs as a strategic development platform [90]. Although these claims seem optimistic, what can be said is that the adoption of LCDPs is growing fast.

Some research suggests that LCDPs stem from several technologies that have been in development for years: Model-driven development (MDD), rapid application development (RAD), code generation, and visual programming [93]. Another describes low-code as an attempt to combine visual development techniques (models) and code generation to reduce manual coding [16].

Therefore, in order to understand low-code development better, it seems wise to also look into the research of model-driven development. One researcher even states that low-code can be seen as a synonym of model-driven development [16]. His paper continues that LCDPs are a fixed MDD-language solution, which only targets a restricted type of software applications: data-intensive web and mobile applications. This interesting view on the low-code trend raises the question of why the term LCDP is trending and not the overarching MDD term.

According to Cabot, this could be due to a rebranding effort by the platform vendors [16]. Low-code seems to be a less ambiguous, more familiar, and clearer term. LCDPs do not use any techniques new to the MDD-community, but manages to sell these techniques in a way better perceived by practitioners. This in turn might lead to more successful projects, as the success of an MDD-approach was found to be more dependent on social and managerial factors than technical aspects [39]. Additional support for the

rebranding theory was found in a recently released Gartner research paper on LCDP: "its (ServiceNow) ability to establish an identity as an LCDP provider is driving increased adoption among its existing customers" [89].

Whether this theory is true or not, it does create confusion and complicates finding resources on this topic online. An example of this can be found on the website of Mendix, one of the largest LCDPs. On their website, they refer to themselves as a model-driven development platform [52], a high productivity app platform [58], a multi experience development platform [59], a rapid application development platform [56], an application platform as a service (aPaaS) [57], a no code platform [54], and on their main page a low code development platform [53].

Although some of these terms have slightly different meanings, they are very minimal in the bigger picture and apply to almost all platforms. Therefore, in order to keep this thesis clear and consistent, it will only use the term low-code development platforms to refer to these platforms. Low-code development platforms are defined as software development platforms provided on the cloud through a platform-as-a-service (PaaS) model, and enable the development and deployment of fully functional software applications utilizing advanced graphical user interfaces and visual abstractions requiring minimal or no procedural code [77].

## 1.2 Problem statement

In 2014, the committee Elias on IT-projects in the Dutch public sector concluded that the government loses one to five billion euros yearly [81]. This report lead to several changes, one of them being the new Bureau of IT Assessment (BIT). However, in 2016 a new research showed only a third of IT projects in the Dutch public sector is completed successfully [13]. In a 2019 study by the Standish Group, 110 IT projects in the Dutch public sector were analyzed for success factors. The success rate (time, budget, and target) of these projects averaged 43%, with large and grand projects succeeding 10% of the time [32]. According to research from Oxford University, IT projects in the public sector which required some form of business transformation, were six times more likely to exceed budget and 20 % more likely to exceed schedules than corresponding projects in the private sector [8].

The large difference in project success between the private and public sector could be explained by several dynamics that only exist in the public sector. For instance, the public sector has to deal with additional management issues, a wider range of organizational mandates and constituencies, and an additional difficulty of having to be flexible during political administration changes [24]. Other differences described in literature are the lack of a competitive environment and no profit-seeking motive [71]. While both the private and public sector need more IT personnel, the public sector has a harder time getting those vacancies filled [23].

These differing dynamics of the public sector could potentially play to the strengths of LCDPs. As described in the introduction, LCDPs promise: privacy, rapidity, cost

reduction, complexity reduction, easy maintenance, increased responsiveness to business, escaping legacy debt, and enabling citizen developers [78]. These advantages could for example reduce the demand for more IT personnel by either decreasing IT workload or enabling citizen developers to help out. The decreased complexity, easy maintenance, and escape of legacy debt could lead to a more flexible IT landscape allowing to better deal with political administration changes. These examples show that some of the marketed advantages of LCDP could help deal with some of the current IT related challenges that the public sector is facing. This makes the public sector an interesting research focus on evaluating whether these advantages are realized in practice.

When looking for previous work on the effectiveness of LCDP in the public sector however, there is little to be found. In an exploratory study for this research paper, only two papers were found looking into LCDPs at public sector organizations. One paper was found to look into the potential of LCDPs in the healthcare sector of Norway [65]. Another was found to look into its potential for higher education institutes of Norway [87]. These papers however did not go into much detail and their research could not be generalized to say something about the entire public sector. Instead, as some consider low-code a synonym of MDD, there was a search performed on that term and related terms (Model-driven engineering, rad, visual programming, code generation). This unfortunately also did not yield many usable results. One meta paper was found discussing 58 MDD case studies, however, not a single case covered an organization in the public sector [85].

Currently, only one large-scale survey has been found to look into providing evidence for the previously stated benefits of LCDPs in both the private and public sector. This survey was performed by Outsystems, an LCDP vendor, in 2019. For the survey they asked 3300 IT professionals across the world for their reasons for using or wanting to use LCDPs [69]. In it, they feature a split in their results for both government and education. This could make it usable as a basis for this research project. However, the survey has severe limitations. In addition to questionable objectivity, of all respondents, 59% have no actual experience with these platforms, but are just interested. Furthermore, these benefits were mainly obtained from IT professionals working in the private sector (90%), leaving the public sector important for this research underrepresented. Moreover, the survey was answered only by IT professionals, leaving out valuable opinions from the business or organizational side of enterprises. Therefore, it was decided that the survey is not usable for this research.

The lack of research on LCDPs in the public sector is disappointing. As the term low code was coined in 2014, one would expect that a technology sometimes presented as a silver bullet (an order of magnitude improvement in productivity, reliability, and simplicity [14]) would have more scientific literature for the sector that perhaps needs it the most. Therefore, the purpose of this research is to determine the effectiveness of LCDPs in the public sector. This should help public organizations better understand what claimed value of LCDPs is being realized in practice. It will also provide a basis for future research to look deeper into this research area.

In summary, there is a lack of research surrounding LCDPs. Most importantly, at the time of writing, no research looking into the effectiveness of these platforms was found. Currently, researchers use many 'grey' sources to describe this new technology, limiting the accuracy and objectivity of the current scientific literature. To help improve future research, an evaluation of the organizations that currently use LCDPs is needed. To scope this evaluation, the public sector is first researched, as preliminary research suggests that LCDPs could stand out in managing the sectors' unique dynamics.

## 1.3 Research questions

The research question is: *Do LCDPs contribute to a more effective use of IT in the public sector?*

Two guiding questions will help answer this research question:

**GQ1** How can the effectiveness of LCDPs in the public sector best be quantified?
**GQ2** How effective are LCDPs currently in the public sector?

For the guiding questions, it is important to briefly talk about effectiveness. The term effectiveness was chosen here instead of value creation or adding value. Although very similar and sometimes used interchangeably for effectiveness, these latter terms often have more narrow definitions that do not fit the goal of this research. Adding value is often defined as bringing small steady advancements of value already existing and creating value is often defined as a new product or process that is aimed at filling a need [22]. For this research, both of these are combined in effectiveness. Public sector effectiveness can be seen as the public administrations reason for being; its existence stems from a legal obligation to serve and advance the common good [61]. One research area focusing on this end-goal for the public sector combines this in public value categories.

Some theoretical models found for IT-value mapping are the production function model and resource-based view. However, these were made with the private sector in mind and do not recognize the quirky dynamics of the public sector [71]. Thus, it is difficult to directly measure the value that IT can bring in the public sector. A sidestep proposed by literature is by first looking at how IT investments change organizational performance [71]. This means looking at what mechanisms in the organization translate the IT investments into value. This idea, among with LCDPs, public value, and e-government research, will be used to develop a conceptual framework summarizing the value LCDPs can bring in the public sector organizations for the first guiding question. The conceptual framework will be built iteratively and checked by domain experts after each cycle. The iterative approach is chosen as the total research time is limited and the goal is to send the survey before the summer holidays to get a higher response rate.

To answer the second question, a survey will be developed and performed within the Dutch public sector. The survey will be developed using the build conceptual framework. In order to properly answer GQ2, a proposed sample size of 80 is preferred. The plan is to reach this objective using the network of Capgemini and its clients. Capgemini currently uses a couple of LCDPs which are popular in the Netherlands and are considered part of the top LCDPs by Gartner and Forrester [75, 90]. Their vision and ability to execute is, according to Gartner, rated the best of the platforms out there. Researching this small selection of the over 104 MDD platforms out there [28] should therefore hopefully yield results that can represent the entire LCDP market.

## 1.4 Scientific and societal contributions

The scientific contributions of this thesis are twofold. The main contribution is the first scientific deep dive into the effectiveness of LCDPs. As described in the introduction, there is a severe lack of scientific research in this area. Most research on LCDPs seems to copy the advantages and disadvantages from whitepapers and marketing documents due to a lack of better alternatives. Initially, this thesis is no different. However, with the domain expert reviews and survey outcomes this study works towards a better understanding of the achieved effectiveness of this technology. The research is scoped on the Dutch public sector. As this sector has its own unique problems as described in the problem statement, this is a major addition to this thesis' scientific contribution.

The second scientific contribution is the used research method, which tries to measure an individual IT technology effectiveness through organizational capabilities and public value contributions. This method is based on the theoretical framework of Pang et al., which is well suited for this research as it focuses on the unique dynamics found in the public sector [71]. In this thesis, that framework is transformed and made measurable for this specific technology for individual organizations. This method can be reused or adjusted to allow the measurement of other IT technologies in the public sector.

Besides the two scientific contributions, this thesis could also help organizations active in the public sector to better understand what claimed value provided by LCDPs is being realized in practice. This could help current and new projects in the public sector for both the demand and supply side. The developed conceptual framework can be used as a basis for future research and can inspire practitioners working with LCDPs in the public sector.

## 1.5 Structure

The thesis consists of 7 chapters. In the following chapter the theoretical foundation of this thesis is discussed and relevant concepts and studies are analyzed. The third chapter dives deeper into the main LCDPs which are subject of study. Chapter four describes the methodology and research approach in detail. Next, chapter five and sex will go over the results of the interviews and survey respectively. Chapter seven presents the conclusions to the guiding questions and the overarching research question. In chapter eight the results are discussed and the limitations of the study are outlined. The final chapter will discuss possible questions for future research.

# 2 Theoretical foundation

In this chapter, academic literature will be analyzed to work towards an initial conceptual framework that can measure LCDP effectiveness in the public sector. It will also highlight or discuss various related topics. First, the concepts of model driven development and abstraction in application development are explained. These can help non IT people understand what makes the approach of LCDPs to development special. Next, several topics will be discussed that work towards measuring the impact of technology changes in any organization. After that, several studies are discussed that focus on the public sector, most going towards the public value research area. This is followed by a section going over found advantages and disadvantages of LCDPs in both academic and grey literature. Additionally, the differences between high-code, low-code, and no-code are briefly discussed and visualized for the reader. At the end of this chapter is the initial conceptual framework for guiding question 1 which was based on found literature.

## 2.1 Abstraction and development

In the introduction, it was described how there is a severe lack of software developers in the Netherlands. It also described how LCDPs could help with this problem by limiting traditional hand-coding as much as possible. To see where this connection comes from for non IT readers, it can be interesting to look at abstraction levels in software development. Abstraction in this context can be seen as "a cognitive means by which engineers, mathematicians and others deal with complexity" [47]. They do this by removing unimportant details as much as possible and by spotting generalizable features. These abilities are key to becoming a skilled software engineer.

This idea of abstraction can also be applied to the development tools themselves. For example, low-level programming languages have more abstraction than machine code, which just consists of zeros and ones. On the other side of the development tools spectrum lie modeling languages. Instead of using textual code it uses visual models, offering an even higher level of abstraction. These models are then transformed into an application by code generation or model interpretation [12]. This increase in level of abstraction for programming languages is demonstrated in Figure 1.

The model-driven form of development can therefore be defined as "a software engineering approach consisting of the application of models and model technologies to raise the level of abstraction at which developers create and evolve software with the goal of both simplifying (making easier) and formalizing (standardizing, so that automation is possible) the various activities and tasks that comprise the software life cycle" [35]. This can allow existing programmers to develop faster, but can also make application development more accessible for those that did not study IT. In its most basic form, this is what makes LCDPs special.

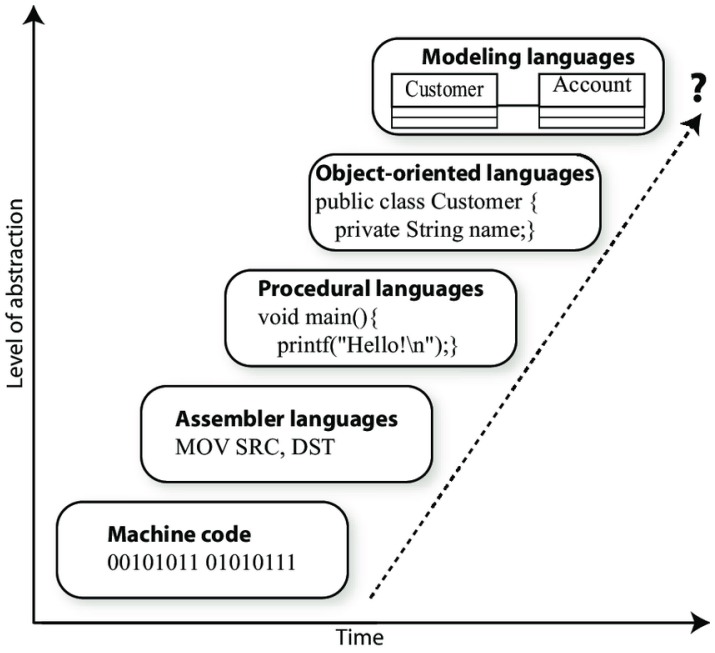



Figure 1: Abstraction levels in programming paradigms [72]

## 2.2 Measuring IT impact

Measuring the impact of technology changes in an organization is a complicated task for multiple reasons [10]. First of all, it can be a challenge to put a boundary on where the impact of the technology ends. Furthermore, the areas impacted are usually numerous and these impacts are hard to distinguish from other organizational changes. To help with these challenges, the technology transfer research area offers ways to work towards a measurement of technology effectiveness.

One of the main problems highlighted in the paper of Bozeman is demarcating the technology object from its environment [10]. Defining which specific characteristics of an IT technology separates it from others can therefore be a difficult task. For some highly standardized applications like e.g., Microsoft Word or PowerPoint, which are usually delivered in a standard socio-technical package, demarcation might not be an important issue. However, for LCDPs which vary significantly between vendors, demarcation becomes a problem. This is emphasized by the existence of social aspects vital to the success of LCDPs. One researcher found that the success of an MDD-approach was found to be more dependent on social and managerial factors than technical aspects [39]. This means that if an LCDP implementation fails in an organization, it is difficult to distinguish if the failure is caused by the technology transfer or because the technology has not worked out in that particular situation.

Bozeman et al. revised their Contingent Effectiveness Model of Technology Transfer (Figure 2) in 2015, with the most important change being the addition of the public value criterion [11]. They note that in the success stories of technology transfer for US

agencies often the social impacts are highlighted. Therefore, they argue that public value is an important criterion for evaluating technology transfer activities. Furthermore, this added criterion makes room for more citizen-centric metrics, counteracting the usual emphasis on enterprise or economic impacts. Ultimately, few citizens care about these latter impacts, caring more about better health, safety, and other values covered in the public value theory. Another important argument for adding public value is that in the end, these projects are funded by tax money. Therefore, it seems fitting to include public value as it features broader values, encompassing more or all citizens.

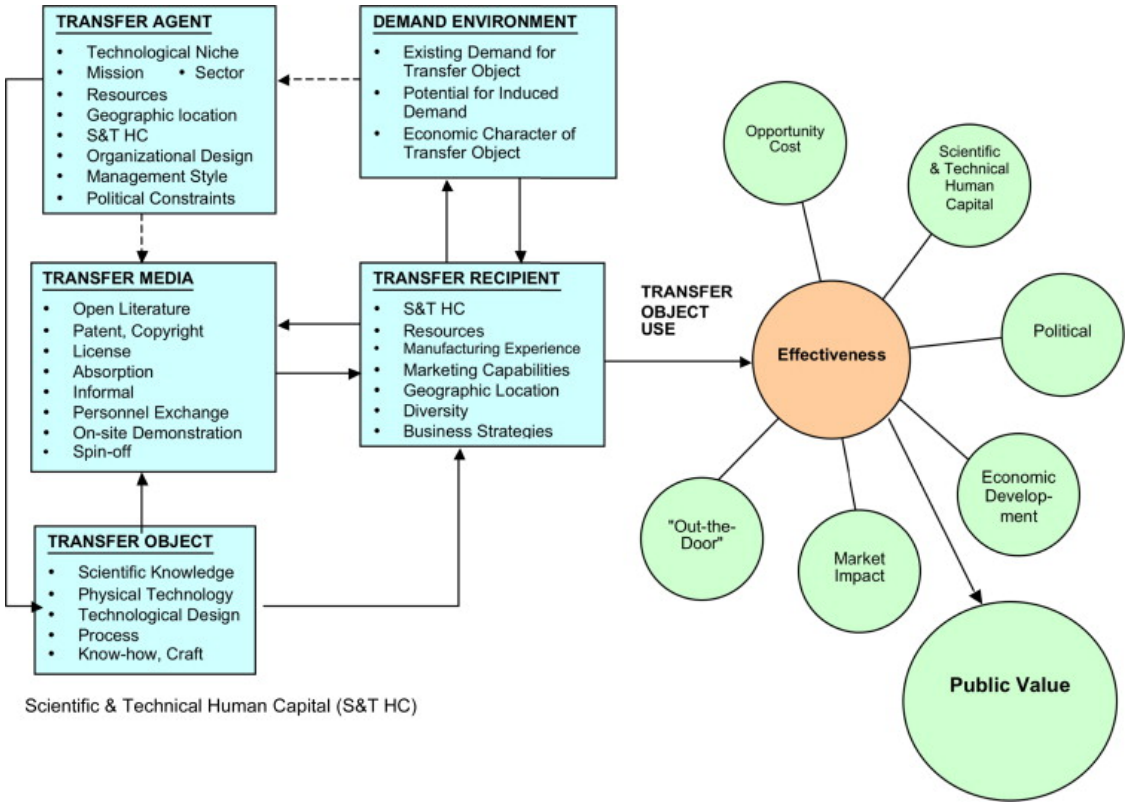



Figure 2: Revised contingent effectiveness model of technology transfer [11].

The potential of this public value theory is that it can "shift the focus of public sector management from internal efficiency to value creation processes that occur outside the organization" [70]. It is perhaps one of the most important concepts surrounding the electronic government (e-government). Public value can be defined as "citizens' collective expectations in respect to government and public services" [44]. A citizen in this definition can be defined as someone in their different stakeholder roles, such as tax-payer, user of public services, policymaker or public servant [17]. Public value is therefore not a simple distinction between private and public, but rather a view on the relationship between citizens and the public sector or society as a whole [60]. It is a citizen-centric metric encompassing broader values covering more or all citizens.

This makes public value an interesting tool for both guiding questions. In a search for

previous work using public value to evaluate technology changes in the public sector few were found. Although many papers use public value theory to assess e-government in its whole, only a couple were found to have used the theory to assess an individual information technology. One research examined the adoption of eHealth applications in the European Economic Area against the dimensions of public value [33]. This paper took an interesting approach, but was not deemed feasible for the limited time span of this study. Furthermore, the chosen approach seems hard to utilize for internally focused applications. The other available paper is the development of an assessment framework to study the effects of the application of AI technologies in governments [86]. As this framework is not tested in practice or made measurable it is of limited use for this research.

One of the reasons for the lack of research papers using this theory in a practical way might be the lack of measurement options presented by current literature [29]. In this paper there is even a warning for theoretical stagnation, as the unanswered calls for public value measurement options date back from 2010. This lack of options results in a challenge for research projects like this study that attempt to measure effects on public values. A reason for this literature gap could be that it is difficult to find values that can be used across all the different kinds of public organizations [80]. That being said, several researchers have attempted to create a standardized set of values that could be used to compare and benchmark public sector organizations [11, 80, 42]. The downside of using a generalized set of public values however, is that they can't take into account all the subtle differences in value creation. For instance, a citizen could value certain public value measures differently depending on events happening at that time. One idea is to therefore combine a general set of public values with a few context-specific public value measures [29].

