Emerging low-code/no-code paradigm: Evaluating adoption, opportunities, and cyber security challenges in the information technology sector

Paavo Nelimarkka

Master’s thesis
May 2023
Master's Degree Programme in Information Technology,
Full Stack Software Development

Jyväskylän ammattikorkeakoulu
University of Applied Sciences

Nelimarkka, Paavo

Emerging low-code/no-code paradigm: Evaluating adoption, opportunities, and cyber security challenges in the information technology sector

Jyväskylä: Jamk University of Applied Sciences, May 2023, 122 pages

Full Stack Software Development, Master’s Degree Programme in Information Technology, Master´s thesis

Permission for web publication: Yes

Language of publication: English

Abstract

Low-code/no-code technologies enable companies to increase the productivity and delivery of software by providing a more streamlined development experience. With these technologies the developers can use graphical interfaces and tools to create software logic with less code or no code at all. Their aim is to democratize application development by empowering people with less technical know-how to create software logic.

The purpose of this research was to figure out the rate of adoption, prospects and cyber security implications of low-code/no-code technologies. The studies were focused on professionals of information technology and/or cyber security fields.

A literature review and a mixed method study was designed and conducted based on the research objectives. The research includes a survey study which was supported with a set of semi-structured theme interviews. Quantitative data produced by the survey was analyzed by dividing respondents into groups and comparing their results with each other. The survey and the interviews also produced qualitative data which was analyzed with qualitative content analysis technique.

Analysis of the data produced interesting findings which were compared to the findings from the literature review and former research. Findings suggest that about half of the information technology and/or cyber security companies are already utilizing low-code/no-code technologies and many companies are expecting to adopt these technologies in near future. Study also accomplished to produce evaluation of cyber security risks related to low-code/no-code technologies and means to mitigate them.

Research accomplished to produce insight based on the research objectives and suggestions for practical implications was presented.

Keywords/tags (subjects)

Low-code, no-code, cyber security

Miscellaneous (Confidential information)

Nelimarkka, Paavo

Description

Emerging low-code/no-code paradigm: Evaluating adoption, opportunities, and cyber security challenges in the information technology sector

Jyväskylä: Jyväskylän ammattikorkeakoulu, Toukokuu 2023, 122 sivua

Full Stack Software Development, Master's Degree Programme in Information Technology, Master´s thesis

Verkkojulkaislupa myönnetty: Kyllä

Julkaisun kieli: englanti

Tiivistelmä

Low-code/no-code teknologiat lisäävät yritysten tuottavuutta sekä vauhdittavat ohjelmistojen toimitusta yksinkertaistamalla ohjelmistokehitystä. Graafisten käyttöliittymien ja työkalujen avulla nämä teknologiat mahdollistavat ohjelmalogiikan luonnin vähemmällä määrällä ohjelmistokoodia, tai jopa täysin ilman. Näiden teknologioiden on tarkoitus mahdollistaa ohjelmistokehitys sellaisellekin ihmiselle, jolla ei ole laajaa kokemusta ohjelmistokehityksestä, ja täten tehdä sovellusten kehittämisestä mahdollista kaikille.

Tämän tutkimuksen tarkoitus oli selvittää näiden teknologioiden käyttöaste, tulevaisuuden näkymät sekä tutkia niitä kyberturvallisuuden näkökulmasta. Tutkimus keskittyi IT- sekä kyberturvallisuusalan ammattilaisiin.

Kirjallisuuskatsaus sekä monimenetelmällinen tutkimus suunniteltiin ja toteutettiin tutkimustavoitteisiin pohjautuen. Tutkimuskokonaisuus sisältää kyselytutkimuksen, jonka tueksi toteutettiin myös puoliavoimia teemahaastatteluja. Kvantitatiiviset tulokset analysoitiin jakamalla kyselyyn vastanneet ryhmiin ja vertailemalla ryhmiä keskenään. Tutkimukset tuottivat myös kvalitatiivisia tuloksia, jotka analysoitiin kvalitatiivisella sisältöanalyysi-menetelmällä.

Tulosten analysointi tuotti mielenkiintoisia löydöksiä, joita peilattiin kirjallisuuskatsauksesta, sekä aikaisemmasta tutkimuksesta löydettyihin tutkimustuloksiin. Löytöjen perusteella noin puolet IT- sekä kyberturvallisuuden yrityksistä tai osastoista käyttävät jo low-code/no-code teknologioita, ja usean yrityksen työntekijät uskovat yrityksen ottavan teknologioita käyttöön lähitulevaisuudessa. Tutkimuksessa käytiin myös läpi low-code/no-code teknologioihin liittyviä kyberturvallisuusriskejä, sekä kuinka näitä riskejä voisi estää.

Tutkimus onnistui täyttämään tutkimustavoitteensa ja käytännön tapoja jalkauttaa tuloksia yrityksissä tuotiin esille.

Avainsanat

Low-code, no-code, kyberturvallisuus

Muut tiedot (salassa pidettävät liitteet)

Contents

1 Introduction 9
1.1 Background and context 9
1.2 Research objectives and scope 10
1.3 Former research 11
1.4 Structure of the thesis 12
2 Methodology 13
2.1 Research design 13
2.2 Data collection methods 14
2.3 Sampling 14
2.4 Survey 15
2.5 Semi-structured interviews 16
2.6 Data analysis techniques 17
3 Literature review 18
3.1 Literature review method and process 18
3.2 Low-code/no-code technology 21
3.3 Adaptation and prospects of low-code/no-code 28
3.4 Security implications of low-code/no-code technologies 31
3.5 Summary of the literature review 36
4 Results 37
4.1 Survey 37
4.1.1 Basic questions 38
4.1.2 Adaptation and prospects of low-code/no-code technology 47
4.1.3 Cyber security implications 55
4.1.4 Survey analysis 70
4.2 Semi-structured theme interviews 91
5 Discussion 99
5.1 Summary of research questions 99
5.2 Interpretation of findings 101
5.3 Integration of findings with the literature 104
5.4 Reliability, limitations and ethics 105
6 Conclusion 108
6.1 Implications for practice 108
6.2 Future research 108

6.3 Final thoughts..109

References 110

Appendices 118

Appendix 1. Interview guide 118

Appendix 2. Question 16: Please describe how does your company utilize low-code technologies. 119