## 2.3 Process view

Instead of looking just at public value, it is also possible to look at the processes that ultimately lead up to the creation of public value. This can be hard when you need to look at all different kind of processes involved. However, for this research project many can be discarded as it focuses only on the digital aspect of the public sector. One model that does just this is the one developed by Pang et al. [71]. In their paper they try to answer the questions how and what value is created in the public sector by IT.

They note that the positive relationship between IT investments and performance in the private sector is well established. When looking at the public sector however, this relationship is proportionally way less established. They theorize that the lack of a theoretical model for IT value in the public sector is to blame. The popular production function model [15], resource-based view [92], or dynamic capabilities [26] are well-suited for business applications. However, business-minded models focus on creating market share, profit, and competitive advantages over competitors and have no focus on the unique dynamics found in the public sector. Instead, Pang et al. work towards creating their own model that tries to take these dynamics into account. They suggest to use

organizational capabilities as an additional component to explain what and how IT value lead to organizational performance. They propose three views on the relationships between these values. First is the complementary view, in which IT resources and organizational capabilities complement each other to create organizational performance. The second view is the process view, in which IT resources contribute to organizational capabilities which in turn contributes to organizational performance. The third view is the configuration view, that suggests that organizational performance is caused by a complex dynamic mutual interdependence among all variables.

The previously discussed process view was applied by Pang et al. to map information technology resources to organizational capabilities seen in Figure 3. This approach distinguishes itself by taking an interdisciplinary approach that looks at both the information systems and the public administration literature. In their model they identify five organizational capabilities that are vital to creating public-value [71]:

- Public service delivery capability: The ability to deliver the maximum possible outcome of public services with as limited resources given by the public as possible.
- Public Engagement capability: The ability to let a broad range of stakeholders participate in every step of policy formulation and implementation.
- Co-production capability: The ability to marshall all necessary resources from participating partners, to align their competing interests, and to coordinate their efforts and activities toward advancement in public value.
- Resource acquisition capability: The ability to garner resources necessary from resource providers for their initiatives. Public managers play a key role in policy exploration and formulation.
- Public-sector innovation capability: The ability to be vigilant, to be understanding of changing circumstances, and to sense emerging needs and aspirations of various stakeholders.

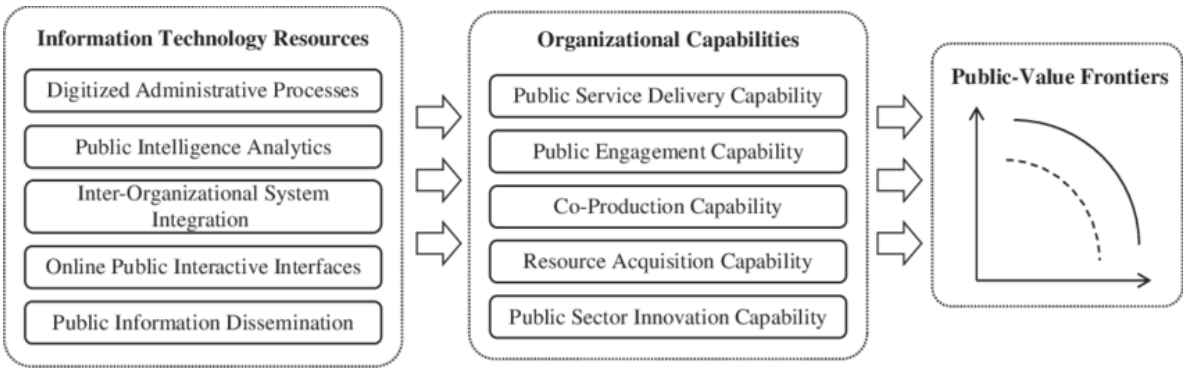



Figure 3: Theoretical framework for IT Value in the public sector [71]

The framework by Pang et al. uses public-value frontiers for their organizational performance. This follows the idea that the public sector aims on reducing conflicts among

competing public values [71]. However, this gives little guidance for measuring organizational performance in practice. As measuring current value brought to the Dutch public sector by an IT technology is required for answering the second guiding question, this could use an alternative approach. Instead of using public value frontiers, it is possible to use the more easily measured public value taxonomy proposed by Twizeyimana and Andersson's [84]. The definitions are derived from their original research paper.

- Improved public services refer to different service improvements like access and delivery of the services offered by e-government.
- Improved administrative efficiency refers to a range of efficiency, stability, and responsiveness metrics.
- Open government capabilities refer to the capabilities achieved through the achievement of democracy dimensions such as openness, transparency, and collaboration.
- Improved ethical behavior and professionalism refer to the foundational values of government operations and policies like responsibility, integrity, and honesty.
- Improved trust and confidence in government refers to the trust of citizens in the way they manage the economy, public resources and the way they handle private information.
- Improved social values and well-being refer to values relating to family, community, or other relationships.

Although the definitions shown might be difficult to understand for a practitioner, their paper presents several pages of key performance indicators (KPI's) which are understandable and measurable [84]. Unfortunately, this taxonomy and its KPI's were found to have some limitations when it was used in this study. The category names were found to not fully convey all items that are in their categories. As an example, the public value category of improved administrative efficiency contains the items greater fairness, honesty, and equality. As this created confusion during the interviews a different taxonomy was also looked at.

The taxonomy of Bannister & Connolly has a slightly different approach [5]. It splits the public value into duty, service, and socially oriented values. These three groups are easier to understand and convey their items better. The following definitions were derived using their research and the original definitions from Hood's paper [36]:

- The duty oriented public values are about properly matching resources to defined tasks, while also looking at the nonfinancial responsibilities of civil servants to the government and state.
- The service oriented public values are about providing a high level of service to citizens the same way a company does: efficient and reliable, while being adaptable and resilient.
- The socially oriented public values are about the aim of an honest and fair public

sector with broad social goals.

The created definitions for the public value areas might still be difficult to understand for practitioners, but the items in these categories are clear and consistent. The full taxonomy can be seen in table 2.3.

| **Duty orientated** | **Service oriented** | **Socially oriented** |
|---|---|---|
| Responsibility to the citizen | Service to the citizen in his or her different roles | Inclusiveness |
| Responsibility to the elected politicians of the day | Respect for the individual | Justice |
| Proper use of public funds | Responsiveness | Fairness |
| Compliance with the law | Effectiveness | Equality of treatment and access |
| Efficient use of public funds | Efficiency | Respect for the citizen |
| Integrity and honesty | Transparency | Due process |
| Facilitating the democratic will | | Protecting citizen privacy |
| Accountability to government | | Protecting citizens from exploitation |
| Economy/parsimony | | Protecting citizen security |
| Rectitude | | Accountability to the public |
| | | Consulting the citizen |
| | | Impartiality |

Table 1: Taxonomy of public value categories [5]

## 2.4 Low-code development platforms

Besides the theory surrounding the measuring of technology changes in the public sector, it is also important to discuss LCDPs themselves. Ultimately, the guiding questions and research question require measurable performance metrics for LCDPs. Therefore, this section will talk about the found documents on LCDPs themselves. One of the most important elements for this research are the mentioned effects on an organization (positive or negative). Although there is a lack of scientific papers on LCDPs, there are several white papers which highlight the main benefits. Especially papers from the market research companies of Gartner and Forrester. Besides these sources, several research papers from LCDPs vendors themselves will be taken into consideration. These less than ideal sources resulted in significantly more advantages than disadvantages. A search was performed on the keywords: “low-code *” OR “no-code *” OR “model-driven *” OR “IT value” Or “citizen development” OR “aPaaS” in the period 2019 - march 2021, written or presented in English in Google scholar. Of this search, a portion of literature was found on LCDPs.

The goal is not to perform a comprehensive analysis of which paper said what and why, but the goal is rather to find relevant concepts which can be discussed during the interviews later. More on this research approach can be found in Chapter 4. Some criteria where applied during the search process. Since the LCDPs develop quickly, the search was limited to the past 2 years. As some of the documents are commercial in nature and some of the academic papers sourced these same documents, the listed advantages should not be taken at face value. Below is the list of all advantages and disadvantages found in this literature. Those closely related are grouped together.

- Less hand-coding [69, 73], faster (5-10x) development [43], reduction of development time [37, 41, 21], 50-70% faster application development life-cycle [91].
- Easy and rapid deployment due to cloud deployment with little effort [73, 64].
- Cost reduction, as less development, less maintenance, and easier deployment can turn into a reduction in starting and long-term costs. [74, 37, 43].
- Complexity reduction, as development is simplified by (re)using prebuilt components [78, 43].
- Easier maintenance, as there is less high-code [69]. Maintenance can be performed by citizen developers, but only if the system's customizations are well designed and structured [28]. However, with an increased project size it can become too complex for citizen developers [91].
- Inclusion of business profiles, as the simpler and more intuitive way of development with LCDPs allows business users to be directly involved in the development of apps [93]. Participation of nontechnical employees (citizen developers) at an earlier stage in the project [91].
- Fast prototyping [37, 21]. A fast minimum viable product ensures quick validation of ideas and customer requirements before committing too many resources [73]. Minimization of unstable or inconsistent requirements as fast prototyping allows the team to quickly find conflicting requirements while their impact is still limited [78].
- Accelerate digital transformation, as LCDPs give new tools to facilitate and automate new application development to promote transformation [69, 78].
- Increased responsiveness to the business and enabling citizen developers to improve internal processes [69]. The developers of LCDPs require a different skillset as they need to be closer to the business and more communicative [37]. The development team can skip technical details and focus on the business rules implementation [64].
- Reduce dependency on hard-to-hire technical skills [69, 91]. LCDPs are easier to learn than traditional hand-coded solutions [21, 73].
- Escape legacy debt, by replacing legacy systems or building layers over legacy

systems organizations can 'escape' the inflexibility caused by older IT systems [69].

- Protect against technology churn, as LCDP take care of certain aspects of application development and deployment which are prone to getting outdated [69].
- Portability of the developed application, as most LCDP can deploy applications as both mobile and desktop versions or as a web application which is usable on both [43].
- Interoperability, as most LCDP allow for many connectors/API's to be made increasing the ease at which information can be exchanged between information systems [69, 85].
- Integrated agile way of working, some LCDP vendors have integrated agile methodologies in their platform [48]. This could result into more communication between stakeholders, faster development cycles, and more developed applications using the LCDP.
- Easy life-cycle management, due to deployment, monitoring, and management functions being built into most LCDP [73].
- No unit-tests, as smaller applications can go straight to production [91].
- Process automation, as small automation features can be easily created, e.g., email confirmation, record creation/updating (might be Salesforce specific) [91].
- Scalability, as most LCDPs are run in the cloud ready to allocate more resources to specific applications if demanded [37, 73].
- Continuous integration, as most LCDPs are run in the cloud they can relieve developers of most back-end work needed to get applications running after new updates [37].
- Lower fixed costs, as most LCDPs offer usage-based subscriptions the initial costs costs for starting an application are lower, especially for smaller organizations [73].

Next is a list of the mentioned limitations of LCDPs using these same sources. These limitations are prone to get outdated with newer versions of LCDPs and might need more validation.

- Scalability, as current platforms are not ready for large-scale projects or mission-critical enterprise applications [83, 69, 28]. Some argue otherwise [37, 73].
- Fragmentation, as LCDPs can become an isolated part of the enterprise portfolio adding cost and complexity [83, 73]. Each LCDP presents their own programming language, resulting in little cross LCDP cooperation and difficulties in reusing pre low-code programs.
- Vendor lock-in, as it is easy to become dependent on LCDP and there is no easy way to transition out of it without high costs [69, 83]. Once locked-in, there is little

bargaining power for a customer to avoid price increases. Some LCDPs produce generated code or other ways to export the models to give customers some tools against vendor lock-in [75].

- Issues in real-time data processing, as LCDP need to be able to handle all sorts of situations they are generally inefficient and therefore struggle to handle real-time data processing needed for e.g. IoT devices [73, 78].
- Limiting creativity and flexibility, as solutions are limited by the capabilities of the selected LCDPs and its domain-specific language [28, 69, 21].
- Lack of a general testing framework, resulting in limited testing capabilities as some LCDPs are reliant on third-party testing tools with limited automation [43].
- Testing can be too difficult for citizen developers. This is a great limitation of the potential as citizen developers are the system functionality and requirement expert [43].
- Lack of evidence of citizen developers in action found in research [73].

During the research a new paper was released which examined knowledge integration using LCDPs. Knowledge integration is defined as applying and synthesizing specialized knowledge residing in disparate parts of an organization [40]. In their research, 10 expert interviews were performed with an LCDP vendor in Switzerland. In addition they analyzed 201 product reviews and archival data from 56 Gartner and Forrester analysts.

In their research they describe five key characteristics of LCDPs which are linked to the knowledge integration mechanisms they support between business and IT [40]. These are:

- Centralized structure; LCDPs provide a centralized platform which IT owns and manages, but provides citizen developers with the tools necessary for application development.
- Reusable development components; pre-programmed building blocks which can be reused across applications.
- Ease-of-mastery; the lower level of technical complexity allows employees to more easily master the platform by users who have little technical know-how.
- Visual interface; the visual interface supports group problem solving and decision-making between business and IT specialists.
- Real-time editing; functionalities can be added and removed easily with the end-result being directly visible, supporting group problem solving and decision-making.

## 2.5 Initial conceptual framework

This concludes the literary section on how to measure the effectiveness of LCDPs in the public sector. For this study, the decision was made to continue with a combination of Pang et. al' framework and public values following the taxonomy of Twizeyimana and Andersson's [71, 84]. These, along with the list of advantages and disadvantages were used to get to a first concept of the framework. More on the methodology to get to this initial concept framework can be found in Section 4.3.1.

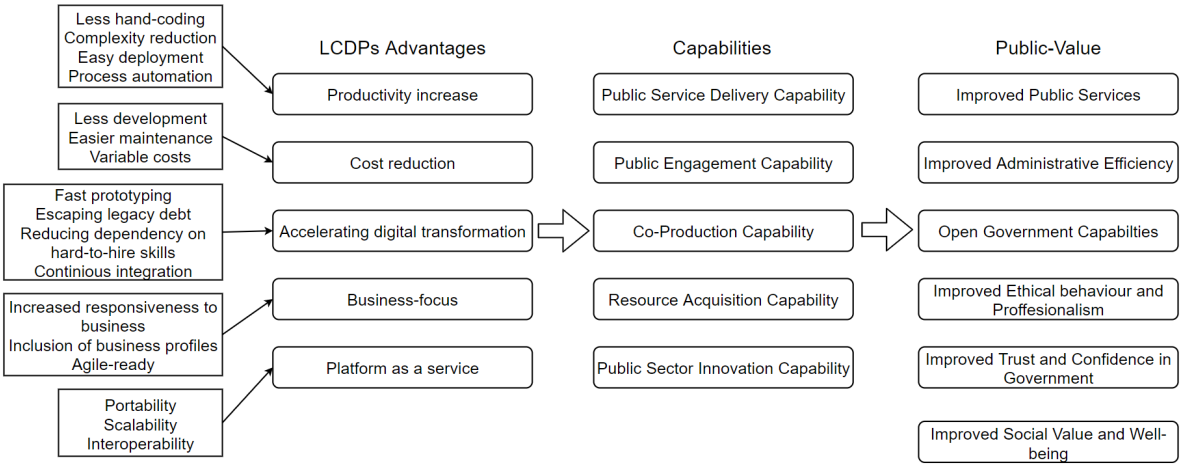



Figure 4: Initial conceptual framework from literature

# 3 Low-code development platforms

Before moving to the methodology in Chapter 4, it is important to briefly talk about the domain of LCDPs. This will aid the reader in their understanding of the LCDP market in its current state. Due to the lack of specific research documents on the Dutch market, it will go over the international market. There are three topics which will be discussed. First, the popularity, maturity, and growth of some of the main platforms is discussed. This is important to look at, since some say that it is likely that only a few winners will remain [82]. Their functional focuses and/or vertical approaches are vital to their differentiating power. Therefore, an attempt to identify the possible 'winners' will be made. Each of them will be briefly introduced with how they differentiate themselves and in what markets they are active. Finally, one interesting newcomer in the Dutch LCDPs is discussed to show how the market might be changing over the coming years.

As introduced in the research approach, there are over a hundred platforms that could qualify as a LCDP. Therefore, to get an overview of the 'winning' platforms used in the market, the 'leaders' of the Gartner research on LCDP will be used. This leaves the following platforms that will be briefly introduced in this chapter: Mendix, Outsystems, Microsoft Power Apps, SalesForce, Appian, Servicenow, Betty Blocks, and Appsemble. This section is based on the following research documents: [90, 77, 28, 2]. As no research documents feature the recently founded startup Appsemble, an interview was performed with the founder. Due to the speed of development of LCDPs and the age of most sources, this is not a comprehensive comparison of the ins and outs of every platform. It is presented as a means to aid the readers understanding of the current market.


Figure 5: Gartner Magic Quadrant for Enterprise low-code application platforms [90]

In order to determine the maturity and popularity of these platforms, five measures were gathered from several sources. These can be seen in in Table 2. The Google Trends indicator was made by taking the mean for the last 12 months from 14 October 2020 until 14 October 2021. The search query used was: 'product name' + 'low code'. As Google Trends only allows comparison and normalization of up to five search queries the selection was split into two groups. Each group had two differing queries and three queries that were the same. This way a comparison could be made between all search queries. The final number is between 0 and 100, which represents the relative search popularity compared to total searches [34].

Employees, Employee growth 1y, and Job openings were gathered using LinkedIn's premium insight feature where possible on company profiles on 14 October 2021. The job openings from Betty Blocks were gathered from their website. The subscription revenue was gathered from year reports or product blogs where possible. The subscription revenue figure is as close of an estimation to the real sales of the LCDP as was available. However, due to the sales construction and a general lack of information the accuracy of these numbers is low. The information can only be used in combination with the other metrics to provide some insight in company size and resources. Note that only Mendix, Outsystems, and Betty Blocks are mainly focused on their LCDP. Since Appsemble is a recently founded startup, they are not featured in the comparison table.

| Company | Google Trends (Worldwide) | Employees | Employee growth 1y | Job openings | Subscription revenue |
|---|---|---|---|---|---|
| Mendix | 54 | 1.117 | 37 % | 411 | $ 100M+ (2020) [55] |
| Outsystems | 74 | 1.976 | 32 % | 229 | $ 100M+ (2018) [68] |
| Microsoft (Power Apps) | 33 | 212.061 | 9 % | 19.102 | N/a |
| Salesforce (Platform) | 73 | 63.432 | 16 % | 17.097 | N/a |
| Appian | 87 | 1.765 | 18 % | 342 | $ 198.7M (2020) [3] |
| ServiceNow (Now platform) | 70 | 16.886 | 24 % | 2.147 | $ 4.5B (2020) [79] |
| Betty Blocks | 29 | 185 | 0 % | 21 | N/a |
| Appsemble | N/a | 3 | N/a % | N/a | N/a |

Table 2: Comparison of low-code development platform companies

As described earlier, there are only a handful of companies that are fully focused on their LCDPs within this set. These are Mendix, Outsystems, and Betty Blocks. Another group that can be made are those that see their LCDP as an extension of their current ecosystem. These are Microsoft Power Apps, Salesforce Platform, Appian, and ServiceNow with their Now platform. The next section will briefly introduce every LCDP that is in the scope of this thesis, along with one innovative newcomer.

## Mendix

Mendix was founded in the Netherlands in 2005. Mendix was bought by Siemens in 2018 for $ 780M, greatly increasing awareness and available financial resources. The Mendix platform offers both no-code and low-code development and operates in the (private) cloud. Most customers are based in Europe and North America from a wide variety of industries [90]. The Mendix platform focuses on both citizen developers and professional developers. It has advanced machine learning and AI assisted features and a tight integration with SAP components.

## Outsystems

Outsystems was founded in Portugal in 2001. Its customers are spread across the world, but mainly in Europe and North America. Their customers are mostly from the insurance, professional services, finance and banking sectors [89]. The platform is focused on low-code development, but does feature some no-code tools. Like Mendix the platform offers strong AI development support in the form of AI augmented development and automated unit testing using AI.

## Microsoft Power Apps

Microsoft Power Apps launched in 2016. It is currently part of the Power Platform, together with Power BI, Power Automate, and Power Virtual agents. The main strength of this platform is the synergy it allows between these products, alongside Azure, Office 365, and more. The platform is focused on citizen developers. Another interesting

development is its cooperation with OpenAI. This allows future developers to generate queries with just natural language statements [89].

### Salesforce platform

The Salesforce low-code platform was launched in 2015 under the name Salesforce Lightning. The platform supports no-code development as well as low-code development using either JavaScript or Salesforce's own programming language Apex. While most LCDPs offer a marketplace for application sharing, the Salesforce platform is unrivaled with its large ecosystem [89]. Salesforce has mostly large customers that are already a customer of their CRM products [90].

### Appian

Appian is one of the oldest LCDP companies founded in 1999 in the USA. Appian is also unique as it is the only large LCDP developer that is featured on the stock exchange. Its platform offers separate design studio's for professional and citizen developers similar to Mendix. Appian focuses mainly on large organizations with sizeable share in the finance and government sectors. One reason for this is their heavy investment in security certifications and audits, with a leading edge in public sector high-security certifications [88].

### ServiceNow

ServiceNow is one of the larger organizations offering LCDP under the name Now Platform. Its clients are mainly large IT organizations that are already using other products from ServiceNow, similar to the Salesforce platform. However, the growth of the platform is limited as the business model seems not to be suited for non-existing ServiceNow customers [89]. The platform is using its vast resources to innovate its AI, ML, and NLP capabilities through several acquisitions and other investments [18]. The Now Platform is well suited for citizen developers, but discouraged for professional developers wanting more control over their applications [88].

### Betty Blocks

Betty Blocks is unique in this line-up as it tries to be a complete no-code development platforms. Forrester calls their strategy "no code for enterprise apps or bust" [76]. They are described as visionaries in their sector. Gartner reports that while a no-code only approach has many benefits for citizen developers, when a building block lacks required customization options, the developers have to resort to complex programming languages [88]. In Table 2 it shows that they might lack the resources to keep up with their competitors who are growing rapidly. However, a recent funding round yielding $ 33M for Betty Blocks might help them expand [7].

Currently, Betty Blocks says it focuses on three kind of developers: citizen developers, no-code developers, and pro-coders [6]. A citizen developer would be someone that, ideally, comes from the business side and can develop simple applications after a short introductory course. A no-code developer is one step above a citizen developer and is someone who can utilize the platform to its fullest without using more coding than a relatively simple styling language like CSS. A pro-coder would be someone who can extend applications with high-code like Java and REST in order to create necessary customizations like a webservice call. Interestingly, this no-code platform thus still focuses partly on pro-coders. Betty Blocks is currently not suited for core systems, it instead finds its added value mainly in developing supporting applications that enrich existing systems.

### Appsemble

An interesting newcomer in the LCDP market in the Dutch public sector is Appsemble. Founded in 2018, the project started as a cooperation with the municipality of Amsterdam under the name of OpenApps [67]. The vision behind this project is to create an open source LCDP where applications can be shared between organizations. Currently there are multiple applications available from the municipalities of Amsterdam and Amersfoort [4]. Appsemble's initial focus seems to be on the Dutch public sector specifically as it has support for Common Ground (a new growing initiative from the Association of Netherlands Municipalities to reform and modernize the municipal information sharing infrastructure [19]).

The platform itself differs from its competitors by having a higher focus on code, but making this code more elegant and simple with an optional visual editor besides it. The focus is therefore not on allowing citizen developers to create applications, but to improve existing developer productivity. The Openapps/Appsemble initiative could in theory create a cheap alternative for the public sector with a reduced chance on vendor lock-in (becoming so dependent on a product or service that switching to another would come with very high costs) due to lower costs and an open source platform. This set of advantages could benefit smaller municipalities with a low budget. However, the success of this LCDP is highly dependent on finding enough organizations to work with the platform, thereby creating enough applications to be shared and keeping the platform updated and secure. Another question that remains is if public organizations can adopt applications from others as easily as is claimed. So far, no public organization has copied and reused an application via this platform.

## 3.1 High-code, low-code, and no-code

As discussed in the introduction, there are multiple names for LCDPs. Most of these can be seen as marketing efforts and will not receive much attention from the average IT-professional. However, the difference between high-code, low-code, and no-code is important to know as a non IT-professional dealing with LCDPs. In order to give the

reader a better background on these terms, this chapter will briefly explain what they mean in a practical sense.

As seen in Figure 1, different kinds of programming have different abstraction levels. When talking about high-code they refer to a low level of abstraction. This means complex and difficult code that requires a significant amount of training for someone to be able to understand it. A no-code approach aims for a higher level of abstraction. Instead of writing difficult and complex code, visual interfaces and models can be used. These are then transformed into high-code using code generation principles or model interpretation [12]. Most low-code and no-code development platforms still need high-code when applications require functionality that is not supported by default. As an example, the Betty Blocks platform, marketed as a no-code platform, might still need high-code for certain customizations.

Some LCDPs have different environments where one is focused on IT professionals and one focused on citizen developers where high-code is not possible. An example of this can be found in the Mendix Studio and Mendix Studio Pro environments. Figure 6 and Figure 7 show the same application displayed in different development environments from Mendix. Figure 6 shows the more citizen focused Mendix Studio which is accessible by browser. Whereas Figure 7 shows some of the more advanced options available in Mendix Studio Pro, a desktop based application. The pro version offers both high-code, low-code, and no-code, the studio version only offers no-code.

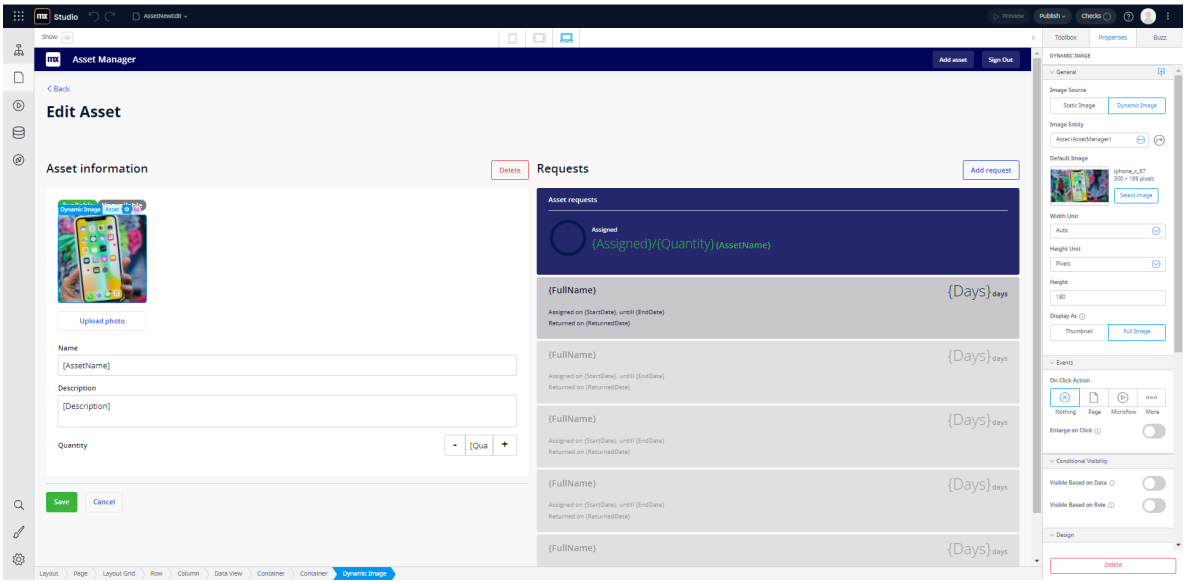



Figure 6: Mendix Studio example

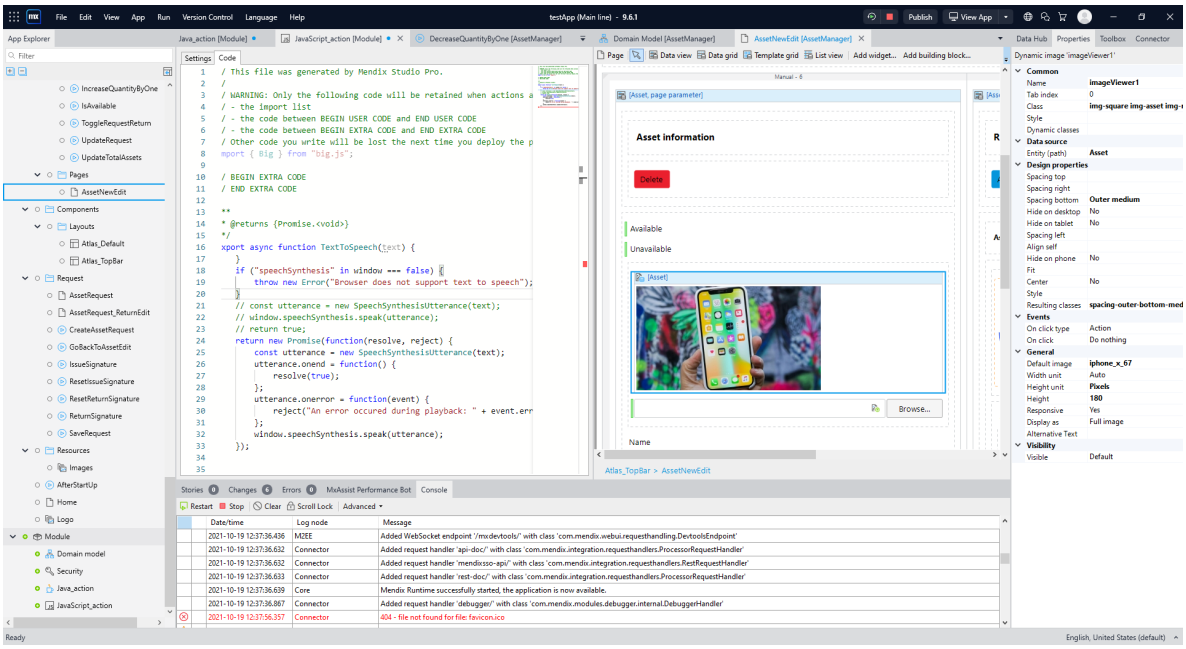



Figure 7: Mendix Studio Pro example

From the brief analysis of the current LCDP market, a few things can be said. First, the market seems to be growing fast with employee growth numbers well into the double digits. Mendix is leading with a staggering 37% growth year over year. Betty Blocks, the only pure play no-code development platform selected, lags behind with no employee growth. The popularity and revenues are hard to compare directly since not all industry leaders (according to Gartner [90]) are solely LCDP companies. Interestingly, every company seems to have their own approach to how accessible their platform is for citizen developers. Some even feature completely different interfaces for professional developers and citizen developers as seen in Figure 6 and Figure 7. Another find is the push by startups like Appsemble to create open source ecosystems to, in their words, reduce the chance of vendor lock-in. While this could present opportunities for the Dutch public sector, it also leaves open several questions on its feasibility.

# 4 Methodology

This chapter will go over the research approach, research perspective, and research design of this study. It will go in depth over both the qualitative side of the study and the quantitative side.

## 4.1 Research approach

The first guiding question asks for a way to measure LCDPs effectiveness in the public sector. The second guiding question requires an actual measurement of this effectiveness and analysis of the results. In this research, these differing yet connected questions are approached with a mixed method study. The first question will be done qualitative and the second guiding question will be done quantitatively. This design, starting with a qualitative phase and then sequentially doing a quantitative phase is called an exploratory sequential mixed method [20].

In general the first qualitative phase ends with a deliverable of codes or conceptual themes. These deliverables are then used to direct and guide the following quantitative phase. Within this theory there are two main variants: the theory development variant and the instrument-development variant [63]. As in this case both the qualitative and quantitative phase represent a guiding question (the first and second), no preference can be made and therefore no variant is used.

Applying this theory, the research approach is as following. The thesis project uses an exploratory sequential design consisting of two phases. In the first phase an initial conceptual framework is created using existing literature which is then refined using expert interviews. This conceptual framework is utilized in the second phase to construct the survey that should now be well adapted to the target audience. The conceptual framework and survey are the deliverables for the first guiding question. The survey is then distributed and the results are analyzed to answer the second guiding question. This research approach is visualized in Figure 8.

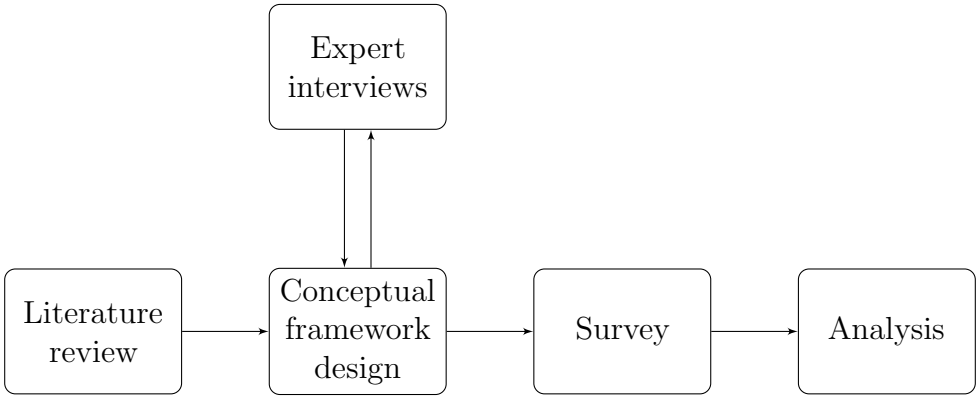



Figure 8: Research design

## 4.2 Research perspective

When reporting research findings it is important to talk about this research' perspective on e-government. Comparing this perspective with the overall metatheory can help uncover basic assumptions made. To assess this this thesis uses the metatheory presented by Meijer & Bekkers [51]. It distinguishes three dimensions: explaining - understanding, holism - individualism, change - maintenance. In the first dimension this research uses both perspectives, but with a focus on explaining. The literature review, expert interviews, conceptual framework, and survey are there to find patterns that can help explain important variables of e-government for this study. Most of this is from an outsider's perspective, however, some parts of the survey also allow respondents to explain the specific socio-structural mechanisms (e.g. norms and rules) that can help understand their perspective. The downside of this mainly explaining focus is that it can "lack attention for context and behavior of specific actors" [51].

The second perspective is on conceptualizing e-government as holism or individualism. As this thesis is mostly focused on structure and not the actions and behavior of individual actors it takes a holistic approach. As mentioned, the survey does allow respondents to clarify their answers leading to some aspect of individualism. In this way, the thesis assumes that organizations contain of many parts which each "fulfill specific functions, coordinated in a proper way to achieve specific goals" [51]. Lastly, their metatheory features the dimension of change or maintenance in e-government. This distinction looks at e-government either as a radical change or as a gradual change of the status quo. In this thesis a maintenance perspective is chosen, implying that e-government initiatives have the tendency to follow pre-existing system characteristics.

## 4.3 Qualitative research design

### 4.3.1 Conceptual framework

A big deliverable for this project is the framework conceptualizing the value of LCDPs for the public sector. The aim is to create a framework that can guide research and practice on this subject. The framework is initially based on the theory discussed in the previous chapters. The framework will therefore be based on scientific literature of public value, e-government, and existing low-code research. It will also use additional grey literature from LCDPs vendors and research institutes like Gartner and Forrester.

As discussed in the literature review, three views can be identified on the relationship of IT resources, organizational capabilities, and performance. For this thesis, the process view is selected as the theoretical basis instead of the other two views. The process view gives, besides input (IT resources) and output (organizational performance) also a way to explain the process between these (organizational capabilities). This is preferred over the complementary view, as it seems probable that this is easier to follow for practitioners. Additionally, it can provide an explanation on how certain IT resources are utilized for organizational performance. The configuration view, although sounding interesting, was not used as it is incompatible with the research goals and setup. That

is why in this paper, the model by Pang et al. will be used as a template instead of the more popular resource-based view, dynamic capabilities, and production function model. Instead of the information technology resources presented in the framework of Pang et al., this framework will feature the advantages of LCDPs. The organizational capabilities of their framework remain. The public-value frontiers were replaced by a generalized set of public value categories. The full conceptual framework can be seen in appendix B.

As discussed in chapter 1, there are only few good sources available on this topic. Most literature that is available is either of a related subject or of questionable objectivity. This makes it hard to create a reliable conceptual framework just from these sources. The decision was made to perform several interviews with domain experts to validate the framework. Therefore, the initial conceptual framework was reviewed with qualitative semi-structured interviews using domain experts at Capgemini. Their knowledge on low-code platforms, the public sector, or both were used to evaluate which aspects of the framework needed revision or were agreed upon. The goal was to both validate the framework and to make the framework more understandable for practitioners to promote future usage. As Capgemini is a huge organization, Marien Krouwel, the Capgemini contact point for this research project, helped with selecting which experts had the expertise and time available to do the interview. The instructions given were to find experts on LCDP or IT projects in the public sector in general. This way both the technical and public sector parts of the framework could be validated. The experts needed to have enough years of experience at, preferably, different enterprises or projects. The position of the experts in the project was less important, but a wide variety of differing positions from developer to enterprise architect was preferred. These criteria gave eight experts, of which six were willing to perform an interview. If there were still large discussion points remaining after the last interview on the list more interviewees could be found.

Performing expert interviews at a large consultant company is a very practical approach of getting high quality data on a complex problem. It takes up relatively little time and is effective in getting detailed information. The major downside of this sample is that a possible bias is introduced in terms of selective sampling. By giving guidelines and interviewee criteria this is somewhat reduced, but not taken away. Another bias introduced is that all participant have a motivation to exaggerate advantages and downplay the disadvantages of LCDPs.

Initial talks after finishing the initial conceptual framework based on the literature gave an indication that there might be many improvement points. Therefore, the possible risk was that all interviews, due to limited time, would only have time to mention the same issues or not have enough time to finish the whole question list. To combat this the decision was made to instead make several interview-improvement cycles. The idea was that this way the most obvious flaws could be resolved in the first round, giving the following experts more time to talk about other topics.

This new interview-improvement cycle was initially planned for three cycles, as there

were six experts who were already willing to participate. Each cycle has two experts that individually give their opinions on the latest framework. Their feedback is then processed and incorporated in the framework to create a next version for the next two experts. As the experts had differing amounts of experience with these platforms and the public sector, not every feedback item immediately led to changes in the framework. When there were doubts or differing opinions, the point was moved as a question to the next interview cycle in order to get more feedback. The interview-improvement cycles would stop once the interviews would yield no large disagreements or discussion points.

Every interview followed the list of questions seen in Section 4.3.2. The questions are mostly open in order to minimize prior bias and get as much information as possible. The interviews themselves were recorded with permission to ensure all feedback was utilized. The answers given often contained confidential project information which, due to the mentioning of the interviewees name, could be traced back to an individual project or organization. Therefore, no direct citations are used in this thesis and the expert names are only noted in the acknowledgments section.

The interviews were concisely described and analyzed every interview cycle. As each cycle only consisted of two interviewees and the interviews followed the question list based on the initial conceptual framework no formal coding was necessary. This deductive method means that the conceptual framework was tested each interview cycle. This approach was chosen as the initial framework was based on questionable source material and thus many changes were expected. Letting all experts review the initial framework would have the risk of them pointing to the same (to them) obvious mistakes. With this iterative approach this risk is limited.

Of each interview a summary was made with the main statements made on the conceptual framework of that round. Each summary was offered to the interviewee to review them for accuracy and completeness. This is vital to make sure key statements in the interview were not missed and answers were not misinterpreted. In the expert opinion section these summaries are grouped together and mixed to make it harder to trace who said what statement. After each summary was made the interview records were deleted.

### 4.3.2 Interview questions

The following interview questions were used, based on the expert interview protocols from [28]:

1. Starting off with a description of the research project, the vision behind the conceptual framework, the main goal of the interview, the expectations of the interviewee, permission to record the interview, and if there are any questions before starting.
2. Mapping the experts background knowledge regarding LCPDs or related experiences:
    - What do you understand under low-code?

- What do you understand under low-code development platforms?

3. Introductory questions:
    - How long have you worked with LCDPs?
    - Which LCDPs platforms do you have experience with?
    - Could you describe what your current position encompasses?
    - How is your position related to LCDPs?
4. Evaluation of the conceptual framework:
    - LCDPs advantages
        - What do you think about these LCDPs advantages?
        - Which advantages should be excluded from the list?
        - Which advantages should be added to the list?
        - What do you think about the grouping of the set of advantages?
        - How would you group them?
    - Organizational capabilities
        - What do you think about the list of key organizational capabilities that can drive public value creation?
        - Which capability should be excluded from the list?
        - Which capability should be added to the list?
        - For each capability, which relations would you draw from a LCDPs advantage (group) tho this capability
    - Public-value
        - What do you think about the list of public-value categories that e-government is supposed to yield?
        - Which public-value category should be excluded from the list?
        - Which public-value category should be added to the list?
        - For each public-value category, which relations would you draw from capabilities to this public value category.
5. Closing:
    - What do you think about my work?
    - Can I use your name in the scientific paper, or do you prefer to remain anonymous?