Appendix 3. Question 17: Please describe how could your company utilize low-code technologies. 120

Appendix 4. Question 30: Do you have any comments concerning the list of risks from OWASP in previous page? Or do you suspect any other possible risks concerning low-code/no-code technologies? 121

Figures

Figure 1. Deductive category application. Adapted from “Qualitative Content Analysis” by Mayring, P., 2000, Forum: Qualitative Social Research. Created by the author. 18

Figure 2. Systematic literary resource process based on PRISMA 2020. Adapted from “PRISMA 2020 flow diagram for updated systematic reviews”, 2020. Created by the author. 20

Figure 3. Translation of high-level programming languages to low level computer languages. Created by the author. 22

Figure 4. History of programming languages based on Sufis article (Sufi, F. 2023). Created by author. 23

Figure 5. A workflow in N8N LCNC platform producing an RSS feed based on the content of a website. Created by the author. 24

Figure 6. Logic for light switch toggle in Unreal Engine 5 Blueprint visual scripting language. Created by author. 25

Figure 7. What are low-code used for? Based on the OutSystems 2020 report ("The State of Application Development”, 2020). 26

Figure 8. How much faster is low-code development compared to the traditional development based on data from Creatios report (“The State of Low-Code/No-Code.”, 2021). Created by author. 27

Figure 9. Technology adoption lifecycle. “Diffusion of Innovations” by Wesley, F. Flickr, licensed under CC BY-SA 2.0. 28

Figure 10. Low-code technology revenues in billion US dollars based on estimations of Gartner. Created by the author. 29

Figure 11. What is the level of your education? 39

Figure 12. Where do you currently live? 40

Figure 13. Which (of these) describes your department or company the best? 41

Figure 14. How many people does your company employ? 43

Figure 15. How familiar are you with software development in general? Option one (1) stands for “not familiar at all and five (5) for “very familiar”. 44

Figure 16. Does your work involve software development? 45

Figure 17. How experienced are you as a software developer? 46

Figure 18. How familiar are you with low-code/no-code software development paradigm or technologies? Option one (1) stands for “not familiar at all” and five (5) for “very familiar”. 48

Figure 19. Is your company already utilizing low-code/no-code software development? 49

Figure 20. When would you roughly estimate your company start utilizing low-code/no-code technology? 51

Figure 21. How well does low-code/no-code technology serve your company's needs? Option one (1) stands for “not at all” and five (5) for “very well”. 52

Figure 22. How well would low-code/no-code technologies serve your company's needs? Option one (1) stands for “not at all” and five (5) for “very well”. 54

Figure 23. Do you have any knowledge on cyber security? 56

Figure 24. How experienced are you in cyber security? 57

Figure 25. Account impersonation. Option one (1) stands for “least likely” and five (5) for “most likely” 58

Figure 26. Authorization Misuse. Option one (1) stands for “least likely” and five (5) for “most likely” 59

Figure 27. Data Leakage and Unexpected Consequences. Option one (1) stands for “least likely” and five (5) for “most likely”. 60

Figure 28. Authentication and Secure Communication Failures. Option one (1) stands for “least likely” and five (5) for “most likely”. 61

Figure 29. Security Misconfiguration. Option one (1) stands for “least likely” and five (5) for “most likely” 63

Figure 30. Injection Handling Failures. Option one (1) stands for “least likely” and five (5) for “most likely” 64

Figure 31. Vulnerable and Untrusted Components. Option one (1) stands for “least likely” and five (5) for “most likely”. 65

Figure 32. Data and Secret Handling Failures. Option one (1) stands for “least likely” and five (5) for “most likely”. 67

Figure 33. Asset Management Failures. Option one (1) stands for “least likely” and five (5) for “most likely”. 68

Figure 34. Security Logging and Monitoring Failures. Option one (1) stands for “least likely” and five (5) for “most likely”. 69

Figure 35. LCNC technology adoption compared to the company sizes. 73

Figure 36. Familiarity on software development between companies using and not using LCNC technologies. 74

Figure 37. Familiarity with LCNC technologies among the companies 75
Figure 38. Companies utilizing LCNC. 76
Figure 39. Comparison of estimated adoption of LCNC technologies. 77
Figure 40. Comparison of cyber security and software companies on the question how well does LCNC technologies server their needs. 78
Figure 41. Company size comparison on how well the LCNC technology server the company's needs 79
Figure 42. Differences between small to mid-sized enterprises against large enterprises. 80
Figure 43. Comparison of groups based on knowledge on cyber security. 82
Figure 44. Comparison of groups in the years of experience on cyber security. 83
Figure 45. Account impersonation comparison. 84
Figure 46. Account impersonation comparison based on company sizes. 84
Figure 47. Authorization misuse comparison between company sizes. 85
Figure 48. Data leakage and unexpected consequences comparison. 86
Figure 49. Security misconfiguration comparison. 87
Figure 50. Comparison on data and secret handling failures risk. 88
Figure 51. Comparison on asset management failures. 89
Figure 52. Comparison on security logging and monitoring failures. 90
Figure 53. Qualitative content analysis theming for the interviews 93
Figure 54. Technology adoption lifecycle. Adapted from “Diffusion of Innovations” by Wesley, F. Flickr, licensed under CC BY-SA 2.0. 100

Tables

Table 1. Example on scarcity of cyber security related academic literature 21
Table 2. Definitions of terminology by NIST. Adapted from National Institute of Standards and Technology. 32
Table 3. Where do you currently live? 40
Table 4. How many people does your company employ? 43
Table 5. How familiar are you with software development in general? 45
Table 6. Does your work involve software development? 46
Table 7. How familiar are you with low-code/no-code software development paradigm or technologies? 48
Table 8. Is your company already utilizing low-code/no-code software development? 50
Table 9. How well does low-code/no-code technology serve your company's needs? 53
Table 10. How well would low-code/no-code technologies serve your company's needs? 54

Table 11. Account Impersonation. 58
Table 12. Authorization misuse. 59
Table 13. Data Leakage and Unexpected Consequences. 61
Table 14. Authentication and Secure Communication Failures. 62
Table 15. Security Misconfiguration. 63
Table 16. Injection Handling Failures. 64
Table 17. Vulnerable and Untrusted Components. 66
Table 18. Data and Secret Handling Failures. 67
Table 19. Asset Management Failures. 68
Table 20. Security Logging and Monitoring Failures. 70
Table 21. Survey analysis groups 71
Table 22. Categories, descriptions and coding rules of the qualitative content analysis 94

1 Introduction

1.1 Background and context

In today’s fast-paced and increasingly digitalized industries, low-code and no-code technologies have emerged as a crucial enabler for rapid application development and innovation. Software development is getting increasingly more automated and low-code/no-code together with, for example, machine learning, artificial intelligence and robot process automation is one of those drivers (Hurlburt, 2021). The demand for solutions to address unique business challenges and the possibility to adapt to changes drive the development of these technologies.

Low-code and no-code are software development technologies that enable developers to create software logic with minimum to no programming code at all. Low-code technologies facilitate faster software development with less coding, while no-code empowers so called “citizen developers” to create software logic even without previous programming experience (Bloomberg, 2017). All this can be achieved via graphical user interfaces by using, for example, forms or drag and drop actions. Low-code development platforms provide a user interface and all necessary tools to design, develop and deliver software. These technologies democratize app development by empowering “citizen developers” (“What is low code?”, n.d.).

In their study Sanchis (et al.) took notice of the lack of current study and publications for the subject of “low code”, especially when the search was fined down with additive keywords (Sanchis, R. et al., 2019). As a new and trending technology cyber security issues on low-code & no-code technologies are also a topic that needs more research (Spets, 2022). To ensure cyber security the same precautions must be applied whether the developer is an IT professional or a “citizen developer” (Hurlburt, 2021). Like any other software development technology, low-code & no-code also have potential security risks which are crucial to be aware of while adopting these technologies.

JYVSECTEC is an independent cyber security research, development and training center which operates in Jamk University of Applied Sciences Institute of Technology (“About Us”, n.d.). JYVSECTEC provides certification, cyber exercises, training, testing, research, and consulting. They have a large customer base from various industries and sectors and need to be aware of current trends in software development technology.

The motivation for this thesis stems from several factors: the scarcity of research on these technologies, particularly in cyber security standpoint, and the significance for JYVSECTEC to be well informed about these technologies, their adoption, and their attributes, with a specific emphasis on cyber security implications.

In this report low-code/no-code will be shortened 'LCNC' and low-code development platforms 'LCDP'. The term 'low-code' is used as an umbrella term for both, low-code and no-code, where the separation is not deemed necessary.

1.2 Research objectives and scope

The objective of this research is to figure out the current extent of low-code/no-code technology adoption and to gain an understanding of its anticipated prospects in the near future. Primary research questions are:

- RQ1: How far the adaptation of LCNC software development technology has reached in the IT field?
- RQ2: What are the prospects of the LCNC technology in the IT field in near future?

A secondary objective of this research is to examine the technical attributes of low-code and no-code technologies in relation to cyber security. Secondary research question is:

- RQ3: What are the potential cyber security threats concerning the LCNC technologies?

This research focuses mainly on the current day technologies marketed with the term "low-code" and/or "no-code". They are usually cloud-based web platforms, but they can also be on-site installations or other kinds of tools as well. Model-driven engineering (MDE) was not excluded because of the similarities of these technologies. The study of prospects of these technologies is limited in the near or semi-near future.

Although low-code/no-code technologies cater to users without professional software development experience, this study specifically targets professionals in the information technology field.

1.3 Former research

In the early stages of this thesis research in 2021 there was not that much similar research available. While conducting the research there have been several novel bachelor and master level research published.

Virta (2018) studied the nature, differences, and possibilities of low-code development by interviewing professionals in Finnish Salesforce consulting company. This research compared the low-code technology to more traditional software development (Virta, T. 2018).

Kermanchi (2022) studied the developer experience of software developers using low-code development platforms. This study also compared the differences between these low-code platforms compared to traditional development platforms (Kermanchi, A. 2022).

Alyousef (2021) lists challenges developers encounter while developing with low-code technologies in their master’s thesis. For example, among citizen developers a lack of experience in information technology was found to be one of these challenges, or the lack of experience on low-code platforms in the company. Extending applications with features not provided in the low-code platform was also deemed difficult because of the lack of proper integrated development environment (IDE) (Alyousef, Z. 2021).

In their master’s thesis, Spets (2022) studied the compliance of a large low-code development vendors platform with the Application Security Verification Standard (ASVS). They found out that the platform was mainly compliant with the standard. The author also presents fixes for those deficiencies deemed minor. They also found one larger problem in input validation. Spets suggested the topic of low-code and security requires more research (Spets, 2022).

One particularly similar research was published from Lapland University of Applied Sciences. In their bachelor thesis Turunen (2022) implemented a survey study on the adoption, experiences, possible barriers and the future of low-code technologies in Finnish small-medium sized IT companies. Although it has a lot in common with this thesis, the scope is only in Finnish companies, and it does not address cyber or information security issues (Turunen, O. 2022).

This thesis study aims to produce data on the most recent adoption and prospects of low-code/no-code technologies. Compared to the previous studies it has an emphasis on the cyber security implications of low-code/no-code technologies which is an area where current studies are lacking. In their thesis Spets (2022) evaluated the compliance of an existing low-code development platform against a security standard created by the OWASP community (Spets, 2022). This study uses a risk listing by the same OWASP community and evaluates these risks related to low-code technologies with a broader scope.

As stated earlier, the motivation for this thesis stems from multiple factors. The low-code/no-code movement lacks academic research, especially from the cyber security perspective, which can be seen as the main motivation for this thesis research. The thesis study was commissioned by JYVSECTEC to gather insight on these technologies and their cyber security implications. Additionally, the author wanted to produce data for the academic and professional community interested in the research on low-code/no-code technologies.

1.4 Structure of the thesis

This thesis report continues with the methodologies in chapter 2. The survey and interview studies are described along with research design, data collection methods, sampling, data analysis methods, ethical considerations and limitations.

Literature review is conducted in chapter 3. The review provides understanding on the low-code/no-code software development movement, the adaptation and prospects of these technologies and the security implications. This knowledge gives insight needed for the study conducted.

The data of studies conducted is represented in chapter 4. The analysis of the data is also under the same chapter.