- Do you have any questions or comments?
- May I contact you if I have any further questions?

### 4.3.3 Interviews

Six experts from Capgemini were interviewed for validation and improvement of the framework, two at each cycle. Of these six, two are enterprise architects with decades of experience. Two are solution architects of which one with a focus on LCDPs and one with a more general focus on the public sector. Another interview was held with a business technology consultant with five years of hands-on experience with different LCDPs at several organizations in the public domain. Lastly, a business analyst and LCDPs developer were interviewed.

The individual interviews were conducted between May and July 2021. The average length of the interviews was 1 hour and 34 minutes, with the range spanning between 46 minutes and 2 hours and 52 minutes. Some interviews were done over multiple meetings to be able to fit into the tight schedules of the interviewees. All interviews were conducted in Dutch and were audio-recorded to allow for accurate, detailed data analysis. Five of the interviews were performed via Microsoft Teams due to Corona restrictions and one was performed face to face.

## 4.4 Quantitative research design

The vision for the quantitative research is to develop a survey based on the developed framework. In the survey, the independent variable are LCDPs (in comparison to traditional programming methods). The dependent variables are picked from the conceptual framework seen in Section 10. As well as several controversial or judged as most important performance aspects of LCDPs by the experts. These are, amount of manual coding, technical interoperability, ease-of-mastery, maintenance, business-IT alignment, vendor lock-in, and customization. Every dependent variable is looked at in a 'survey block' consisting of one 6-point likert-type question, one importance score (0-10) question, and an optional comment box where respondents can explain their choices. An example of a standard survey block can be seen in Figure 9. In addition, the survey has two open questions, one allows respondents to add new advantages or disadvantages and one for discussing LCDPs and public value.

To map the background of the respondents and their organizations the following descriptor questions will be asked:

- At which organization are you currently working?
- Would you describe your current role as more IT-focused, business focused, or both?
- What is your current job title?
- What coding experience did you have prior to using the LCDP?

- What low-code development platform are you currently using?
- How long have you been working with this/these platforms?
- Is the LCDP part of a pilot project?
- How many applications have been developed using the LCDP in your organization?

The survey will be made available only in Dutch, as an interviewee has pointed out that certain public sector organizations might be offended by an English survey. The full Dutch survey can be found in Appendix C. Every item that is asked about in the survey has the same three question boxes in the survey seen in Figure 9.

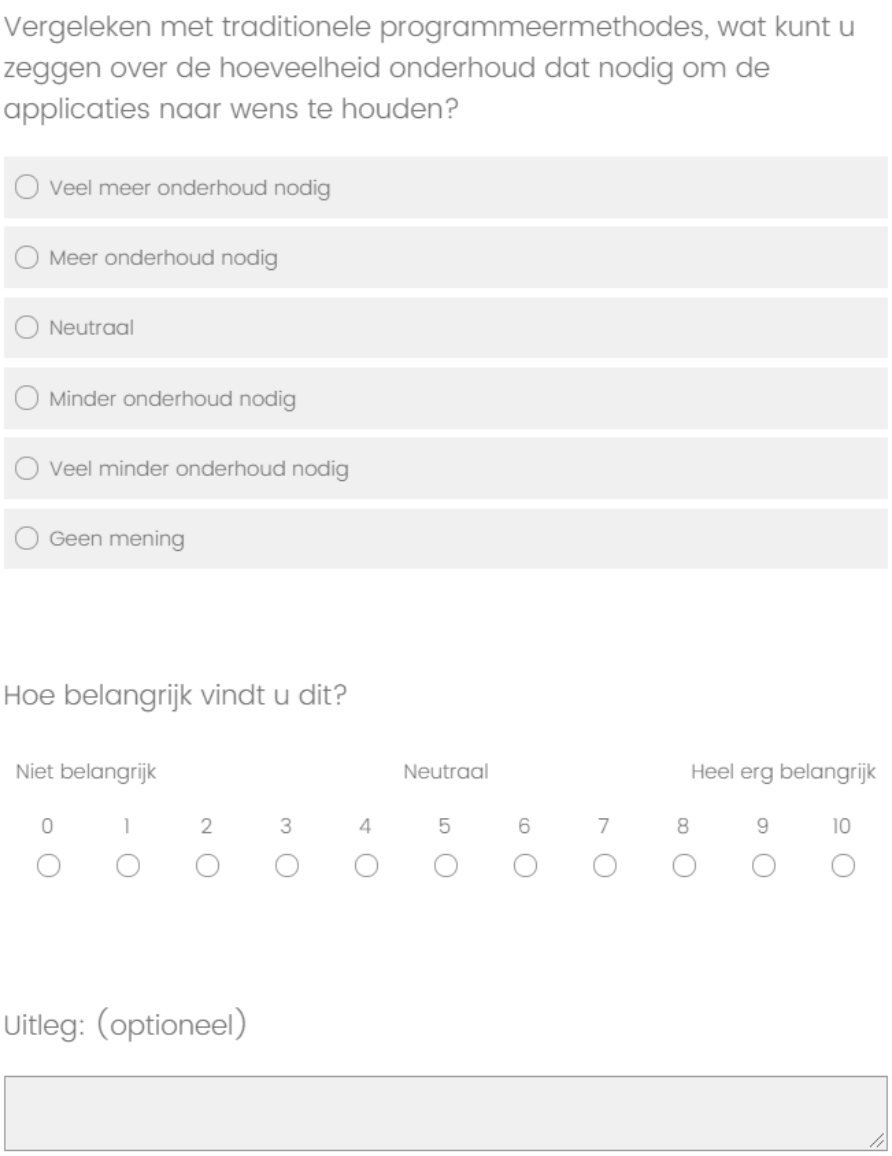



Figure 9: Standard survey block in the survey

The first block of questions will go over the most controversial or important LCDP performance aspects found in the expert interviews. Most of these were judged as advantages and some were seen as disadvantages. In the survey each of these will be framed neutrally leaving it to the respondents whether they see it as an advantage or disadvantage. It will also feature every advantage group. Respondents can choose whether they agree with the statements following a 6-point Likert-scale. Next, the respondents will be asked to map the advantage groups to the organizational capabilities. In the third block the respondents will be asked for each connection made in the second block, to which public value category they think it contributes. This dynamic question should be possible to be made on the web-based Qualtrics survey platform. If this is not possible, the connections could be based on the framework and the links made by the domain experts in the interviews.

The 6-point Liker scale questions consist of two agreement, one neutral, two disagreement options, and a no opinion option. This last option was added for two reasons. The first reason is that one of the goals of the survey is to get as many and as diverse response group as possible. A respondent coming from a non-IT background might be discouraged to continue the survey when asked to judge about a majorly IT subject (e.g. amount of manual coding). The hope is that when given a no opinion option there is a higher chance that they will continue. Another reason is that the analytics used on the survey use the median to determine the sentiment. Respondents who have no experience with those aspects would otherwise be forced to pick the neutral option resulting in a bias in the final outcome.

The downside of this approach is that it gives respondents an easy option to pick when faced with a more difficult question. This means that there is a bigger chance on getting less information out of the sample size that is reached, which can be a problem when ending up with a small sample size. This is a big down side of this approach, but the decision was made to keep the no opinion option as the quality of the answers would go up significantly.

Even though the survey has open-ended questions and therefore allows respondents to give their opinion. It does not necessarily mean that this is their fully nuanced opinion on the matter. This is also not expected for an optional comment question. If the results of the survey would raise any major questions, follow-up interviews could be held. The survey ended with a question if follow-up questions could be send to the respondent and via what mail address. However, as no major issues were found and almost all answers were clear no follow-up interviews or mail exchanges were done.

Due to concerns regarding legal issues and safety, Dutch organizations seem to prefer LCDPs made in the Netherlands, specifically Mendix and Betty Blocks [28]. Therefore, it is possible that the sample following the survey has a bias towards these platforms. No meaningful way was found to mitigate this bias.

A note some experts made in their interview was to ask the survey respondents to make the lines between LCDP advantages to organizational advantages to public value. This

would have more added value in their opinion, as the respondents should have a better grip on these connections. In addition, if examples could be given this would greatly increase the strength of the conceptual framework. However, as the survey was already on the long side this idea was scrapped.

The target group of the survey are people working for a public organization working with LCDPs. Both IT and non-IT professionals are welcome as well as employees from all layers of the organization. The strategy for the survey send out was to leave one or two weeks between contact points to increase the chance it would be read due to holidays or other activities. The survey was also postponed to make it start after the peaks of the holiday season. The survey was opened on 26th of August 2021 with a LinkedIn post via the network of thesis supervisor Marien Krouwel, who is highly connected within the LCDP world. This was combined with a post in the LCDPs expert LinkedIn group.

Soon after the Dutch public-sector clients of Capgemini who are known to use low-code development platforms were asked to participate via mail. An example of this message can be seen in appendix D. Both these methods yielded decent results considering the target group size, but were still far from a significant amount. To increase the amount of respondents the Royal Dutch Association of Information Professionals (KNVI) was contacted. They had recently done a session on a LCDPs that many professionals had followed. The participants of that session were contacted via the KNVI seen in appendix D.

As the amount of respondents was still low, reminder mails were send out two weeks after their initial contact point. These mails can also be found in appendix D. During this process, several people were approached via LinkedIn directly via friend requests. However, as this had severe limitations and was going slowly, a premium account was bought till survey was closed. Using LinkedIn InMails several public sector employees who had stated to have experience with LCDPs in their profiles were approached. However, as most of the messages were unanswered the premium credits ran out quickly (these 'InMail credits' get returned when people reply).

These added channels yielded some results, but still no significant sample size was reached. To give people more time to fill in the survey it was kept open another two weeks. Meanwhile, several other options to increase the amount of respondents were considered. One of these was to ask consultants working at Capgemini who have experience with LCDPs in the public sector. This could easily double or triple the amount of respondents. However, this would reduce the quality of the overall results significantly. The objectivity of the answers would be questionable, though the experts were highly critical in the conceptual framework. It would also shift the perspective from the inside the organizations to outside. Even though the sample size could be increased, it was decided that the high quality that was currently reached was more important.

This logic was also the reason for not contacting LCDP vendors and other consultancy companies. Another option that was proposed was to ask students active in the public sector and/or had experience with LCDPs. This obviously had a much lower risk of

subjective answers. However, due to the lack of experience and many being active in Belgium this offer was refused. This meant that the survey was closed on 22 October after almost two full months with a total of 23 respondents, of which 16 could be used. Six responses had to be deleted as they were not part of a public sector organization. One response did not fill in the questions after the descriptor questions. The remaining 16 responses finished the survey completely. The full survey outcomes can be seen in Chapter 6.

### 4.4.1 Qualitative side of the survey

In order to allow respondents to clarify their answers each item (two questions per item) in the survey had an optional comment box. Not many comments were expected to come from these optional boxes, but in total 117 comments were made across the 16 participants. As some of the comments had interesting information and there were few respondents in total it was decided to analyze this information in depth as well. This means that besides the exploratory sequential design there is an additional short qualitative study on the survey results.

These comments, along with two open questions from the survey, will be analyzed in a separate section in the results based on the qualitative research methods guide from Anderson [1]. The section should use quotes that are most representative of the sentiment found in the data. Each quote should end with a respondent number. This way it is possible for the reader to see that not all quotes come from the same individual, but rather come from a wide range of participants. This can also reduce the confirmation bias of the researcher.

In addition, it is important to highlight contradictory or deviant cases in the data. This helps with making sure that the researcher bias is minimized and the perception of the data is not altered. As described earlier, quotes should be picked that represent the sentiment of the data best. Therefore, there needs to be a careful balance of picking quotes that fit the overall sentiment and quotes from deviant cases. As the quotes were originally in Dutch they have been translated to English by the author.

### 4.4.2 Statistical methods

The first question per block as seen in Figure 9 is a 6-point Likert scale with an extra no opinion option. To analyze a Likert-type question it is typical to approach it as ordinal-data. Some argue that it is possible to turn it into numeric data to allow parametric analysis. However, this requires a minimum of 5-10 responses per choice and normally distributed data. Both of these conditions are not met. In order to analyze ordinal Likert-type questions it is possible to get the central tendency using mode or median [9]. In Figure 4 the sentiment score was determined using the median.

For the importance score a t-test can be performed to determine if the observed response is significantly different than a neutral response. The answers are between 0 and 10 meaning that a neutral response is distributed around 5. The Z-test is not used as the

sample size is too low ($n < 30$). For the t-test, an alpha level of 0.05 will be used. A one-tailed t-test is used for the importance scores of which the results are summarized in Figure 5.

# 5 Expert review of the initial conceptual framework

In this section, the feedback of all six interviews is summarized in a few short paragraphs. It highlights how agreement on certain terms was reached and how certain marketing terms were translated to more unambiguously and accurately worded items. It also highlights the doubts and criticism that remain on certain aspects of the framework. This section is meant to describe how different experts look at the framework and place the conceptual framework in a more realistic context. After the experts reviewed the conceptual framework their comments were processed of which the outcomes can be seen in Table 3 and Figure 10.

The advantage of complexity reduction was removed as the concept of complexity can be multi-interpretable. As interpreted by one of the interviewees, complexity is characteristic of the context in which LCDPs is used. Saying complexity is reduced by being able to reuse certain modules in development was therefore not clear. The decision was made to replace this element with increased reusability.

Process automation was found to be unclear in the framework. LCDPs could be used to automate some low-level processes like e-mail send outs, record changes, etc. However, this is not exclusive to LCDPs and it is less impactful as the wording suggests. Therefore, it was removed from the framework.

The variable cost element was found to be an interesting advantage at first glance. However, it is not a characteristic of LCDPs, but more a result of the business model that most platforms-as-a-service vendors use. Therefore, it was argued that it is not inherent to LCDPs and should be removed.

Escaping legacy debt was found to be too much of a marketing term and was changed to reducing legacy debt initially in order to word it more accurately. The opinions on this item were slightly conflicting in the first round and it was decided to leave it in for the next interview cycle. The next interviews both stated that even reducing legacy debt was too much of a marketing term with little to no evidence. Architecture here is more important than the LCDP. Therefore, the item was removed from the framework in its entirety.

Another advantage that saw conflicting opinions in the first round was the reducing dependency on hard-to-hire skills. One side argued that the platform is easier to learn than regular programming languages for analytically minded people and it could therefore be valid. However, other interviewees argued that there is currently a lack of Mendix developers meaning technically there is still a dependency on hard-to-hire skills, just different skills. Initially the item was renamed to steep learning curve which was agreed upon by some, but this could be interpreted in both a positive and a negative way. Thus the item ended up as ease-of-mastery.

Agile-ready was seen as a bit of an understatement and could be improved by renaming it to agile-supportive. The item also saw some criticism as any programming language with the right tools is agile-ready and agile-supportive. However, what was agreed upon

in the end is that most LCDP come with these tools out-of-the-box with almost no setup required. Therefore, like the items easy deployment and continuous integration, it was decided to leave these items in.

Items that got a lot of mixed answers were those of citizen developers and inclusion of business profiles (later changed to facilitating business-IT alignment). Every LCDP has a vision for what a citizen developer should be able to do and where a professional developer should take over. Front-end changes in most platforms could easily be done by citizen developers. Developers are usually needed for the more technical tasks such as creating or connecting to API's. Some platforms are more focused on citizen developers and some more on professional developers. In the case of a citizen development focus this is likely to lead to extra limitations on what can be made with this platform.

The concept of citizen developers was praised by the experts, but not entirely believed as the required formal and abstract way of thinking is hard to learn as a non-developer. They did agree on that analytically minded people, depending on the environment, could be able to become productive citizen developers. What should be noted here is that even though the experts did not see citizen developers in their projects, most of the experts also did not participate in enough projects to be able to judge this aspect fairly. As the concept of citizen developers was supported for analytically minded people and existing developers can develop faster the item was merged into ease-of-mastery. Ease-of-mastery has the side note that this is only meant for existing developers and analytically minded people.

The inclusion of business profiles partly relies on citizen developers as supposedly business employees could develop their own applications. As this in general was not believed by the experts this reasoning was removed. However, the inclusion of business profiles also saw different explanations, as it could be argued that technical details could be skipped and therefore there is more time available for e.g. a better inclusion of business demands. Through a visual interface and fast prototyping it is also easier to communicate with the business side. To better describe this idea the item was changed to facilitating business-it alignment.

Another item that was hard to assess was the cost reduction item as the experts had low direct experience with the financial aspects of all their projects. For some organizations, the LCDP is an extra and does not replace existing IT systems by itself and thus costs extra. Some organizations use the LCDP to make only one application which might be more expensive than traditional programming methods, due to sometimes high license fees. What found some support was that a cost reduction can be achieved when making multiple applications using the same LCDP. In that case, the licenses, developers, resources, and the platform itself are already in place resulting in a cost reduction in those areas.

Interoperability was deemed to be too broad as in technical terms there are all sorts of interoperability defined in, e.g., the New European Interoperability Framework [25]. Therefore, to more correctly label the item it was changed to technical interoperability.

The next round of experts still had some doubts on technical interoperability as an advantage as it is very platform dependent. E.g. out-of-the-the-box Mendix works very well with SAP, but the Salesforce ecosystem does less so. This was also noted to be a major factor when analyzing what LCDP to recommend. Even though this item might be platform dependent it was decided to leave it in the framework as the major players mostly suffice in this aspect.

Support for easier maintenance saw mixed opinions from the experts. Some argued that easier maintenance can be achieved when the application is built well and is not too complex, similar to a high-code application. One advantage is that an application manager (functioneel beheerder) can change more aspects of the application more easily to keep it up-to-date. If this is enough for a cost reduction was not agreed upon.

Besides discussions about the individual performance aspects themselves, the relations between them and organizational capabilities was also discussed. One expert found the relation between the LCDPs advantages and the public service delivery capability, public engagement capability, and co-production capability unclear and dependent on many factors. For resource acquisition capability and public-sector innovation capability the relation is reasonable. Another said that the LCDPs advantages are hard to translate to public-value categories directly and that using organizational capabilities as intermediary step is good. However, these capabilities were still quite abstract. A suggestion would be to ask for the connections from advantages to capabilities to public value in the survey. Another expert noted that for the organizational capabilities the 'public' can be removed to create a broader framework which would still hold up.

The public value column of the framework was generally seen as not strongly related by the interviewed experts. Some argued that LCPDS could be used to improve these values, but it is not certain or inherent to the technology. One expert noted that LCDPs could just as well be used to close off the government and reduce these values. It depends on how the advantages, e.g. reduced development time, are used within the organization. This time could be used to increase cooperation with citizens the public values improve. However, this time could also be used in a way which does not improve public values. Therefore a suggestion was made to explicitly ask for these public values in the survey.

Several discussions with the experts highlighted the importance of a survey to be done on the advantages in practice. Items such as cost reduction, ease-of-mastery, easier maintenance, business-it alignment, technical interoperability were hard to confirm either due to insufficient experience or the complexity of the items. Similarly, some organizational capabilities and public value categories were seen as only moderately related to the advantages.