After the presentation and analysis of the data they are discussed further in chapter 5, where the interpretations of findings, integration with the literature, reliability, limitations and ethics are reviewed. The concluding chapter is chapter 6, where the implications for practice and possibilities for future research are presented.

2 Methodology

2.1 Research design

Since the LCNC movement is not just purely about technology but it also has social aspects, like the concept of governing “citizen developers” for cyber and information security reasons, there was a need for deeper understanding of emerging ideas and thoughts about the movement overall. In their book Feilzer (2009) summarizes positivism/post-positivism to point towards a single truth of reality, which can only be studied objectively with quantitative research methods. Constructivism on the other hand indicates multiple realities based on subjective experience and needs to be studied with qualitative methods. Pragmatism can be seen as a middle-ground where the focus is on the problem being researched rather than the dichotomy of these two philosophies (Feilzer, M. Y. 2009, pp. 6-7).

So, is it productive to subscribe to either positivism or constructivism? How can we produce the best possible study for this research problem? LCNC technology movement is novel and academic study on the topic is still quite scarce. For this research all the possible information gathered on the issue was deemed valuable, be it quantitative or qualitative, and based on the nature of the study it’s safe to conclude the study follows the research paradigm of pragmatism.

The research was conducted by using a mixed method survey and a supportive semi-structured theme interview.

The survey questionnaire has multiple-choice questions, but also open question forms for the responder to write longer and more detailed answers. Therefore, the survey gathers quantitative and qualitative data, and can be categorized as mixed method study.

Theme interviews on the other hand did not gather any quantitative data, only qualitative. The interview had a structured set of question topics, but also allowed open discussion, especially in the last part, hence it is referred to as semi-structured interview.

2.2 Data collection methods

For the survey study a Webropol service provided by the Jamk University of Applied Sciences was utilized. Webropol is a survey and feedback service which provides multiple modules to handle and assess the data (Webropol, n.d.). For this study the basic survey, reporting and analysis tools were used. The final survey was a web-based online questionnaire which could be shared via hyperlink.

As stated earlier, the scope of responders to the study was professionals of information technology and/or cyber security professionals. To reach responders for the survey a social media campaign was conducted. Because of the sizeable networks of the author and his near colleagues the professional social network LinkedIn was heavily utilized for this purpose. Survey was also shared on common channels of the Microsoft Teams chat services used in Jamk University of Applied Sciences. An application for a permit to share on the university's Teams channels was submitted and permission was granted for this purpose. To gather more interest to respond, a raffle of three Hack the Box gift cards was arranged between the respondents willing to participate.

Although the marketing was conducted in English, there is an assumed emphasis on people working in Finland within the social networks and chat channels involved.

The survey study was open for responses from 1.12.2022 to 18.12.2022 and the theme interviews were held from February to March 2023.

Main concern with gathering the data was the willingness of cyber security professionals to answer the survey or participate in an interview. For this reason, the anonymity of the respondents was ensured.

2.3 Sampling

In this study a non-probability sampling method was chosen. Berndt (2020) divides sampling methods in probability and non-probability sampling in their article. Probability sampling has usually features such as random selection, objective method and statistical interference, for example.

Features like non-random selection, subjective method and analytical interference represent non-probable sampling (Bernt, A. E. 2020).

Based on the research objectives the aim was to focus the survey on professionals of information technology and cyber security. The survey was freely shared among the professional network where the potential respondents made the decision to participate in the study, therefore, the study followed the self-selection sampling. In their article Berndt warns of possible selection bias in this method (Berndt, A. E. 2020).

In the end, the sample size (n) of 76 was accomplished in the survey study.

2.4 Survey

In their book on mixed-mode surveys Dilman et al (2014) lists four crucial errors to be aware of while conducting a survey:

- Coverage error happens when the set of respondents does not represent attributes of the estimated population.
- Sampling error is when only a certain part of the framed sample is surveyed, but the estimation is based on the full population.
- Nonresponse error happens when just one group of the sample responds to the survey and differ from those who did not respond, producing a flawed estimation.
- Measurement error derives from inaccurate responses provided by the respondents, because of flawed design of questions of poor data collection, for example.

The quality and accuracy of the results can be increased by minimizing these potential errors while designing and conducting the study (Dilman, D. A. et al., 2014. pp. 3-4).

Because of the disruptive nature of LCNC movement a wide range of differing answers and opinions was expected and appreciated, as stated in the earlier chapter. The structure of the survey followed the research objectives and the questions, and was divided into two sections:

- Adaption and prospects of low-code/no-code technologies
    - Basic information of the responder
    - Adaption of low-code/no-code technologies (RQ1)
    - Prospects of low-code/no-code technologies (RQ2)

- Cyber security professionals' perspective on the technology (RQ3)

These sections started with multiple-choice types of forms. After some of these questions there are open-text inputs for the users to provide additive details. In the end of these sections were usually open-ended questions to provide a broader description on the subject of the section. Because of this structure, the survey produces mostly quantitative data, but some qualitative data was gathered as well.

A pilot test phase was conducted on the study on November 2022 and the survey was updated based on the feedback gathered. Respondents of the pilot study were colleagues from the university.

After the survey phase all the results were read through by the author and deemed unique. The margin of error was also calculated and can be found in chapter 5.3.1.

2.5 Semi-structured interviews

Dilman et al (2014) emphasizes the mixed-mode survey methodology in their book. By using more than a single mode of survey it is possible to minimize the four errors listed in the previous chapter (Dilman, D. A. et al., 2014. pp. 12-13).

To increase the quality of the research it was supplemented with five (5) semi-structured theme interviews. This interview was divided in five sections:

- How long work experience do you have?
- How have you become familiar with LCNC technologies?
- What is your perspective on the future developments of LCNC technologies?
- Have you considered the potential impact of LCNC technologies on cyber security?
- Open discussion about the topic

These sections also had sub-questions, for example, 'Do you have any experience on cyber security?' or 'What was the reasoning behind selecting these technologies?'. In their paper Kallio et al (2016) argued that creating and presenting an interview guide for the semi-structured interview increases the credibility of the study. (Kallio, H. 2016) The full interview guide was implemented and was included in this thesis report (Appendix 1.).

The final section was open discussion where the interviewer and the interviewee could discuss and share information.

2.6 Data analysis techniques

The quantitative data was initially explored with Spearman and Pearson correlation coefficient, and some potential correlations with a statistical significance (p-value < 0.05) were found. These correlations did not end up being useful in the context of analysis, and it was deemed that the sample (n=76) was too small for proper statistical analysis.

The analysis of the quantitative data was concluded by forming groups and comparing the results of the groups against each other. The formation of the groups to be compared was based on the literature review, topic of survey question and the significance in terms of analysis and research objectives.

To analyze the semi-structured theme-interviews and some of the open questions in the survey, a qualitative content analysis was used. In their article Mayring (2000) suggests qualitative content analysis to include some beneficial aspects of quantitative content analysis. Mayring summarizes qualitative content analysis in four parts:

- The part of communication from where the inferences shall be made, should be determined.
- Gradual analyzation should follow the process of inventing analytical units based on the material.
- Interpretations of the text should form categories based on the research questions. This process should be an iterative process which should be revised along with the analysis.
- For the tests for reliability and validity the criteria of the coding should be made understandable for others (Mayring, P. 2000).

These categories and coding rules should be created systematically following an iterative process where the categories and rules are redefined to reach a desired level of reliability and results others (Mayring, P. 2000). The process of defining the categories and rules can be seen in Figure 1 below.

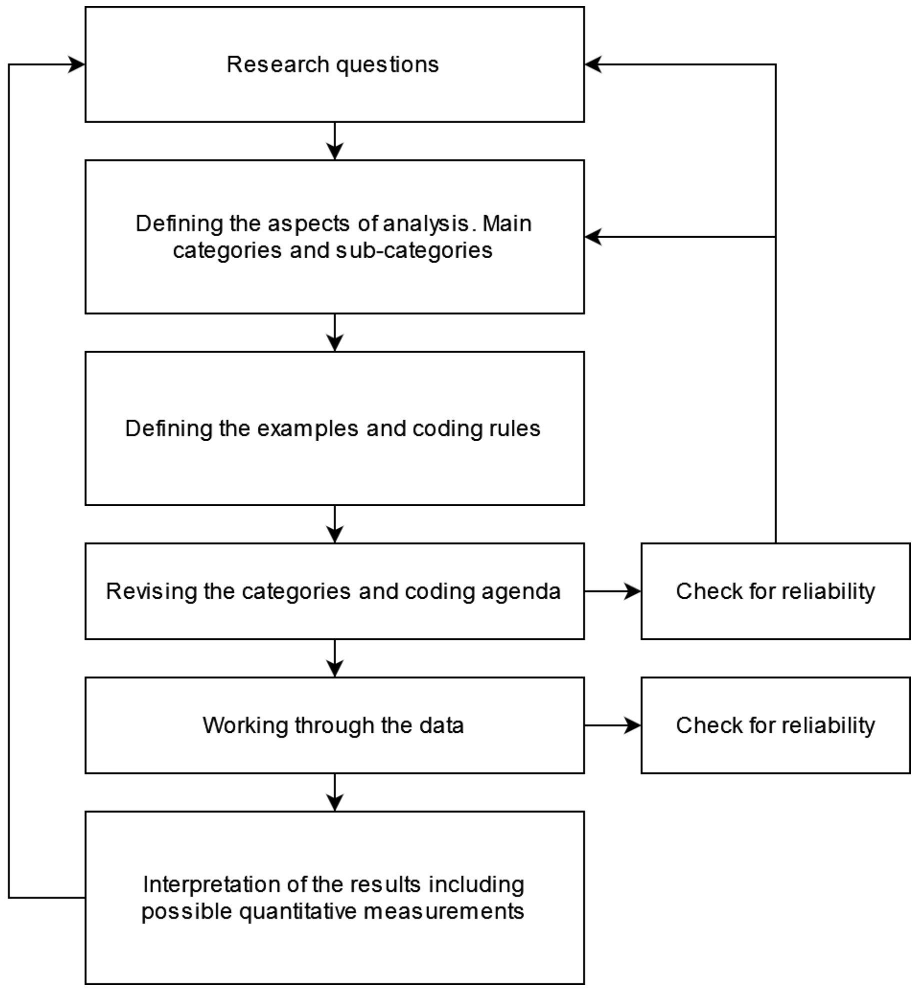

Research questions
Defining the aspects of analysis. Main categories and sub-categories
Defining the examples and coding rules
Revising the categories and coding agenda
Check for reliability
Working through the data
Check for reliability
Interpretation of the results including possible quantitative measurements

Figure 1. Deductive category application. Adapted from “Qualitative Content Analysis” by Mayring, P., 2000, Forum: Qualitative Social Research. Created by the author.

In the context of the conducted theme interview study these categories were referred with the term “themes”. The data gathered from the open questions in the survey is quite condensed, mainly composed of short written responses of one to two sentences, and therefore the analysis process for those questions is simplified compared to the theme interview analysis.

3 Literature review

3.1 Literature review method and process

This literature review briefly summarizes the topic and current studies of low-code/no-code technologies, platforms, and movement. Paré & Kitsiou (2017) describes narrative review methods being the traditional way, usually based more on qualitative analysis. In narrative reviews the assessed studies can be limited, and the selection of articles might be susceptible to subjective biases. On the other hand, this type of review can be beneficial on summarizing a topic and emphasizing new research. In scoping reviews there is an effort to present the extent of literature on a novel topic. The scoping review method can be used to pinpoint research gaps, the Paré & Kitsiou also note in their article. (Paré, et al. 2017). This literature review uses narrative method, but also has some characteristics of a scoping method.

Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) 2020 was chosen to be utilized for the systematic process of the academic literature selection. The method itself is divided into three phases: identification, screening and inclusion (“Home”, 2020).

Records were searched by using ExLibris Central Discovery Index (CDI) and Google Scholar. ExLibris CDI index contains over five billion records of multiple resource types (“An Overview of CDI“, n.d.). Google Scholar is Googles search engine for scholarly literature (“Google Scholar, About”, n.d.).

For the initial phase of identification**,** a search was conducted by using the keyword ‘low-code’. This gathered 42 155 results from ExLibris CDI and 12500 results from Google Scholar. The first 50 results from both platforms were chosen for screening to maintain relevance.

In the initial screening phase 19 results from ExLibris CDI and eight (8) results from Google Scholar were excluded based on duplicates and lack of relevance. When the identified results from both search indexes were combined 73 results were acquired. In these 73 results a substantial number of duplicate resources was recognized, and 29 results was excluded.