## 5.1 Conceptual framework

The conceptual framework can be seen in Figure 10. It shows the relations from LCDPs advantage groups to public value categories. Every item and relation in this figure followed from the had interviews. Each advantage group consists of several smaller advantages seen in Table 3.

| | Scalability | Knowledge integration | IT-productivity increase | Cost reduction | Increased business-agility |
|---|---|---|---|---|---|
| End-product portability | | | x | x | |
| Easy deployment | x | | x | | x |
| Continuous integration | x | | x | | x |
| Platform-as-a-Service | x | | x | | |
| Technical interoperability | x | | x | | |
| Visual interface | | x | x | | |
| Real-time editing | | x | x | | |
| Reusable components | x | | x | x | |
| Ease-of-mastery | | x | | | |
| Less hand-coding | | | x | | |
| Easier maintenance | | | x | x | |
| Fast prototyping | | x | x | x | x |
| Agile-supportive | | x | x | | x |
| Facilitating business-IT alignment | | x | | | x |

Table 3: Individual LCDPs advantages mapped to advantage groups

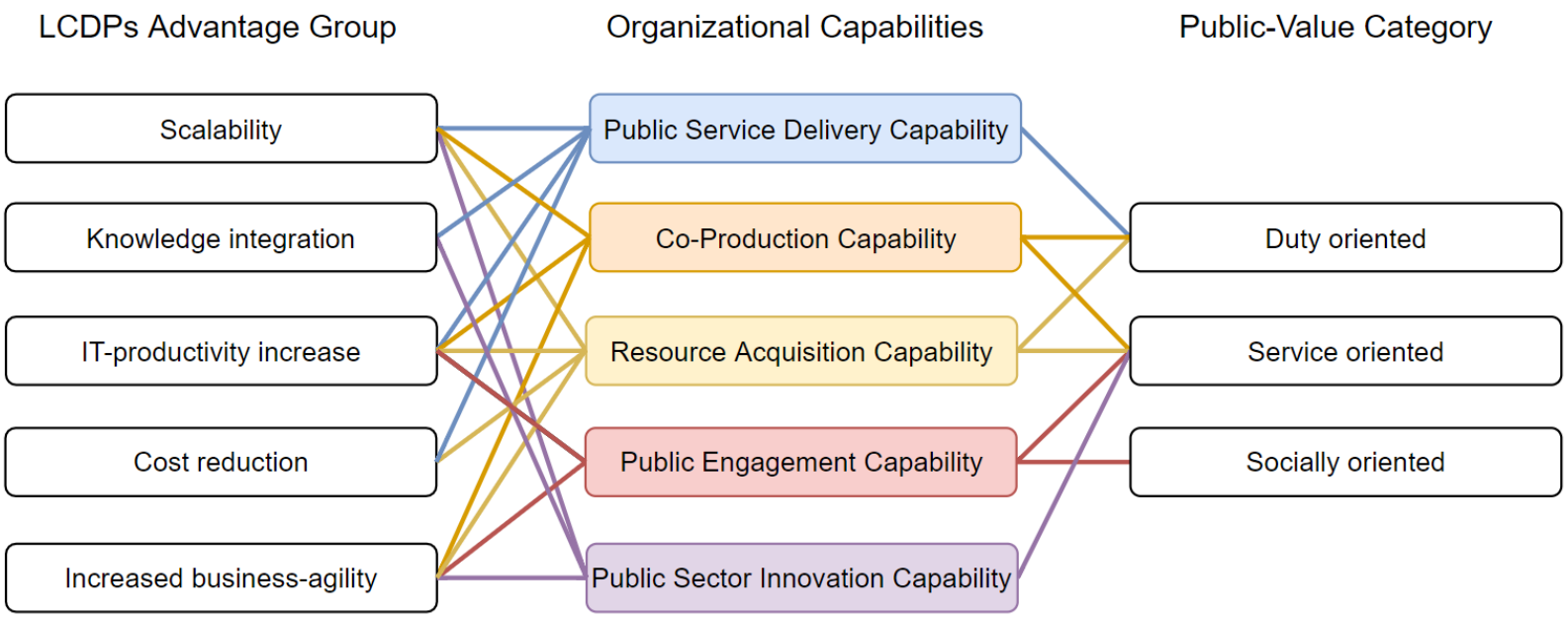



Figure 10: Conceptual framework: Public value creation through low-code development platforms (Colors only for visibility)

# 6 Survey

## 6.1 Results

In total the survey was filled in 23 times. Six responses had to be deleted as they were not part of a public sector organization. One response did not fill in the questions after the descriptor questions. The remaining 16 responses finished the survey completely. One response was flagged as a bot by Qualtrics, but this was deemed incorrect as the explanation fields were filled in too accurately to have come from a bot. On average the participants took 24 minutes to finish the survey. This average is slightly skewed due to a few outliers. This is expected as it is easy to get distracted while filling in a longer survey. The median of 16 minutes shows that the survey was close to its intended target of 10-15 minutes.

The total of 16 respondents were from 14 different public sector organizations. They come predominantly from IT or information oriented divisions with only two respondents coming from a business oriented division.

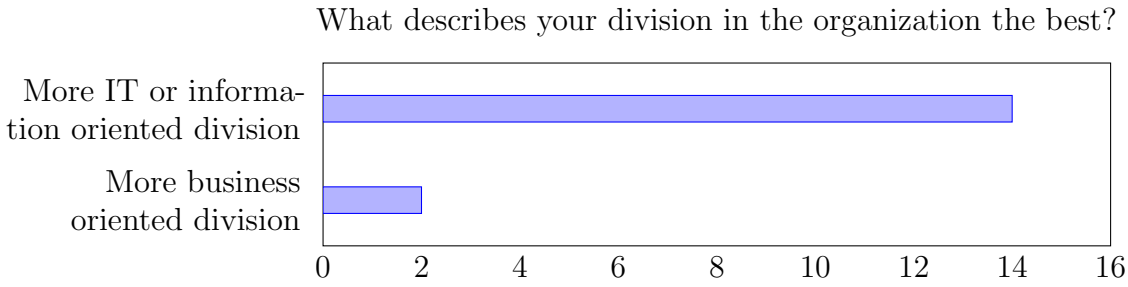



Figure 11: Respondents' division orientation

Next, the respondents were asked to write down their function titles in their organization. The exact titles are not listed here as this could give away who participated on the surveys. What can be said is that a wide variety of professionals participated from developers, managers, to CxO. Another positive result is that almost all respondents were using the platform(s) for more than two years, had developed more than 5 applications, and weren't part of a pilot project.

Three of the respondents selected that their project was part of a pilot project. This could be a reason to exclude their data from the rest of the analysis. However, as they all had more than two years experience with their project, the decision was made to include them in the results. Therefore, all responses are seen as coming from mature implementations of LCDP and can be seen as reliable from that aspect.

One third of the respondents had no coding experience before their LCDP. Two of these were the same respondents that described their division as more business oriented. This could indicate that there is some proof for citizen development. However, as it is not

clear if these respondents were directly involved in creating the applications this cannot be concluded.

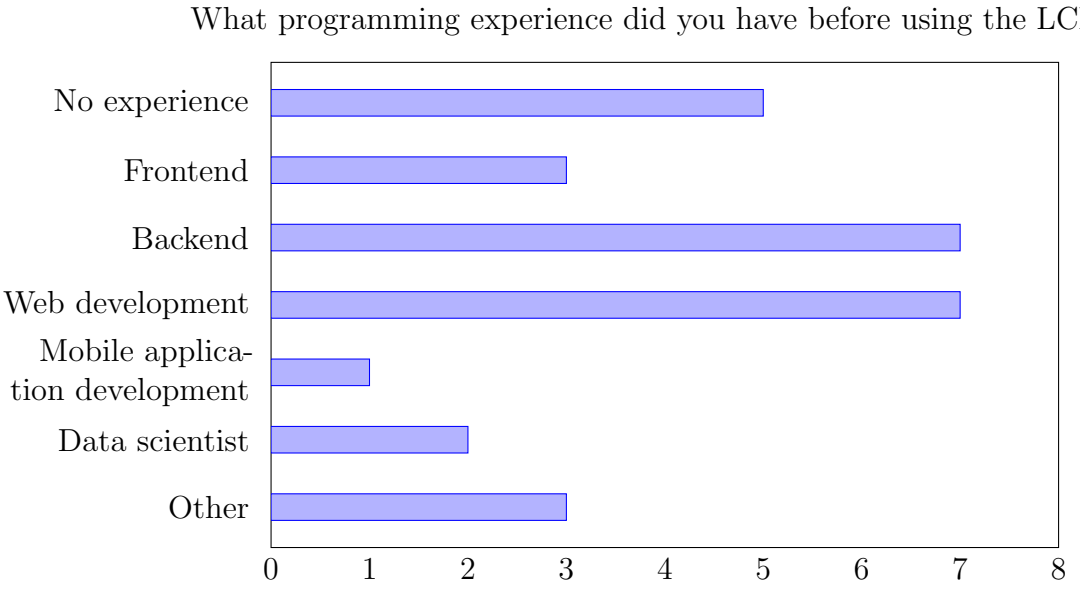



Figure 12: Respondents' coding experience

As predicted, the main LCDP that are in use in the sample are dominated by Mendix and Betty Blocks. Interestingly, only one other LCDP (WEM Modeler) was mentioned besides the four presented options. This could indicate that the variety between LCDPs used in the Dutch public sector is small. However, with only 14 public sector organizations taking part in the survey this is hard to conclude.

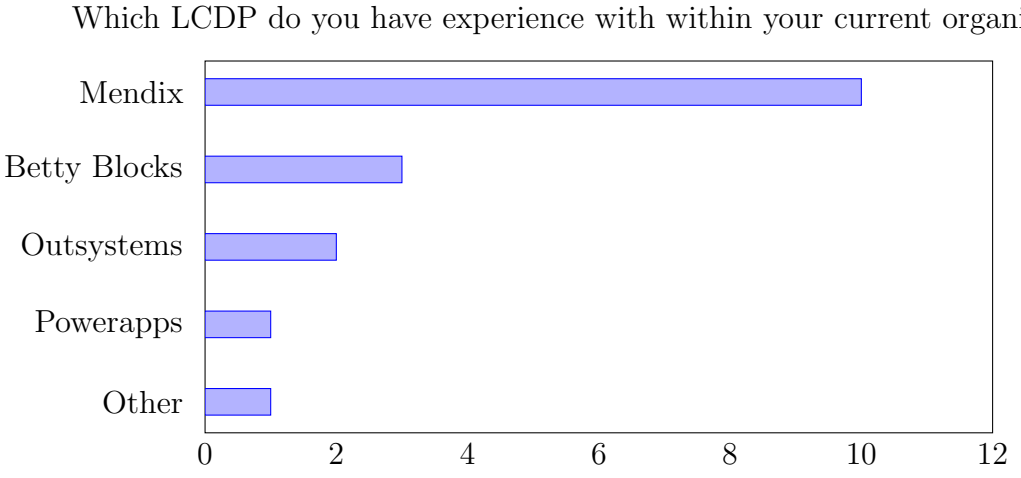



Figure 13: Respondents' LCDP experience

Below are the results for the Likert-type questions and the importance score questions in Table 4 and Table 5 respectively. The sentiment can be read as seen from the perspective of LCDPs. E.g. a positive sentiment in costs means that the median response noted a cost reduction compared to traditional programming approaches.

| | Observed (n = 16) | | | | | | |
|---|---|---|---|---|---|---|---|
| | −− | − | *o* | + | ++ | No opinion | Sentiment |
| Amount of manual coding | 0 | 1 | 2 | 0 | 11 | 2 | ++ |
| Technical interoperability | 0 | 0 | 5 | 7 | 3 | 1 | + |
| Ease-of-mastery | 0 | 1 | 1 | 4 | 7 | 3 | ++ |
| Maintenance | 1 | 1 | 5 | 6 | 2 | 1 | + |
| Business-IT alignment | 0 | 0 | 6 | 2 | 8 | 0 | +/++ |
| Scalability | 2 | 1 | 5 | 5 | 2 | 1 | *o* |
| Knowledge integration | 0 | 0 | 3 | 4 | 4 | 5 | + |
| IT productivity | 0 | 0 | 4 | 4 | 6 | 2 | + |
| Costs | 1 | 1 | 4 | 5 | 3 | 2 | + |
| Business agility | 0 | 0 | 5 | 3 | 8 | 0 | +/++ |
| Vendor lock-in | 3 | 3 | 5 | 3 | 0 | 2 | *o* |
| Customization | 0 | 4 | 4 | 2 | 5 | 1 | *o* |
| Service delivery capability | 2 | 2 | 2 | 3 | 6 | 1 | + |
| Co-production capability | 3 | 3 | 2 | 2 | 4 | 2 | *o* |
| Resource acquisition capability | 2 | 2 | 2 | 7 | 2 | 1 | + |
| Engagement capability | 2 | 3 | 2 | 6 | 0 | 3 | *o* |
| Innovation capability | 1 | 2 | 2 | 6 | 5 | 0 | + |

Table 4: Survey Likert-type questions response summarized

| Item | Mean | Std Deviation | n | $p$-value ($\alpha = 0.05$) |
|---|---|---|---|---|
| Amount of manual coding | 8.06 | 1.48 | 16 | $< 0.01$ |
| Technical interoperability | 7.81 | 1.51 | 16 | $< 0.01$ |
| Ease-of-mastery | 7.87 | 1.59 | 15 | $< 0.01$ |
| Maintenance | 8.25 | 1.39 | 16 | $< 0.01$ |
| Business-IT alignment | 8.00 | 1.84 | 16 | $< 0.01$ |
| Scalability | 7.19 | 1.84 | 16 | $< 0.01$ |
| Knowledge integration | 6.50 | 2.29 | 14 | 0.02 |
| IT productivity | 7.94 | 1.38 | 16 | $< 0.01$ |
| Costs | 7.93 | 1.44 | 15 | $< 0.01$ |
| Business agility | 8.13 | 1.32 | 16 | $< 0.01$ |
| Vendor lock-in | 6.80 | 1.68 | 15 | $< 0.01$ |
| Customization | 6.93 | 1.84 | 15 | $< 0.01$ |
| Service delivery capability | 8.27 | 1.12 | 15 | $< 0.01$ |
| Co-production capability | 5.80 | 3.04 | 15 | 0.31 |
| Resource acquisition capability | 7.13 | 2.33 | 15 | $< 0.01$ |
| Engagement capability | 5.64 | 2.84 | 14 | 0.41 |
| Innovation capability | 7.88 | 1.41 | 16 | $< 0.01$ |

Table 5: Survey item importance score questions response summarized

## 6.2 Qualitative results

As discussed in the method it is important that when quotes are used they should represent the overall sentiment of the dataset. Showing deviant cases is great, but should be used with care as these can be used to change the narrative of the sample. However, as the sample size is small and the respondents who explained every answer is even smaller this is difficult. Especially as those that did use the comment box were often motivated by unhappiness about a certain aspect of their project. Therefore, each item presented will be accompanied by their Likert and importance scores to show the general sentiment of the dataset. The LCDP vendors in quotes have been omitted to keep the respondents anonymous.

The first question was about the amount of manual coding needed to get an application to their liking compared to traditional programming methods. The question could be answered in a Likert 5-point scale, however due to the comparison to traditional programming methods there was little expectation to get negative responses. As this would mean there is more manual code needed than coding everything manually. As expected following the expert interviews, this saw a pretty clear sentiment towards way less manual coding needed. This question was rated important with a significant mean of 8.1. Interestingly, there was one respondent that chose for more manual coding needed. His explanation for this answer was that there were so many requested features that were not supported by their LCDP that they would have been better of coding everything manually.

The second question was about the interoperability of the LCDPs with existing IT systems compared to traditional programming methods. The sentiment on this question was overall positive, with a couple of neutral responses. Most comments mention that the interoperability is highly dependent on the support for API (technology that facilitates exchanging messages or data between two or more different software applications [66]) technology of the existing IT sytems. Some have success and an easy integration process using REST, others have more difficult times when the legacy application do not have this supported. This question was rated very important with a significant mean of 7.8.

> *"In <LCDP>, connectors are built by various parties to integrate with existing systems. When these are sufficient then it is much easier to integrate. However, when you want to do something that is not in the connector then I find it comparatively more difficult to get it right than when an application is built by hand." - Respondent 1*

The next question was about the ease-of-mastery of their LCDP compared to traditional programming methods. The respondents rated this aspect as important with a mean of 7.9. The sentiment was predominantly positive with one negative response. Like the interviewed experts, two respondents noted that it can be easy to master, but only if you have prior experience with basic application development. The one negative response is interesting, as it shows some deeper insight into practical challenges that might occur.

> *"Creating simple screens and logic within the standard functionality is easy*

*to learn. However, we also had to create a lot of customization, as extensions within the <LCDP>. This turned out to be quite poorly documented and even took experienced programmers quite some time to figure out. They found out how to solve these problems only through trial-and-error." - Respondent 2*

The maintenance aspect of LCDP compared to traditional programming methods saw a positive sentiment in general. It was rated highest in importance with a mean of 8.3. Most of the comments were positive ranging from the maintenance is different but equal in workload to less maintenance. Two comments stood out as providing additional insights in this area. The first quote below is from someone who has experienced less maintenance required and the second one from someone who experiences more maintenance.

*"In particular, the technical must-haves do not require maintenance, the platform takes care of that. The functionalities still need to be maintained, but because it is no high code you can get insight into someone's configuration faster. Especially when you use conventions like ours." - Respondent 3*

*"I think this is what disappointed me the most. Creating a first version of the application was relatively quick. Faster than traditional programming if you stick to the standard functionalities of the product. But changes became more and more time-consuming and it became more and more difficult to maintain the right quality. <LCDP> doesn't have good version control, so collaborating within a team also often proved difficult." - Respondent 2*

The first quote suggests using coding conventions as a best practice could help utilize the more visual nature of these platforms for easier maintenance. It also highlights like others have that the maintenance reduction comes partly from the technical dependencies being taken care of by the LCDP. There were also hints that the faster development speed can pose a danger to the continuity of the organization's IT department if not well managed. While the overall sentiment is that the maintenance is reduced, if an organization can not achieve this reduction, the sudden increase in applications can cause a maintenance crisis for the IT department. The lack of a good version control in their LCDP is also interesting, as this is generally easy to do when using traditional programming methods.

Next the item of business-IT alignment was questioned. On the Likert-type question this had a positive to very positive sentiment. However, there were also many neutral responses. Business-IT alignment was considered an important aspect with a mean of 8.0. Notes are made on better cooperation with the business through fast prototyping, fast cyclical development, and less technical developers that have more affinity with the business. However, many comments were also made on how their LCDP did not change their business-IT alignment at all.

Scalability received a neutral score from the respondents, with a mean of 7.2. Many respondents noted that they did not consider the scalability superior to other cloud based solutions or solutions with a good architecture. One noted that scalability was not an option due to the license fees of their LCDP. Two organizations experienced issues with scalability seen in the quote below.

*"<LCDP> has scalability issues in our environments, which manifest themselves in stability and performance issues." - Respondent 4*

After scalability the item of knowledge integration was asked. Knowledge integration was defined as 'identifying and applying specialized knowledge that resides in different parts of an organization'. This topic saw a considerably lower importance score with a mean of 6.5. Five of the sixteen respondents also chose for the 'no opinion' option in the Likert scale. Therefore the positive sentiment seen in Figure4 can be considered weakly supported. The comments give no further insight into the reasons for these scores.

Next is the important item of IT productivity compared to traditional programming methods. The sentiment on this item was positive with an importance score with a mean of 7.9. Some noted that this was a hard question to answer as it is hard to compare their IT productivity for their developed applications with other development methods that they did not use. The respondents that had difficulties with maintenance asked in one of the previous questions were also negative here. Their IT productivity stayed the same even while agreeing with faster development as explained in the quote below.

*"Often with small applications or in the first few weeks you see that development can be done extremely fast and reliable, with which the productivity is very high. However, when the applications become larger and more complex I often see technical debt forming that is expensive to solve. This happens earlier and is worse compared with traditional programming methods." - Respondent 1*

The cost item had a positive sentiment in general. The importance score was a 7.9. The comments received on this question were very mixed. Some noted that the licensing fees were high compared to traditionally developed applications running in the cloud as seen in the quote below. An other respondent noted that the costs were not necessarily lower, but that these costs seemed to shift from development to maintenance.

*"The reason we stopped using <LCDP> is that the licensing model was very expensive. Our prototype would cost almost 10,000 euros per year in licenses, while a similar application developed manually and running on the Microsoft Azure platform would cost less than 1,500 euros per year." - Respondent 1*

*"It certainly is cheaper, but that is mainly because we have a covering agreement (mantelovereenkomst) with our <LCDP>." - Respondent 5*

Next is the item of business agility achieved with LCDPs compared to traditional programming methods. In the survey it got a positive to very positive sentiment with an importance score of 8.1. Overall the comments are positive. Two respondents note that the reusing of modules helps them realize better business agility. Another respondent wholeheartedly agrees as seen in the quote below.

*"Business agility and flexibility is what low-code is meant for and it certainly delivers on that. In a matter of months, we have realized applications*

*that we could never have imagined. They generate enormous value for the organization." - Respondent 3*

Interestingly, the respondent from the previous cost section also shows how the high costs was preventing them from achieving the business agility they wanted.

> *"From a technical perspective, a low code platform allows you to respond very quickly to questions and developments from the market. For example, we had an issue where it had to be possible for municipalities and healthcare providers to provide data about various compensation schemes for corona. Cost-wise, however, it turned out again and again that, due to the expensive licenses, no business case could be made." - Respondent 1*

In addition to the researched advantages of LCDPs there were also two questions on mentioned disadvantages. The first one being the added risk on vendor lock-in. Overall this question received a neutral sentiment with six respondents going towards the negative side and two the positive side. The respondents saying LCDPs had a more severe risk of vendor lock-in mostly mentioned the inability to export or translate developed applications to a new development environment. The neutral respondents noted that any development method comes with lock-in of some sorts and it is not worse for LCDP. Vendor lock-in received an importance score of 6.8.

Another researched disadvantage of LCDPs is less options for customizing your applications. This item had an importance score of 6.9. Although there was an expected negative response, only four respondents picked less customization achievable. The overall sentiment was neutral with seven respondents picking more customization achievable. This is interesting as intuitively it is hard to be able to customize more than is possible using high-code solutions. Part of the reason might be that the question was worded as 'how much customization is achievable' and not how much customization is possible. This confusion can be seen in one of the comments below. This could mean that the answers on this questions are not accurate.

> *"Customization possible within the standard functionality of the platform? In that case, lower than expected. Or possibilities for customizations using programming to add additional functionality? In that case, it is possible but took a lot of time and figuring out." - Respondent 2*

The final question of this survey page asked the participants about any (dis)advantages of LCDPs that had not been discussed yet. Mentioned here are a lack of LCDP developers worldwide, possible (cyber) risks, LCDP being an enabler for agile and devops principles, and the transformational tendency of LCDP. This last item was also touched upon by two respondents quoted below.

> *"All traditional disciplines are necessary in applying Low- and No Code platforms. The dynamics of this collaboration creates an organizational and cultural change that cannot be underestimated." - Respondent 6*

> *"Disadvantages: it is more complex, you hand over your continuity to the*

*cloudvendor, which is often not that great, unless you have an on prem version which is very laborious and expensive and where we lag behind the corporate cloud versions by several months. It is an expensive platform, which only starts to pay off when scaled up and widely used. Setting up libraries, knowledge, culture, app management etc. is intensive. The entire ICT and support organization must be included in the transition, processes must be adapted to cash in on the speed gains. It is a major driver for innovation and change." - Respondent 7*

*"Advantages: short cycle, build lightweight applications. Disadvantage: when these applications come under long term management and maintenance, it turns out that you still have a high-code environment in your organization..." - Respondent 4*

After the page on individual LCDP aspects, the five organizational capabilities from the conceptual framework were tested. First was the item of service delivery capability: The ability to deliver the maximum possible outcome of public services with as limited resources given by the public as possible. The sentiment for this item in the survey was positive with an importance score of 8.3. This was the highest mean and lowest standard deviation in the entire survey. No new information in this was found in the comments.

Secondly, there was the item of public-private cooperation, called co-production in the conceptual framework. This received a neutral sentiment with a wide range of opinions. The importance score averaged on 5.8 with a relatively high deviation. With these results the importance score for this item is statistically insignificant. This sentiment is reflected in the received comments. Three respondents note that they do not see a link between the public-private cooperation and LCDPs.

The resource acquisition capability was perceived to have improved with LCDPs. The importance of this organizational capability was scored a 7.1. One respondent noted that program managers and roles responsible for budget allocation were found inadequately aware of how LCDPs operate. This resulted in difficulties as the demands for projects using LCDPs were not understood. Three other respondents noted three different LCDPs performance aspects that helped them launch initiatives more easily: lower maintenance, lower costs, and a fast proof of concept.

The next question was on the capability of involving citizens in the creation and/or execution of public policy through the usage of a LCDP. Overall this received a neutral sentiment. The importance score had a mean of 5.6 with high deviation. The score had a statistically insignificant results similar to co-production. The comments in the survey supported this finding, where three stated that the LCDP made no difference in this aspect.

The last organizational capability that was asked about was innovation capability. This received a positive sentiment with an importance score of 7.9. One respondent noted that the experimentation with limited resources and subsequent continued development lead to more innovation. Another would agree with an increase in innovation if it weren't

for the fact that a large portion of the applications got rejected, because they are too expensive to bring operational.

The final question of the survey asked the respondents to discuss the LCDPs and their possible effects on public value. Examples given for public value were efficient use of budget, integrity, transparency, responsibility towards the citizen, respect of privacy, and equal treatment of requests. Four respondents noted having trouble attributing public value increase following their LCDP usage. Two others only agreed on efficient use of public funds, saying that any other public value could also be reached using other development methods with similar results. One respondent mentions that open source can be a game changer in this aspect. They are willing to share their developed models to other organizations.

> *"Efficient use of resources is the most important driver for us, the other effects can also be achieved with traditional programming or with the purchase of standard packages." - Respondent 8*

On the flip side, seven respondents reacted more positively. Three mention a faster delivery of applications following business demand. Three note a better ability to deliver customized applications. Efficient usage of budget is agreed on twice, making it a total of four times mentioned.

> *"The application of no and low code platforms offers room to work with more people and less costs on a (more) digital government." - Respondent 6*

# 7 Conclusion

For this thesis an exploratory sequential mixed study was performed to study if LCDPs contribute to a more effective use of IT in the Dutch public sector. For this research expert interviews were held and a survey was performed. In this final chapter the outcomes of this study will be summarized and the research questions answered. Afterwards, these outcomes are discussed and other interesting findings are expanded upon in the discussion section. Next the limitations of this study are outlined and finally recommendations for future research are given.

## Guiding question 1: How can the effectiveness of LCDPs in the public sector best be quantified?

Measuring the impact of technology changes in an organization is a complicated task for multiple reasons. It is difficult to put a boundary on where the impact of the technology ends, the areas impacted are numerous, and these impacts are hard to distinguish from other organizational changes. The 'demarcation' of LCDPs in the public sector therefore presents some limitations. One research area that can enable practical quantification of the outcomes of public sector work, is that of public value. This theory presents broad citizen-centric metrics that encompass more or all citizens. These metrics change the focus from internal efficiency metrics towards external value generation processes.

In order to quantify the effectiveness an approach was chosen that looks at these public values, but also the process leading up to their creation. Inspiration was taken from the theoretical framework for IT Value from Pang et. al [71]. The final developed framework maps performance aspects of LCDPs to organizational capabilities that are important for digitalization and links these to public values. The initial conceptual framework was based on literature and was iteratively checked and improved by six domain experts. Many of the initial items were seen as coming from marketing and were removed or reworded to more unambiguously and accurately worded items. The most important comments of these experts are summarized in chapter 5. The conceptual framework for LCDPs can be seen in Figure 10. The full conceptual framework including the underlying performance aspects can be seen in Appendix B.

## Guiding question 2: What value do LCDPs currently bring in the public sector?

To answer this question, the conceptual framework in guiding question 1 was used to create a survey that was filled in by 16 officials from 14 Dutch public sector organizations. In the survey, all items from the conceptual framework (Figure 10) were asked as well as seven performance aspects that were considered most important or controversial by the interviewed experts. Not all researched performance aspects were asked in the interest of keeping the survey short (under 15 minutes). In the survey, the respondents were asked to judge these aspects in comparison to traditional programming methods. The following

advantages were confirmed in the survey: Reduction in amount of manual coding, improved technical interoperability, easier to master development environment, increased business-IT alignment, improved knowledge integration, increased IT productivity, reduction in costs, improved business agility. When asked about the organizational capabilities important for digitalization, they saw improvements from using LCDPs in improved service delivery capability, improved resource acquisition capability, and improved innovation capability. Finally, the respondents where asked what this can ultimately improve from a public value point of view. Using the taxonomy from Bannister & Connolly [5] the answers included three service oriented values: responsiveness, efficiency, effectiveness, one duty oriented value of efficient use of public funds, and no socially oriented public value. The full quantitative outcomes can be seen in Table 4. The full qualitative results can be seen in section 6.2.

## Research question: Do LCDPs contribute to a more effective use of IT in the public sector?

With the two guiding questions answered, it is now possible to answer the overarching research question of this study. To measure the impact of LCDP in public sector organizations, a conceptual framework was developed from literature and reviewed iteratively by domain experts. In the last interview cycle, the experts agreed on fourteen possible LCDP advantages. Due to considerable overlap they were grouped into five advantage groups as seen in Table 3. The organizational capabilities and public value categories from this framework saw some discussion but were ultimately agreed upon. The domain experts had less experience on this aspect and therefore the validity of these areas from the experts side is less than the LCDP performance aspects and advantage groups.

Next, to test this in practice, guiding question 2 was answered with a survey among 16 public officials experienced with LCDPs. In the survey, all Likert-type questions were put as a comparison to traditional programming methods. Therefore, all received positive sentiments from Table 4 can be seen as an indication of a more effective use of IT. Besides the sentiment scores, there were also questions about the perceived importance of each item in the survey. These scores, summarized in Table 5, can be used to identify what items are deemed necessary for effective IT in the Dutch public sector. Ultimately, the public officials know best what is important for their organization. Using that reasoning, the following LCDP performance metrics are judged as both important and an improvement from traditional programming methods: reduction in amount of manual coding, improved technical interoperability, easier to master development environment, increased business-IT alignment, improved knowledge integration, increased IT productivity, reduction in costs, improved business agility. Improved scalability received a positive importance score of 7.2, however it received only a neutral sentiment and is thus rejected.

Only three of the five listed organizational capabilities were judged significantly important by the public officials in the survey. These were service delivery capability, resource

acquisition capability, and innovation capability. All of these also received a positive sentiment. Service delivery capability and innovation capability were seen as most important with a low deviation. The positive sentiment and some examples from the qualitative part also made it realistic that these are getting improved by LCDPs. *"Business agility and flexibility is what low-code is meant for and it certainly delivers on that. In a matter of months, we have realized applications that we could never have imagined. They generate enormous value for the organization."* - Respondent 3.

In the end, using the outcomes of both guiding questions, the expert interviews, and the survey, we can conclude that LCDPs contribute to a more effective use of IT in the Dutch public sector for the sample researched. This is achieved through a combination of LCDPs advantages over traditional programming methods which improve the organizational capabilities important for digitalization. These may in turn lead to improvements in duty oriented and service oriented public values, but the evidence of this for the researched sample is inadequate. More on this can be found in the discussion in the next section.

## 7.1 Discussion

This research provided interesting observations on several topics such as citizen developers, LCDP app sharing marketplaces, and the efficiency drive versus public values perception by officials. Each of these observations will be outlined and discussed in this section as well as a few general remarks about the research.

When conducting and analyzing the expert interviews, the need for a more academic look at the performance aspects of LCDPs became clear. Many of the advantages or disadvantages found online were seen as marketing terms and were changed or removed by the experts. Complexity reduction, process automation, variable cost, escaping legacy debt, and reducing dependency on hard-to-hire skills are all examples of this. Items that were accepted but were met with doubt were cost reduction, easier maintenance, business-it alignment, and technical interoperability. These were therefore put as question in the survey and were confirmed by the respondents.

Another major change compared to the initial conceptual framework was the change of public-value categories. The initial taxonomy and its KPI's were found to have limitations according to the experts. The category names were found to not fully convey all items that are in their categories. As an example, the public value category of improved administrative efficiency contains the items greater fairness, honesty, and equality. As this created confusion during the interviews a different taxonomy was chosen. Compared to the initial framework, new LCDPs advantages were also added. These were visual interface, platform as a service, real-time editing, and ease-of-mastery. Interestingly, these items did not introduce a completely new subject or angle, but rather encompass several items from the initial framework phrased in a more concise, unambiguous way.

Another interesting view from the experts interviewed was on citizen developers. The concept itself was praised by experts, but no proof of it actually working was found. Most argued that the formal and abstract way of thinking that is required is hard to learn. This finding is supported by a KPMG report for the States General of the Netherlands [46]. A consensus between experts was reached that analytically minded people could be able to become productive citizen developers after some training. Due to this consensus and the lack of experience with citizen developers as intended, the question on this topic in the survey was left out. Originally, there was a question asking respondents if they have citizen developers in their organization or are planning to. However, as the survey was already considered too long, this question was left out. Preference was instead given to topics that were disputed between the interviewed experts.

In the quantitative side of the survey there were mostly positive responses on the LCDP performance aspects. Only scalability, vendor lock-in and customization saw a neutral response. For scalability, the respondents of two organizations noted issues in their environments. Most other respondents consider it on-par to other cloud bases solutions or other solutions with good architecture. Another reason for poor scalability could be that most public sector organizations use on-premise versions that naturally scale harder than the cloud versions. The reasons for using on-premise versions are usually

related to GDPR and security concerns. For vendor lock-in and customization, the neutral response is surprising, as most papers put these forward as the main limitations of LCDPs and thus a negative sentiment was expected. Especially, the apparent lack of consideration for vendor lock-in is interesting. These results can be interpreted as part of an ongoing trend in which public organizations seem to focus too much on efficiency and cost reduction [30]. Failing to see the risks posed by the shifting of power which some call 'platform capitalism'. The dangers of platform capitalism can be explained as: "Through its emphasis on transformation, innovativeness and opportunity, platform capitalism obfuscates the politics and power relations hidden behind the concepts of crowdsourcing, gig economy, sharing economy and platform economy" [49].

In the qualitative side of the survey some points are worthy of a highlight. One such highlights in the qualitative section was on maintenance. While the majority agrees on easier maintenance, two respondents indicated major issues. A best practice found in the comments was that enforcing proper coding guidelines, just as in traditional development methods, was key to get positive results on this aspect. Some comments on costs were also insightful. Generally, LCDPs are advertised as cost savers. However, some highlighted exorbitant costs due to the previously described need for on-premise versions. For most LCDPs, going on-premise means going for the more expensive packages. Two respondents noted that the only reason they achieved cost savings was through framework agreements and covering agreements. Another reason why the majority does not seem to have this issue could be that they have long running contracts from when prices were lower and the Dutch LCDP vendors were looking to increase their legitimacy through partnerships with the Dutch public sector.

One of the possible LCDP advantages that was not looked at in this research is that of the app sharing marketplaces. On these marketplaces applications can be standardized and uploaded to share with others in a semi open source way. Semi open source as the licensing fees for the LCDP still apply. Open source is an item which is seen as an important aspect for the digital future in the public sector [45]. LCDPs can be both an opportunity and a challenge in this aspect. Within most LCDPs developed applications can be easily shared with other organizations. Some platforms, like Appsemble seem to have this as their main purpose. Due to the easier deployment and maintenance of LCDPs this could be a major efficiency driver organizations, especially smaller public sector organizations. In its current form, the respondents of the survey did not note LCDPs as a higher vendor lock-in risk than traditional programming methods. When choosing to share these applications with other public sector organizations however, this risk is substantially increased. As seen in the survey, the licensing fees can become so high that developing the application traditionally can be a lot cheaper. Perhaps this is why there was no evidence of application sharing via LCDPs found yet.

Another reason why no cases of this were observed could be explained by other research on WiGo4it [50]. WiGo4it focused on the development of a common, future-proof IT solution for several participating municipalities on the topics of income, labor, and healthcare. For the development of its first online form, the holiday form (vakantieformulier), the

parties decided to work towards an ideal form. This ignored current processes and policy at the municipalities. Only after the development had been completed and the form was ready to be used did the flaws of this approach came to light. To quote the author who conducted a case study on this topic, *"The further development of the form became trench warfare. In future development of e-services, policy-sensitive topics were avoided"* [50]. WiGo4it is not the only initiative of this kind. Dimpact and the GovUnited foundation were other attempts to standardize municipality's IT which all seem to under deliver. In the paper, the researcher concludes that the lesson learned from this case should be that current processes, policy, and IT solutions should be looked at holistically when working toward one IT solution. This same principle, working towards an ideal form and not looking at a holistic IT solution, would apply to these marketplaces as well. The applications would be standardized, but still be made according to the processes of that organization. For these reasons it appears unlikely that the app-sharing marketplaces of LCDPs will deliver positive results for the public sector and the open source movement.

One of the side results of this study is that the concept of public value and its different aspects received very little interest by the surveyed officials. The ability to improve private-public collaboration and the ability to involve citizens in the formulation and implementation of public policy received a comparatively low importance score. Four of the respondents noted in the final question that they did not see the relationship of the public values listed in the survey at all. In total, only responsiveness, efficiency, effectiveness and efficient use of public funds were found in the responses. This seems to indicate that public officials focus more on the efficiency drive of their internal organization and give little attention to public values.

Multiple reasons for this apparent lack of focus on the generation of public values can be theorized. First of all, it could be that most digitalization efforts are focused on internal processes (overhead) instead of the primary processes that are more citizen-focused and thus involve public values directly. Especially HR and finance were mentioned as the main digitalization domains at the moment. As this was not asked in the survey, it is not possible to confirm this theory. Another reason could be that by the time an IT technology like a LCDP gets involved, the public policy is already fully made. Meaning that there is little opportunity for the project team to think about public value aspects related to the IT technology.

A third reason could be that the ever increasing ICT costs per citizen forces departments to focus only on efficiency focused metrics. For municipalities, according to a benchmark by M&I partners, the costs of ICT per citizen rose from 79 euro's in 2016 to 93 euro's in 2020 [62]. Other private research also concluded that municipalities primarily make IT related decisions based on costs, not value delivered by IT [38]. They state an urgent need to make value generated by IT more visible. Interestingly, The Association of Netherlands Municipalities (VNG) does seem to have some focus on public values. During this research several public values were found for the Common Ground project (a new growing initiative from the VNG to reform and modernize the municipal information sharing infrastructure [19]), it mentions "this enables municipalities to organize people-

oriented, low regulation, clear, fast and certain appropriate services" [19]. It also notes better citizen engagement and cost savings (in healthcare). Knowing this, it could also be the case that in this small sample simply no one was responsible for considering public values in their projects. However, even if the policy was already made there is still room for interpretation. Which in turn could be used to consider improvements in public value.

Looking back on the research approach, some subjects need to be discussed. One of the big choices early on in the study was to use public value theory to evaluate a technology change in the public sector. As mentioned in the conclusion, this theory presents broad citizen-centric metrics that differ by shifting the focus from internal efficiency metrics towards external value generation processes. In the interviews and other casual conversations had with experts, there was great interest in the potential of this theory. However, there were several disagreements from the experts as well. The taxonomies chosen saw some critique and the mapping of the LCDP performance metrics or organizational capabilities to these public value categories were considered only moderately related. Another problem with this approach is that the public officials had difficulties thinking in public values or downright disagreed with any relation between them and IT changes. This begs the question whether this approach using public value should be repeated in this form.

With the current results, one has to conclude that measuring the effects of IT change using public value did not yield many usable results. Using the LCDPs performance aspects and organizational capabilities gave far more insights into the impact (good or bad) that LCDPs can make. As earlier described, factors like the overhead/primary process automation differences or the fact that only two out of the sixteen respondents came from a business oriented division could have limited the effectiveness of the public value measurements. In the end, one has to conclude that even though the theory gathers a lot of interest from outside experts and the aspirations of citizen-centric measurements are admirable, the surveyed public officials themselves seem not to think in these values while working on IT projects. Therefore, the public value mapping in its current form seems inadequate to measure the effects of IT change in a public organization for this sample.

## 7.2 Limitations

One of the main limitations of this research is that the survey sample size was low. As the survey was anonymous, it is hard to figure out why the response rate was so low. Interestingly, the survey platform Qualtrics reported 28 cases where people clicked the survey link, but did not go to the next page (where the first questions of the survey start). The average time these people spent on the introduction page is, with the removal of two outliers, 40 seconds with a median of 27 seconds. This seems to suggest that these people read the full introductory text of the first page and stopped afterwards. It is interesting why so many stopped the survey there as survey was sent out in a targeted manner. Reasons for this could be that the survey was deemed too long (10-15 minutes), they did not agree on the survey vision/goals, it was not clear enough if they could participate, or they did not think their LCDP was part of the target group. Another reason could be the cooperation with Capgemini. As the survey was anonymous it is unfortunately not possible to ask why they decided not to continue the survey.

The smaller sample meant that only very conservative statistical measures could be used to reach conclusions. Another possible bias in the survey is the central tendency bias due to the usage of the 5-point Likert scale with no opinion option [27]. As discussed earlier, the survey results were positive, sometimes even when a negative sentiment was expected. One of the contributing factors could be an acquiescence response bias, which means that respondents can lean towards agreeing with questions in a survey. This could explain why the median ends up being neutral in the case of vendor lock-in and customization in Table 4.

Another aspect of this research that needs to be discussed is the cooperation with Capgemini. Although very fruitful, it does introduce some bias in this study. First, the experts interviewed were handpicked by the in-company supervisor. Although several criteria were given on the nature of these experts, they could have been chosen favorably. Second, it is highly likely that most of the respondents in the survey were or are currently Capgemini clients. This selection bias means that the (positive) results of the survey are possibly not representative of the average LCDP project. Similarly, the dominance of Mendix in this sample means that the outcomes might not be representative of the overall market composition of the Dutch public sector LCDP market. No public information on these market shares was found to confirm this.