In the next screening phase these 44 results were read through and 30 were sought for retrieval. The 14 resources that could not be accessed freely or via access of the university institute were excluded.

The last part of the screening process was to read through these retrieved resources and evaluate their eligibility. Many of these resources had very much in common and the relevance was measured based on two factors: number of citations and the research objectives.

In the end 11 resources were chosen to be included in the literary review. The process can be seen in Figure 2 below.

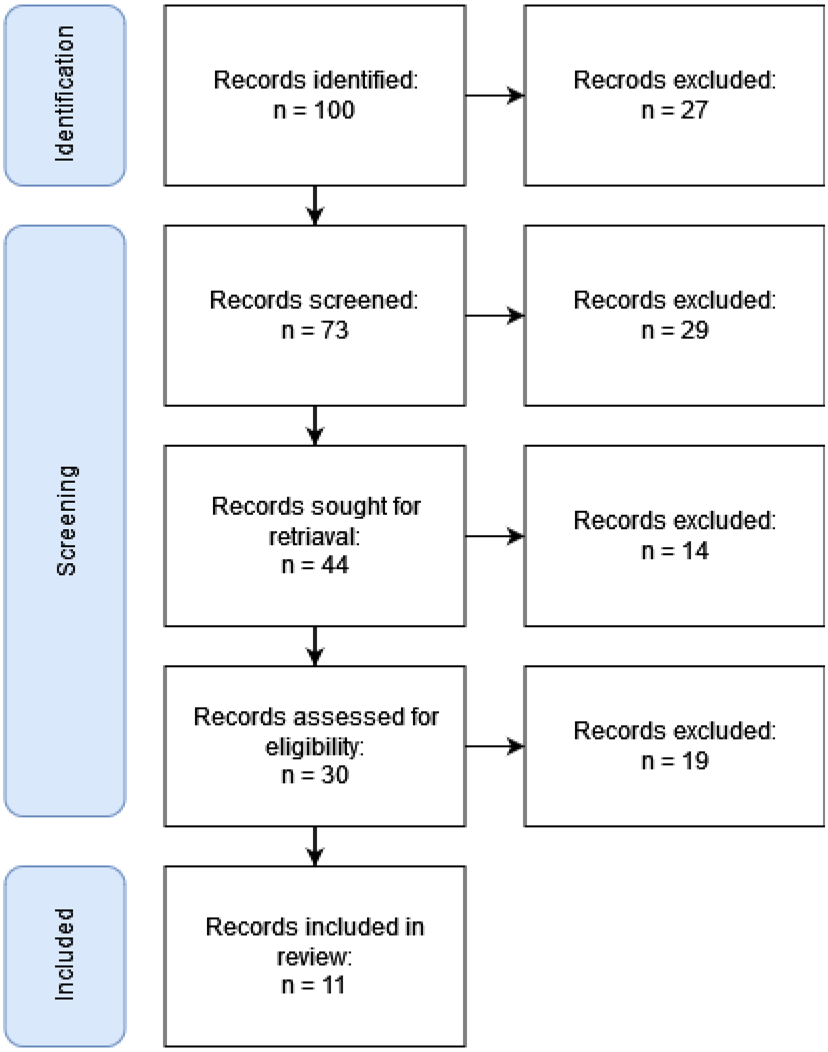


Figure 2. Systematic literary resource process based on PRISMA 2020. Adapted from "PRISMA 2020 flow diagram for updated systematic reviews", 2020. Created by the author.

Some studies, especially the statistics on the adoption of low-code/no-code technologies were industry-driven. These resources were searched using regular search engines and by browsing professional web publications.

When searching for academic resources concerning low-code technologies´ cyber security implications, searches were conducted with the following keywords: "low-code" and "security", "low-code" and "cyber security", and "low-code" and "information security". Although the searches yielded some results it was hard to find purely academic research that had a strong focus on the security implications of low-code technologies. As an example, Table 2 below demonstrates how search results for the searches in IEEE Xplore get scarcer when security-related keywords were combined with “low-code”.

Table 1. Example on scarcity of cyber security related academic literature

	“low-code”	and “security”	and “cyber security”	and “information security”
IEEE Xplore	473 results	27 results	2 results	15 results

Academic research on low-code which included some discussion on security implications was available, but strongly security-focused research was found to be scarce. For that reason, industry-driven research, web articles, blog posts from IT professionals and other non-academic resources were also used for security implications of low-code/no-code technology.

3.2 Low-code/no-code technology

Programming languages can be divided into low-level and high-level languages based on their level of abstraction from machine language. Low-level languages, like assembly or machine code, are closer to the hardware and offer more precise control over the system. In contrast, high-level languages, such as Python, Java, and C#, provide a greater degree of abstraction from the underlying hardware, allowing developers to write code more efficiently using syntax easier to read for human. High level code must be translated for the computer to be able to process it. Figure 3 below divides the language types and the translation of high-level language to a low-level language for the machine (“Types of programming language”, n.d.).