## 7.3 Future work

During the research, several topics were found that would be interesting for future research. First, this study can be used as a starting point to investigate the role LCDPs can or currently play in the public sector. A repeat of the study with a larger and more randomized sample would be a way to make more progress in this area. The current research focused specifically on the Dutch public sector. This raises the question of how well it holds up in other countries or in the private sector. In the expert interviews, one mentioned that, excluding public value, the framework could be reused for all types of organizations.

A result that raised questions was the possible "efficiency drive" found in the survey sample. Although understandable due to budget constraints and increased IT costs, it seems unproductive in the long run to overlook areas such as vendor lock-in, customization, and citizen-centric metrics. Several possible reasons for this discrepancy were coined in the discussion, but none of these could be tested with the gathered data. More research in this area could help to uncover if this is indeed a general theme and what can be done.

The LCDP application marketplace is another interesting area that was left out of this thesis. Their promise is that standardized applications can be uploaded and shared with other organizations. The potential for the public sector with its many overlapping tasks and decentralized IT systems is obvious. However, no evidence of IT transfer was found through an application marketplace between public sector organizations in The Netherlands. Even though they seem to be trying via initiatives like OpenApps. Therefore, the question remains whether these standardized uploaded applications can and will be used in a different public organization that already has its own (differing) processes.

Another area that deserves more research is that of citizen developers. Unfortunately, this study did not have the time to look deeper into this area. Interviewed experts praised the concept, but did not believe the "hype" created by most LCDP vendors. A study looking into the quantities and experiences of citizen developers working in both the public and private sectors would greatly aid this research area. This research could also look at the theory coined in this thesis that citizen developers can come only from trained "analytically minded" people.

# References


[1] Claire Anderson. "Presenting and Evaluating Qualitative Research". In: *American Journal of Pharmaceutical Education* 74.8 (2010). ISSN: 15536467. DOI: 10.5688/AJ7408141. URL: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2987281/.

[2] Anggi Ratna Anggraini and J. Oliver. "Selection of low-code plataforms based on organization and application type". In: *Journal of Chemical Information and Modeling* 53.9 (2019), pp. 1689–1699. URL: https://lutpub.lut.fi/bitstream/handle/10024/158389/selection%20of%20low-code%20platforms%20based%20on%20organization%20and%20application%20type.pdf?sequence=1&isallowed=y.

[3] Appian. *Appian Announces Fourth Quarter and Full Year 2020 Financial Results — Appian Corporation - IR site*. URL: https://investors.appian.com/news-releases/news-release-details/appian-announces-fourth-quarter-and-full-year-2020-financial (visited on 10/15/2021).

[4] Appsemble. *Apps · Appsemble*. URL: https://appsemble.app/en/apps (visited on 10/15/2021).

[5] Frank Bannister and Regina Connolly. "ICT, public values and transformative government: A framework and programme for research". In: *Government Information Quarterly* 31.1 (2014). ISSN: 0740624X. DOI: 10.1016/j.giq.2013.06.002.

[6] Betty Blocks. *Developing in Betty Blocks*. URL: https://www.bettyblocks.com/evaluation-guide/developing-in-betty-blocks (visited on 01/12/2022).

[7] Betty Blocks. *No-Code Platform Betty Blocks Raises $33 Million to Empower a New Breed of Developer in Enterprise*. URL: https://blog.bettyblocks.com/no-code-platform-betty-blocks-raises-33-million-to-empower-a-new-breed-of-developer-in-enterprise (visited on 10/19/2021).

[8] Michael Bloch, Sven Blumberg, and Jürgen Laartz. "Delivering large-scale IT projects on time, on budget, and on value". In: *Harvard Business Review* (2012), pp. 2–7.

[9] Harry N. Boone and Deborah A. Boone. "Analyzing Likert data". In: *Journal of Extension* 50.2 (2012). ISSN: 10775315.

[10] Barry Bozeman. "Technology transfer and public policy: A review of research and theory". In: *Research Policy* 29.4-5 (2000). ISSN: 00487333. DOI: 10.1016/S0048-7333(99)00093-1.

[11] Barry Bozeman, Heather Rimes, and Jan Youtie. "The evolving state-of-the-art in technology transfer research: Revisiting the contingent effectiveness model". In: *Research Policy* 44.1 (2015). ISSN: 00487333. DOI: 10.1016/j.respol.2014.06.008.

[12] Marco Brambilla, Jordi Cabot, and Manuel Wimmer. "Model-Driven Software Engineering in Practice". In: *Synthesis Lectures on Software Engineering* 1.1 (2012). ISSN: 2328-3319. DOI: 10.2200/s00441ed1v01y201208swe001.

[13] Wouter Laurentius Bronsgeest. "Meer vorm dan inhoud: onderzoek naar evaluaties van ICT-projecten bij de overheid". PhD thesis. Enschede, The Netherlands: University of Twente, Nov. 2016. ISBN: 9789462334083. DOI: 10.3990/1.9789462334083.

[14] F Brooks and H Kugler. *No silver bullet*. University of North Carolina at Chapel Hill, 1987.

[15] Erik Brynjolfsson and Lorin Hitt. "Paradox lost? Firm-level evidence on the returns to information systems spending". In: *Management Science* 42.4 (1996). ISSN: 00251909. DOI: 10.1287/mnsc.42.4.541.

[16] Jordi Cabot. "Positioning of the low-code movement within the field of model-driven engineering". In: *Proceedings - 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems, MODELS-C 2020 - Companion Proceedings*. 2020. DOI: 10.1145/3417990.3420210.

[17] Walter Castelnovo and Massimo Simonetta. "A public value evaluation of e-Government policies". In: *ECIME 2007: European Conference on Information Management and Evaluation*. 2007.

[18] Channele2e. *M&A List: 8 ServiceNow Acquisitions Extend Beyond ITSM Software - ChannelE2E*. URL: https://www.channele2e.com/investors/mergers-acquisitions/list-of-servicenow-acquisitions/ (visited on 10/18/2021).

[19] Common Ground project. *Common Ground: wat, hoe en waarom? · Common Ground*. URL: https://commonground.nl/cms/view/77953921-4161-462c-af24-6ac3cd7d61a7/common-ground-wat-hoe-en-waarom (visited on 11/25/2021).

[20] J Creswell and V Plano Clark. "Designing and Conducting Mixed Methods Research. Thousand Oaks, CA: Sage". In: *Organizational Research Methods* 12.4 (2009). ISSN: 1445-2197.

[21] Daniel Dahlberg. "Developer Experience of a Low-Code Platform : An exploratory study". In: (2020).

[22] Moshe Davidow. "Value Creation and Efficiency: Incompatible or Inseparable?" In: *Journal of Creating Value* 4.1 (2018). ISSN: 2454213X. DOI: 10.1177/2394964318768904.

[23] Annemieke Diekman and Bram Steijn. "Tekort aan ICT'ers - 'Het zal krap blijven'". In: *VNG Magazine, volume 18* (Nov. 2020). URL: https://vng.nl/artikelen/tekort-aan-icters-het-zal-krap-blijven.

[24] Cem Dilmegani, Bengi Korkmaz, and Martin Lundqvist. "Public-sector digitization: The trillion-dollar challenge". In: *McKinsey. com, December* (2014).

[25] Directorate-General for Informatics (European Commission). *New European Interoperability Framework*. Publications Office of the EU, 2017. ISBN: 978-92-79-63756-8. DOI: 10.2799/78681.

[26] Omar A. El Sawy et al. "Seeking the configurations of digital ecodynamics: It takes three to tango". In: *Information Systems Research* 21.4 (2010). ISSN: 15265536. DOI: 10.1287/isre.1100.0326.

[27] N. S. Fagley. "Methodological issues & strategies in clinical research". In: *Psychotherapy Research* 14.1 (2004). ISSN: 1050-3307. DOI: 10.1093/ptr/kph012.

[28] Siamak Farshidi, Slinger Jansen, and Sven Fortuin. "Model-driven development platform selection: four industry case studies". In: *Software and Systems Modeling* (2021), pp. 1–27.

[29] Nicholas Faulkner and Stefan Kaufman. "Avoiding Theoretical Stagnation: A Systematic Review and Framework for Measuring Public Value". In: *Australian Journal of Public Administration* 77.1 (2018), pp. 69–86. ISSN: 14678500. DOI: 10.1111/1467-8500.12251.

[30] Tobias Fiebig et al. "Heads in the Clouds: Measuring the Implications of Universities Migrating to Public Clouds". In: (Apr. 2021). URL: http://arxiv.org/abs/2104.09462.

[31] Meg Fryling. "Low Code App Development". In: *J. Comput. Sci. Coll.* 34.6 (2019), p. 119. ISSN: 1937-4771.

[32] Maurice Gaikema et al. "Increase the success of Governmental IT-projects". In: *Systemics, Cybernetics and Informatics* 17.1 (2019), pp. 97–105. URL: http://www.healthpayrollinquiry.qld.gov.au/__data/assets/pdf_file/0014/207203/Queensland-.

[33] Paolo Gerli et al. "Beyond contact-tracing: The public value of eHealth application in a pandemic". In: *Government Information Quarterly* (2021). ISSN: 0740624X. DOI: 10.1016/j.giq.2021.101581.

[34] Google. *FAQ about Google Trends data - Trends Help*. URL: https://support.google.com/trends/answer/4365533?hl=en-GB&ref_topic=6248052 (visited on 10/18/2021).

[35] Brent Hailpern and Peri Tarr. "Model-driven development: The good, the bad, and the ugly". In: *IBM Systems Journal* 45.3 (2006). ISSN: 00188670. DOI: 10.1147/sj.453.0451.

[36] C Hood. "Hood, C. (1991), 'A public management for all seasons?' Public". In: *Public Administration* 69 (1991). ISSN: 14679299.

[37] SJBA Hoppenbrouwers and P van Bommel. "The selection process of model based platforms". In: (2017).

[38] Mark Huijben, Johan Posseth, and Jeroen Wismans. *Kosten en kansen van ICT in de publieke sector - Platform Overheid*. URL: https://platformoverheid.nl/kosten-en-kansen-van-ict-in-de-publieke-sector/ (visited on 11/25/2021).

[39] John Hutchinson, Jon Whittle, and Mark Rouncefield. "Model-driven engineering practices in industry: Social, organizational and managerial factors that lead to success or failure". In: *Science of Computer Programming* 89.PART B (2014), pp. 144–161. ISSN: 01676423. DOI: 10.1016/j.scico.2013.03.017. URL: http://dx.doi.org/10.1016/j.scico.2013.03.017.

[40] Satu Iho, Désirée Krejci, and Stephanie Missonier. "Supporting Knowledge Integration with Low-Code Development Platforms". In: *ECIS 2021 Research Papers*. Vol. 38. 2021.

[41] Barbora Jancovicova. "Next Generation Methods for Development of Enterprise Information Systems". In: (2019).

[42] Naci Karkin and Marijn Janssen. "Evaluating websites from a public value perspective: A review of Turkish local government websites". In: *International Journal of Information Management* 34.3 (2014). ISSN: 02684012. DOI: 10.1016/j.ijinfomgt.2013.11.004.

[43] Faezeh Khorram, Jean-marie Mottu, and Gerson Sunyé. "Challenges & Opportunities in Low-Code Testing". In: *Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings*. 2020, pp. 1–10. ISBN: 9781450381352.

[44] John Kincaid. " Creating Public Value: Strategic Management in Government. Mark H. Moore ". In: *The Journal of Politics* 59.1 (1997). ISSN: 0022-3816. DOI: 10.2307/2998228.

[45] Raymond Knops. *In gesprek met staatssecretaris Knops over de digitale toekomst*. 2020. URL: https://www.youtube.com/watch?v=upZgdMiWBhI.

[46] KPMG Advisory N.V. *Rapportage Fase 2. Achtergrondstudie Strategisch organisatieadvies SSC-ICT. Deel 1 — Tweede Kamer der Staten-Generaal*. Tech. rep. Mar. 2020, pp. 1–64. URL: https://www.tweedekamer.nl/kamerstukken/detail?id=2020D08398&did=2020D08398.

[47] Jeff Kramer and Orit Hazzan. "Summary of an ICSE 2006 Workshop: The Role of Abstraction in Software Engineering". In: *1st International Workshop on The Role of Abstraction in Software Engineering*. Vol. 31. 2006.

[48] Mary Lebens and Roger Finnegan. "Using a Low Code Development Environment to Teach the Agile Methodology". In: *International Conference on Agile Software Development*. Springer, 2021, pp. 191–199.

[49] Yin Liang, Jeremy Aroles, and Bernd Brandl. "Charting platform capitalism: Definitions, concepts and ideologies". In: *New Technology, Work and Employment* (Feb. 2022). ISSN: 0268-1072. DOI: 10.1111/ntwe.12234. URL: https://onlinelibrary.wiley.com/doi/10.1111/ntwe.12234.

[50] Casper van der Marel. *Leren digitaliseren: e-dienstverlening en WiGo4it*. Erasmus Universiteit, 2009.

[51] Albert Meijer and Victor Bekkers. "A metatheory of e-government: Creating some order in a fragmented research field". In: *Government Information Quarterly* 32.3 (2015). ISSN: 0740624X. DOI: 10.1016/j.giq.2015.04.006.

[52] Mendix. *Developing Applications in Mendix - Collaborative Model-Driven Development — Mendix Evaluation Guide*. URL: https://www.mendix.com/evaluation-guide/app-lifecycle/developing-in-mendix/ (visited on 03/29/2021).

[53] Mendix. *Low-code Application Development Platform - Build Apps Fast & Efficiently — Mendix*. URL: https://www.mendix.com/ (visited on 10/14/2021).

[54] Mendix. *Low-Code vs. No-Code: Differences, Similarities & Use Cases*. URL: https://www.mendix.com/blog/understand-no-code-vs-low-code-development-tools/ (visited on 10/14/2021).

[55] Mendix. *Mendix Fires On All Cylinders; Surpassed $100 Million Annual Recurring Revenue and Now on Trajectory to Double in 18 Months — Mendix*. URL: https:

//www.mendix.com/press/mendix-fires-on-all-cylinders-surpassed-100-million-annual-recurring-revenue-and-now-on-trajectory-to-double-in-18-months/ (visited on 10/14/2021).

[56] Mendix. *SAP Application Development & Integrations with Mendix Low-Code*. URL: https://www.mendix.com/sap/ (visited on 10/14/2021).

[57] Mendix. *Understanding Application Platform as a Service (aPaaS) - Examples & FAQ*. URL: https://www.mendix.com/application-platform-as-a-service/ (visited on 10/18/2021).

[58] Mendix. *What Is Mendix & How Does It Support the App Development Lifecycle?* URL: https://www.mendix.com/evaluation-guide/what-is-mendix/ (visited on 10/14/2021).

[59] Mendix. *What is MXDP? Guide to Multiexperience Development Platforms*. URL: https://www.mendix.com/multiexperience-development-platform/ (visited on 10/14/2021).

[60] Timo Meynhardt. "Public value inside: What is public value creation?" In: *International Journal of Public Administration* 32.3-4 (2009). ISSN: 01900692. DOI: 10.1080/01900690902732632.

[61] Timo Meynhardt et al. "Public Value Performance: What Does It Mean to Create Value in the Public Sector?" In: *Public Sector Management in a Globalized World*. 2017. DOI: 10.1007/978-3-658-16112-5{_}8.

[62] MI Partners. "ICT-Benchmark Gemeenten 2021 conclusies". In: (2021). URL: www.mxi.nl/benchmarks.

[63] Paul Mihas. "Learn to Use an Exploratory Sequential Mixed Method Design for Instrument Development". In: *Learn to Use an Exploratory Sequential Mixed Method Design for Instrument Development* (2019). DOI: 10.4135/9781526496454.

[64] Rodrigo Salvador Monteiro, Roque Elias, and Assumpção Pinel. "The MDArte Experience Organizational Aspects Acquired from a Successful Partnership between Government and Academia using Model-Driven Development". In: *Model-Driven Engineering and Software Development (MODELSWARD), 2014 2nd International Conference on* (2005), pp. 575–586.

[65] Cecilie Ness and Marita Eltvik Hansen. "Potential of low-code in the healthcare sector: an exploratory study of the potential of low-code development in the healthcare sector in Norway". In: (2019).

[66] Isaac Odun-Ayo, Chinonso Okereke, and Hope Orovwode. "Cloud and application programming interface – Issues and developments". In: *Lecture Notes in Engineering and Computer Science*. Vol. 2235. 2018.

[67] OpenApps. *OpenApps*. URL: https://openapps.nl/ (visited on 10/15/2021).

[68] Outsystems. *OutSystems Achieves Record Revenue in 2018 with Successful Customers Fueling Low-Code Market Leadership — OutSystems*. URL: https://www.outsystems.com/news/low-code-market-leadership-record/ (visited on 10/14/2021).

[69] Outsystems. *The State of Application Development - Is IT Ready for Disruption?* 2019.

[70] Panos Panagiotopoulos, Bram Klievink, and Antonio Cordella. "Public value creation in digital government". In: *Government Information Quarterly* 36.4 (2019). ISSN: 0740624X. DOI: 10.1016/j.giq.2019.101421.

[71] Min Seok Pang, Gwanhoo Lee, and William H. Delone. "IT resources, organizational capabilities, and value creation in public sector organisations: A public-value management perspective". In: *Journal of Information Technology* 29.3 (2014), pp. 187–205. ISSN: 14664437. DOI: 10.1057/jit.2014.2.

[72] Axel Reichwein. "Application-specific UML profiles for multidisciplinary product data integration". PhD thesis. Stuttgart University, 2012.

[73] Clay Richardson and John R Rymer. "Vendor Landscape : The Fractured , Fertile Terrain Of Low-Code Application Platforms". In: *Forrester Research* (2016).

[74] Clay Richardson et al. "New Development Platforms Emerge For Customer-Facing Applications". In: *Forrester: Cambridge, MA, USA* (2014), p. 15.

[75] John R. Rymer et al. "The Forrester Wave ™ : Low-Code Development Platforms For AD & D Pros , Q1 2019". In: *Forrester* (2019), pp. 2–16.

[76] John R. Rymer et al. "The Forrester Wave™: Low-Code Platforms For Business Developers". In: *Forrester* (2019).

[77] Apurvanand Sahay et al. "Supporting the understanding and comparison of low-code development platforms". In: *46th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*. December. 2020, pp. 171–178. ISBN: 9781728195322. DOI: 10.1109/SEAA51224.2020.00036.

[78] Raquel Sanchis et al. "Low-code as enabler of digital transformation in manufacturing industry". In: *Applied Sciences* 10.1 (2020), p. 12. ISSN: 20763417. DOI: 10.3390/app10010012.

[79] Servicenow. *2020 Annual Report*. Tech. rep. Delaware, 2020.

[80] Alessandro Spano. "How do we measure public value? from theory to practice". In: *Studies in Public and Non-Profit Governance*. Vol. 3. 2014. DOI: 10.1108/S2051-663020140000003015.

[81] Tweede Kamer der Staten-Generaal. *Parlementair onderzoek naar ICT-projecten bij de overheid, eindrapport (TK 33 326, nr. 5)*. Tech. rep. 2014.

[82] Laela Sturdy. *No-Code Applications Are Shaking Up the Enterprise*. URL: https://capitalg.com/insights/leading-no-code-investor-and-capitalg-partner-laela-sturdy-explains-why-no-code-applications-are/ (visited on 03/08/2022).

[83] Massimo Tisi et al. "Lowcomote: Training the next generation of experts in scalable low-code engineering platforms". In: *STAF 2019 Co-Located Events Joint Proceedings: 1st Junior Researcher Community Event, 2nd International Workshop on Model-Driven Engineering for Design-Runtime Interaction in Complex Systems, and 1st Research Project Showcase Workshop co-located with Soft*. 2019.

[84] Jean Damascene Twizeyimana and Annika Andersson. "The public value of E-Government – A literature review". In: *Government Information Quarterly* 36.2 (2019). ISSN: 0740624X. DOI: 10.1016/j.giq.2019.01.001.

[85] Antonio Vallecillo. "On the Industrial Adoption of Model Driven Engineering. Is your company ready for MDE". In: *International Journal of Information Systems and Software Engineering for Big Companies (IJISEBC)* 1.1 (2014), pp. 52–68.

[86] Colin Van Noordt and Gianluca Misuraca. "Evaluating the impact of artificial intelligence technologies in public services: Towards an assessment framework". In: *ACM International Conference Proceeding Series*. 2020. DOI: 10.1145/3428502.3428504.

[87] Erik Vikebø and Ludvig B Sydvold. "An inquiry into low-code solutions in institutions for higher education: a case study of low-code implementation at the Admissions Office at the Norwegian School of Economics". In: (2019).

[88] Paul Vincent et al. *Licensed for Distribution Magic Quadrant for Enterprise Low-Code Application Platforms*. Tech. rep. Gartner, 2019. URL: https://www.gartner.com/doc/reprints?id=1-1ODOM46A&ct=190812&st=sb.

[89] Paul Vincent et al. *Magic Quadrant for Enterprise Low-Code Application Platforms*. Tech. rep. Gartner, 2021, pp. 1–33.

[90] Paul Vincent et al. *Magic quadrant for enterprise low-code application platforms*. Tech. rep. Gartner, 2020.

[91] Tatu Virta. "Relation of low-code development to standard software development: Case Biit Oy". In: *Master's Thesis* (2018). URL: https://lutpub.lut.fi/bitstream/handle/10024/158441/masters_thesis_virta_tatu.pdf?sequence=1&isAllowed=y.

[92] Michael Wade and John Hulland. "Review: The resource-based view and information systems research: Review, extension, and suggestions for future research". In: *MIS Quarterly: Management Information Systems* 28.1 (2004). ISSN: 02767783. DOI: 10.2307/25148626.

[93] Robert Waszkowski. "Low-code platform for automating business processes in manufacturing". In: *IFAC-PapersOnLine* 52.10 (2019), pp. 376–381.

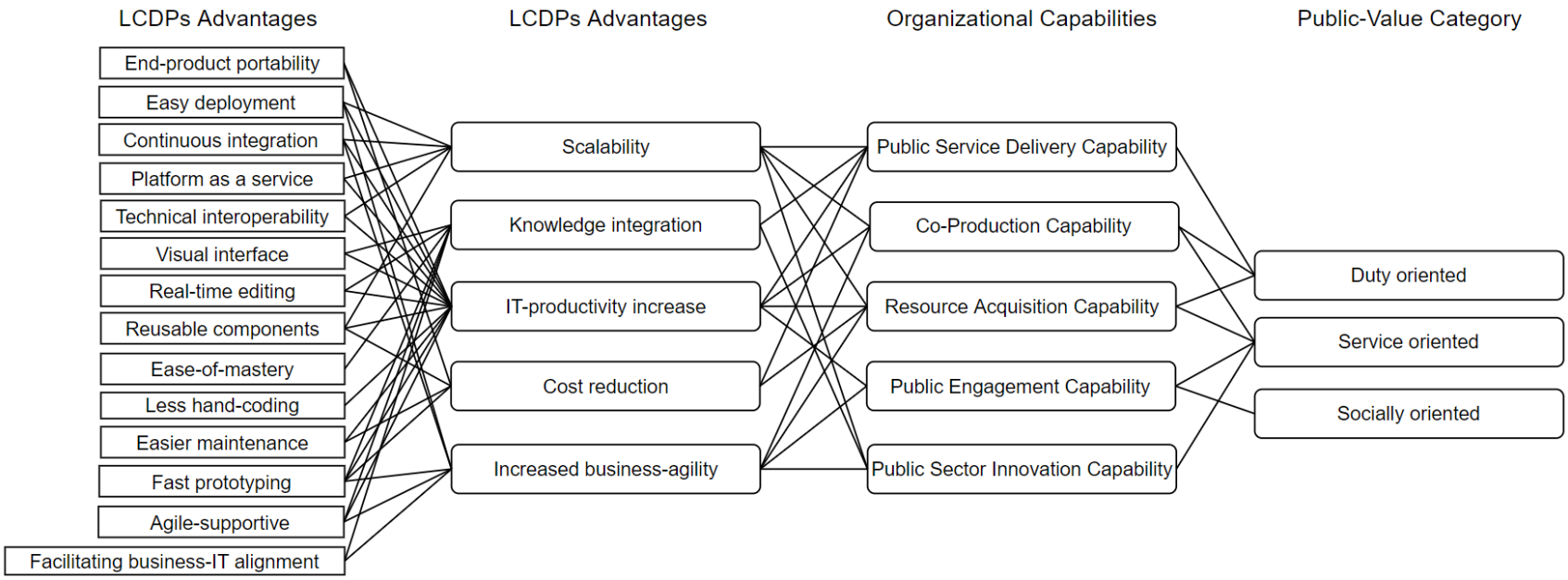
# A Key performance indicator examples for public value

1. Improved Public Services [84]
    - Increased quality of public information and services
    - Enabled transparency, participation, and collaboration in the delivery of Public services
    - Provision of more responsive, efficient, and cost-effective public services
    - Improved access to government information and services
2. Improved Administrative Efficiency
    - Cost-reduction
    - Reduced bottleneck and queues in the delivery of services to citizens
    - Reduced or eliminate the risk of corruption and abuse of the law by public Servants
    - Enabled greater fairness, honesty, equality
3. Open Government (OG) capabilities
    - Increased public/citizens participation in government actions and policy making
    - Improved public engagement and well-informedness
    - Improved partnerships (within government or in the form of public private Partnerships (PPP))
    - Improved political possibilities and innovations
4. Improved Ethical Behavior and Professionalism
    - Demand for good information for decisions
    - Reduction or elimination of the risk of corruption and abuse of the law by public servants
    - Achievement or increased robustness, reliability, security, efficiency and Effectiveness of government
    - Maintenance of accurate durable records
5. Improved Trust and Confidence in Government
    - Better security of public information and privacy of citizens
    - Citizens have better access to government information and services

- Improved interaction at the local level (e.g., visiting a local government website increase citizens' trust in local governments)
- Protection of foundational values of trustworthiness, openness, robustness, Reliability, accountability and security

6. Improved Social Value and Well-Being
    - Creation of value(s) for families, community, and other enabling freedom and equal rights
    - Increase ease of doing business (i.e., create a value for citizens in terms of increased citizens' well-being and quality of life)
    - Increased citizens' well-being and quality of life
    - A more flexible, pervasive, and cost-effective public sector (e.g., provision of online applications and transactions)

# B Conceptual framework: Public value creation through low-code development platforms



# C Survey

Q1.1 Beste deelnemer,

Bedankt voor het vrijmaken van 10-15 minuten van uw tijd voor het invullen van deze survey over low-code ontwikkelingsplatformen in de publieke sector.

Low-code ontwikkelingsplatformen zijn producten en/of clouddiensten voor applicatieontwikkeling die gebruikmaken van visuele technieken om sneller te kunnen programmeren. Bij deze ontwikkelingsplatformen kunt u denken aan bijvoorbeeld Mendix, Bettyblocks, Outsystems en Powerapps.

1. Over deze studie Deze survey is ontwikkeld als onderdeel van een afstudeer onderzoek voor de master ICT in Business & the public sector van de Universiteit Leiden. Het doel van de survey is om in kaart te brengen hoe deze ontwikkelingsplatformen in de praktijk presteren en hoe dit zich verhoudt tot traditionele ontwikkelmethodes in de publieke sector. Naast voor- en nadelen wordt ook gekeken naar eventuele organisatorische en maatschappelijke bijdrages.

2. Wie kan meedoen? De survey is bedoeld voor elke gebruiker van deze ontwikkelingsplatformen in de publieke sector. Zowel voor IT'ers, als niet-IT'ers.

3. Wat kan je verwachten? De survey bestaat uit 18 vragen. Bij elke beantwoorde vraag kunt u aangeven in hoeverre u de gevraagde eigenschap belangrijk vindt. Daarnaast is er een optioneel veld om uw antwoord te verduidelijken met voorbeelden of uitleg.

4. Privacy en klachten Door naar de volgende pagina te gaan, bevestigt u uw deelname aan dit onderzoek. U behoudt daarbij het recht deze instemming weer in te trekken zonder dat u daarvoor een reden hoeft op te geven. Als de onderzoeksresultaten worden gebruikt in wetenschappelijke publicaties, of op een andere manier openbaar worden gemaakt, dan zal dit volledig geanonimiseerd gebeuren. Uw persoonsgegevens worden niet door derden ingezien zonder uw uitdrukkelijke toestemming. Voor eventuele klachten over dit onderzoek kunt u zich wenden tot de Ethische Commissie Wiskunde en Natuurwetenschappen: ethicscommittee@science.leidenuniv.nl

Als u meer informatie wilt, nu of later, ben ik bereikbaar via mail:
j.p.sijtstra@umail.leidenuniv.nl of telefonisch: 06 832 698 78.

Alvast bedankt voor uw deelname.

Q2.1 Bij welke organisatie bent u werkzaam?

Q2.2 Wat omschrijft uw organisatieonderdeel het beste?
o Meer IT of informatie georiënteerde afdeling
o Meer business georiënteerde afdeling

Q2.3 Wat is uw functietitel?

Q2.4 Welke programmeerervaring had u voor dat u gebruik maakte van het low-code ontwikkelingsplatform (u kunt meerdere antwoorden aangeven) ?
- Geen ervaring
- Frontend
- Backend
- Website ontwikkeling
- Mobiele applicatie ontwikkeling
- Data scientist
- Anders, namelijk:

Q2.5 Met welke low-code ontwikkelingsplatformen heeft u ervaring bij uw huidige organisatie (u kunt meerdere antwoorden aangeven)?
- Mendix
- Bettyblocks
- Outsystems
- Powerapps
- Anders, namelijk:

Q2.6 Hoelang werkt u zelf al met deze platform(en) ?
o Minder dan 6 maanden
o Tussen de 6 maanden en twee jaar
o Langer dan twee jaar

Q2.7 Is het low-code ontwikkelingsplatform een pilot project (praktijktest) ?
o Ja
o Nee

Q2.8 Hoeveel verschillende applicaties zijn er tot nu toe mee ontwikkeld binnen uw organisatie?
o 0
o 1
o 2 tot en met 5

o Meer dan 5
o Weet ik niet

Q3.1 Hieronder volgen 10 factoren die te maken hebben met low-code ontwikkelingsplatformen. Sommige van deze factoren kunnen heel algemeen worden opgevat als voor- of nadelen van IT. We willen u vragen om hier naar te kijken met specifiek low-code ontwikkelingsplatformen in het achterhoofd. Het gaat hierbij om de vergelijking met traditionele programmeermethodes.

Bij elke beantwoorde vraag kunt u aangeven in hoeverre u deze eigenschap belangrijk vindt. Daarna is er de mogelijkheid om uw antwoord te verduidelijken met voorbeelden of uitleg, dit is niet verplicht.

Als u een vraag niet begrijpt of niet genoeg ervaring mee heeft opgedaan om de vraag goed te kunnen beantwoorden kunt u de optie geen mening kiezen.

Q3.2 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over de hoeveelheid handmatig codeerwerk dat nodig is om een applicatie naar wens te krijgen?
o Veel meer handmatige code nodig
o Meer handmatige code nodig
o Neutraal
o Minder handmatige code nodig
o Veel minder handmatige code nodig
o Geen mening

Q3.3 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.4 Uitleg: (optioneel)

Q3.5 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over de integratie van bestaande en nieuwe IT systemen met het low-code ontwikkelingsplatform?
o Veel moeilijker om te integreren
o Moeilijker om te integreren
o Neutraal
o Makkelijker om te integreren
o Veel makkelijker om te integreren
o Geen mening

Q3.6 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.7 Uitleg: (optioneel)

Q3.8 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over de leersnelheid van het low-code ontwikkelingsplatform?
o Veel moeilijker om te leren
o Moeilijker om te leren
o Neutraal
o Makkelijker om te leren
o Veel makkelijker om te leren
o Geen mening

Q3.9 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.10 Uitleg: (optioneel)

Q3.11 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over de hoeveelheid onderhoud dat nodig om de applicaties naar wens te houden?
o Veel meer onderhoud nodig
o Meer onderhoud nodig
o Neutraal
o Minder onderhoud nodig
o Veel minder onderhoud nodig
o Geen mening

Q3.12 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.13 Uitleg: (optioneel)

Q3.14 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over de afstemming van business en IT (Business-IT alignment) ?
o Veel slechtere afstemming
o Slechtere afstemming
o Neutraal
o Betere afstemming
o Veel betere afstemming
o Geen mening

Q3.15 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.16 Uitleg: (optioneel)

Q3.17 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over de schaalbaarheid van ontwikkelde applicaties met een low-code ontwikkelingsplatform? (eenvoudig groter maken voor meer gebruikers, meer verzoeken verwerken, etc.)
o Veel slechtere schaalbaarheid
o Slechtere schaalbaarheid
o Neutraal
o Betere schaalbaarheid
o Veel betere afstemming
o Geen mening

Q3.18 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.19 Uitleg: (optioneel)

Q3.20 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over kennisintegratie met een low-code ontwikkelingsplatform?
(Identificeren en toepassen van gespecialiseerde kennis die zich in verschillende delen van een organisatie bevindt)
o Veel slechtere kennisintegratie
o Slechtere kennisintegratie
o Neutraal
o Betere kennisintegratie
o Veel betere kennisintegratie
o Geen mening

Q3.21 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.22 Uitleg: (optioneel)

Q3.23 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over de algehele IT productiviteit?
o Veel lagere IT productiviteit
o Lagere IT productiviteit
o Neutraal
o Hogere IT productiviteit

o Veel hogere IT productiviteit
o Geen mening

Q3.24 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.25 Uitleg: (optioneel)

Q3.26 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over de algehele kosten van ontwikkelde applicaties?
o Veel duurder
o Duurder
o Neutraal
o Goedkoper
o Veel goedkoper
o Geen mening

Q3.27 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.28 Uitleg: (optioneel)

Q3.29 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over een verandering in flexibiliteit of wendbaarheid van uw organisatie m.b.t. het low-code ontwikkelingsplatform?
(snel in kunnen spelen op veranderende externe en interne ontwikkelingen, business agility)
o Veel slechtere flexibiliteit
o Slechtere flexibiliteit
o Neutraal
o Betere flexibiliteit
o Veel betere flexibiliteit
o Geen mening

Q3.30 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.31 Uitleg: (optioneel)

Q3.32 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over de afhankelijkheid of in andere worden de moeilijkheid van overstappen naar een nieuw IT systeem (vendor lock-in) ?
o Veel slechter
o Slechter
o Neutraal
o Beter
o Veel beter
o Geen mening

Q3.33 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.34 Uitleg: (optioneel)

Q3.35 Vergeleken met traditionele programmeermethodes, wat kunt u zeggen over de hoeveelheid customisatie die haalbaar is met het low-code ontwikkelingsplatform. o Veel minder customisatie
o Minder customisatie
o Neutraal
o Meer customisatie
o Veel meer customisatie
o Geen mening

Q3.36 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q3.37 Uitleg: (optioneel)

Q3.38 Welke nog niet besproken voor- of nadelen van low-code ontwikkelingsplatformen ziet u zelf nog meer?

Q4.1 Hieronder volgen 5 organisatiecapaciteiten die te maken hebben met low-code ontwikkelingsplatformen en de publieke sector. Bij elke vraag kunt u aangeven in hoeverre u deze capaciteit belangrijk vindt. Daarna is er de mogelijkheid om uw antwoord te verduidelijken met voorbeelden of uitleg. Als u een vraag niet begrijpt of niet genoeg ervaring mee heeft opgedaan om de vraag goed te kunnen beantwoorden kunt u deze overslaan.

Q4.2 Door het gebruik van een low-code ontwikkelingsplatform zijn wij in staat een betere service te verlenen met dezelfde middelen (budget, fte) .
o Helemaal mee eens
o Mee eens
o Neutraal
o Niet mee eens
o Helemaal niet mee eens
o Geen mening

Q4.3 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q4.4 Uitleg: (optioneel)

Q4.5 Door het gebruik van een low-code ontwikkelingsplatform zijn wij beter in staat samen te werken met andere organisaties (privaat en publiek) .
o Helemaal mee eens
o Mee eens
o Neutraal
o Niet mee eens
o Helemaal niet mee eens
o Geen mening

Q4.6 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q4.7 Uitleg: (optioneel)

Q4.8 Door het gebruik van een low-code ontwikkelingsplatform zijn wij beter in staat om de benodigde middelen (budget, goedkeuring, etc.)
te verzamelen om een nieuw initiatief te lanceren. o Helemaal mee eens
o Mee eens
o Neutraal

o Niet mee eens
o Helemaal niet mee eens
o Geen mening

Q4.9 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q4.10 Uitleg: (optioneel)

Q4.11 Door het gebruik van een low-code ontwikkelingsplatform zijn wij beter in staat het publiek te betrekken bij het opstellen en/of uitvoeren van beleid. o Helemaal mee eens
o Mee eens
o Neutraal
o Niet mee eens
o Helemaal niet mee eens
o Geen mening

Q4.12 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q4.13 Uitleg: (optioneel)

Q4.14 Door het gebruik van een low-code ontwikkelingsplatform zijn wij beter in staat te innoveren. o Helemaal mee eens
o Mee eens
o Neutraal
o Niet mee eens
o Helemaal niet mee eens
o Geen mening

Q4.15 Hoe belangrijk vindt u dit?
o 0 o 1 o 2 o 3 o 4 o 5 o 6 o 7 o 8 o 9 o 10

Q4.16 Uitleg: (optioneel)

Q5.1 Kunt u uitleggen wat volgens u de meerwaarde van het gebruik van een low-code ontwikkelingsplatform is voor de maatschappij in het algemeen? (Denk aan bijvoorbeeld verantwoordelijkheid naar de burger, efficient gebruik van budget, integriteit, transparantie, respecteren van privacy, gelijke afhandeling van verzoeken)

Q5.2 Mag ik contact opnemen als er aanvullende vragen zijn naar aanleiding van uw antwoorden?
o Ja, dit kan op mailadres:
o Nee, bedankt

# D Survey invitations (in Dutch)

## Mail or LinkedIn example

Beste,

In samenwerking met Capgemini en de Universiteit Leiden doen wij onderzoek naar de prestaties van low-code ontwikkelingsplatformen in de publieke sector (Mendix, Bettyblocks, etc). Rondom deze platformen wordt veel geclaimed, maar is helaas nog geen objectief onderzoek naar gedaan. Hier willen we met dit onderzoek verandering in brengen door een wetenschappelijk referentiekader te creëren. Van dit onderzoek kunnen bestaande en nieuwe projecten binnen de publieke sector profiteren door de betere beeldvorming over welke voor- of nadelen andere projecten en organisaties daadwerkelijk ervaren.

Hierbij hebben wij uw hulp hard nodig! Het invullen van de survey kost maar 10-15 minuten. Mocht u andere geïntereseerden kennen, schroom dan niet dit bericht door te sturen. De (anonieme) survey is beschikbaar op:
https://leidenuniv.eu.qualtrics.com/jfe/form/SV_6zAC3KLwwOyUUcu.

Alvast onze hartelijke dank,

## Follow-up mail example

Beste relatie,

Graag herinneren wij je aan onderstaande mail. Mocht je deze vragenlijst al ingevuld hebben, kun je deze mail negeren. Indien je de vragenlijst voor de evaluatie van low-code prestaties in de publieke sector wilt invullen, willen wij u vriendelijk verzoeken dit binnen 2 weken te doen via de volgende link:
https://leidenuniv.eu.qualtrics.com/jfe/form/SV_6zAC3KLwwOyUUcu.
Het invullen kost maar 10-15 minuten. Mocht u andere geïntereseerden kennen, schroom dan niet dit bericht door te sturen.

In ieder geval hartelijk dank voor jouw tijd. Mocht u vragen hebben, neem dan graag contact op via (ommitted for privacy)

Alvast onze hartelijke dank,

## The Royal Dutch Association of Information Professionals (KNVI)

Beste deelnemer van de KNVI Organisatie en IT wendbaarheid door DEMO & low-code,

Naar aanleiding van jullie aanwezigheid bij de KNVI sessie over low-code zou ik willen vragen of jullie de wetenschap een handje willen helpen. Jelle Sijtstra doet wetenschappelijk onderzoek naar de waarde van low code platformen zoals Mendix, Betty blocks, Outsystems en Powerplatform in de publieke sector via onderstaande enquete. Werk

je in de publieke sector en heb je ervaring met deze platformen, vul hem dan in, max 10-15 min.

https://leidenuniv.eu.qualtrics.com/jfe/form/SV_6zAC3KLwwOyUUcu

Alvast bedankt,
Christiaan Konstapel
Bestuurslid KNVI interessegroep Architectuur