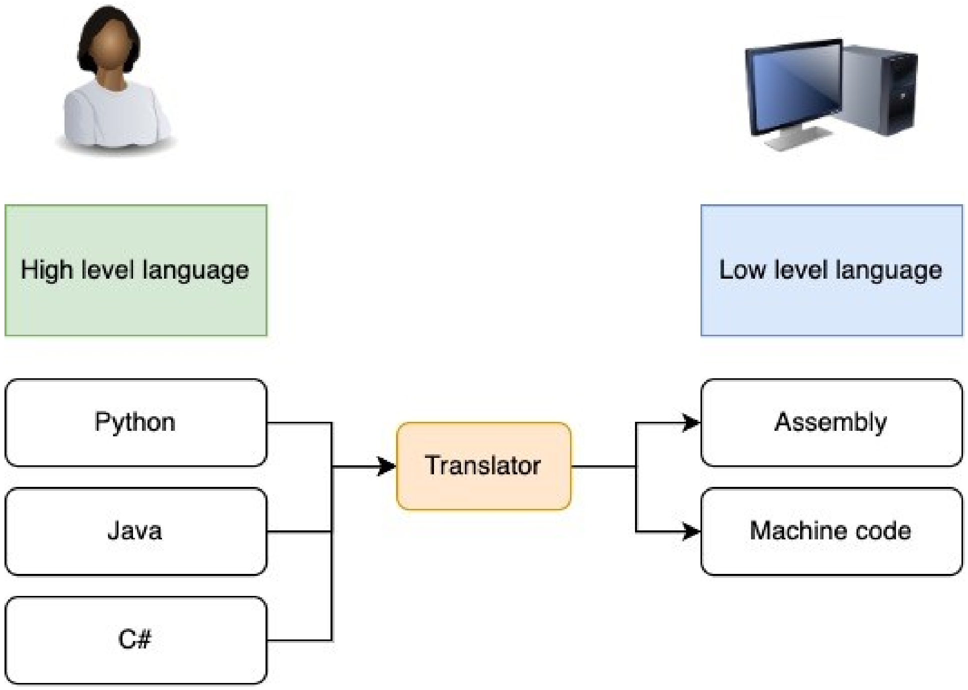


Figure 3. Translation of high-level programming languages to low level computer languages. Created by the author.

In their article Sufi describes the raising level of abstraction which have led up to the current low-code technologies, starting from the machine code for the first electronic numerical integrator and computer to the OutSystems first iteration of low-code platform in 2014. The history of programming language abstraction can be seen in Figure 4 below (Sufi, F. 2023).

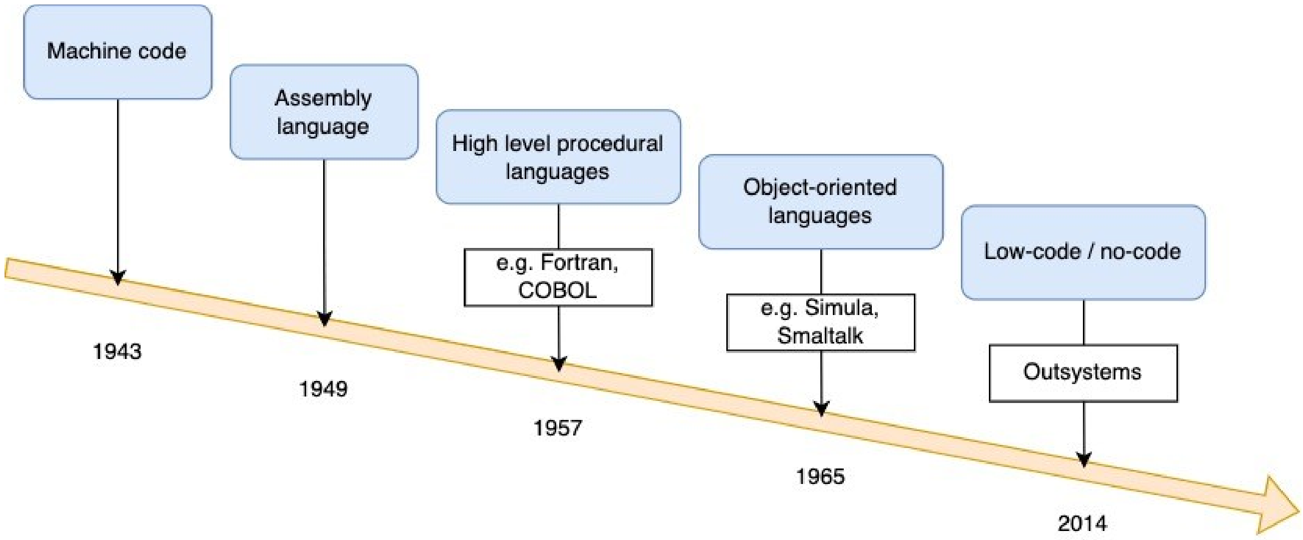


Figure 4. History of programming languages based on Sufis article (Sufi, F. 2023). Created by author.

Low-code/no-code (LCNC) platforms take the concept of abstraction even a step further by reducing the need for traditional programming in favor of visual development environments and prebuilt components. These platforms allow developers to design software and applications using, for example, drag-and-drop interfaces and forms, while automatically generating the underlying code. This higher level of abstraction facilitates rapid application development and reduces the learning curve for non-programmers and simplifies the software creation process (“Low-Code vs. No-Code: What’s the Difference?“, 2022).

These platforms can also be called "citizen automation platforms". They are, usually, cloud-based application platforms which the developer can use via web-based user interface. The difference between LCNC can be summarized so that a low-code platform requires the developer to have some level of knowledge on programming, where no-code platforms require no previous programming knowledge. Figure 5 below demonstrates how a logic to create RSS feed can be created in N8N LCNC platform. The workflow consists of blocks that can be interpreted as functions, which take an input and produce some kind of output or action (“Low-Code vs. No-Code: What’s the Difference?“, 2022).

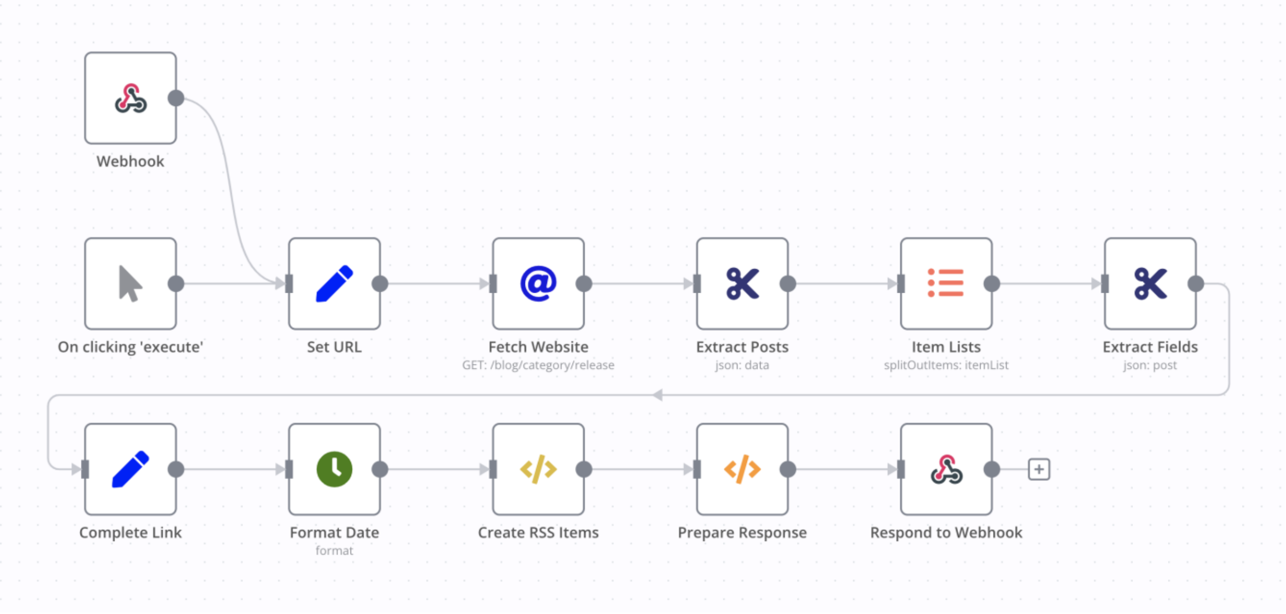


Figure 5. A workflow in N8N LCNC platform producing an RSS feed based on the content of a website. Created by the author.

Low-code development platforms (LCDP) are usually aimed more towards software developers. Their goal is to streamline the development, but to also allow some level of customization by code. No-code platforms on the other hand are aimed more towards so-called "citizen developers". They democratize software development by allowing non-programmers to develop software applications, but they on the other hand allow less customization ("Low-code vs. no-code app development", n.d.).

These novel development platforms did not have a proper name until 2014, when the analyst company Forrester came up with the term “low-code” (Richardson, C., Rymer, J., 2014). A bit later Forbes was the first one to call it, together with no-code, a movement (Bloomberg, J. 2017).

Historically LCDPs have evolved from former Rapid Application Development (RAD) tools. Software that provided some kind of low-code development capabilities for the user are, for example, Excel, Lotus Notes, Microsoft Access (Pratt, M. 2021). In their article on Freecodecamp, Kolade justifies that Wordpress is also a LCNC platform, since it empowers users to create websites with minimal to no code at all (Kolade, C. 2022). Many game engines have included a visual scripting capability, which allows users to create game logic by visually connecting logical nodes, for example (Bay, J., n.d.). Unreal Engine has a visual scripting language called Blueprint. Figure 6 below demonstrates the creation of light switch toggle. The first node sends a signal when an event is being triggered. This signal is received by a toggle node, which toggles between two nodes setting the intensity of a spotlight element.

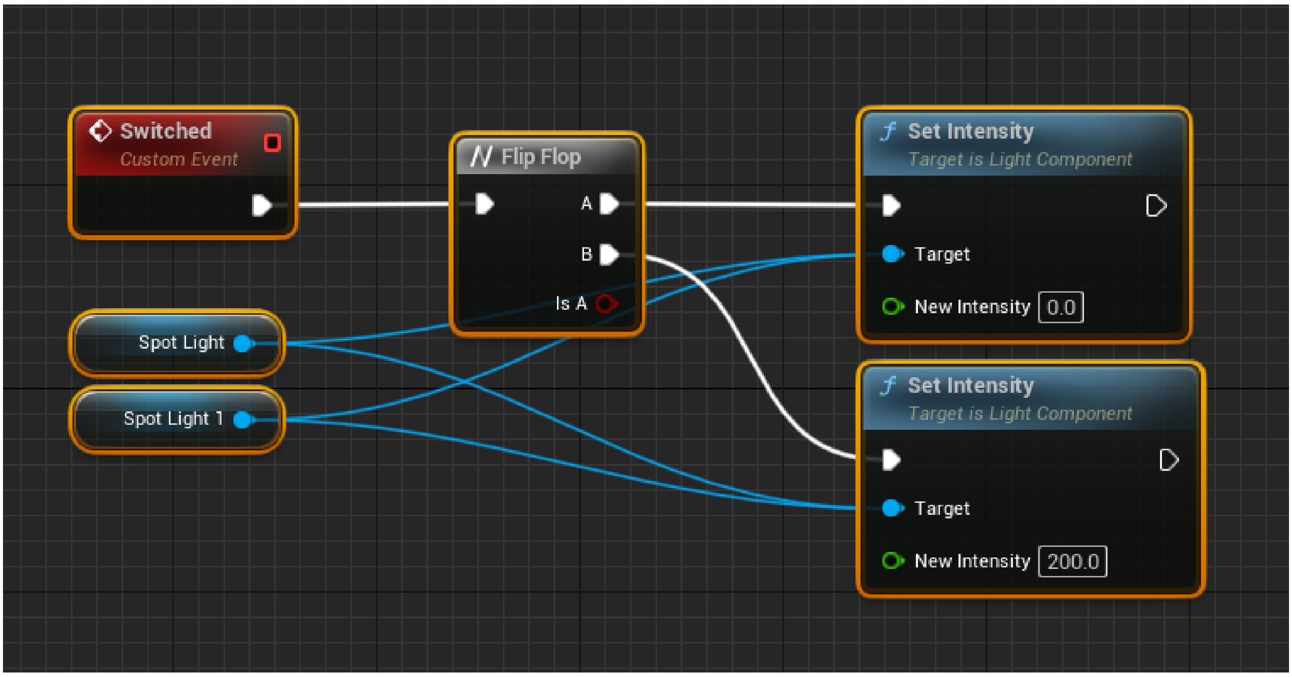


Figure 6. Logic for light switch toggle in Unreal Engine 5 Blueprint visual scripting language. Created by author.

There is currently some ongoing debate on how LCNC technologies differ from model-driven engineering (MDE). In their research Ruscio (et al). states that in model-driven engineering models are used as a first-class artifact, with tasks including code specification, testing, simulation, verification, modernization, maintenance, comprehension, and generation. Together with LCNC platforms MDE also aims to step up in the abstraction level, but not all MDE solutions aim to reduce the amount of code, which can be seen as a major difference between these two technologies (Ruscio, D. et al., 2022).

Because LCNC applications are normal software code under the hood, there are not a lot of limitations where they can't be used. OutSystems implemented a survey study on 3,300 IT professionals (2019) and found out that four highest industry segments using LCNC at the time were:

- software (20%)
- technology/computers/telecoms and internet (16%)
- consultant/consultancy/SI (13%)
- government & education (10%)
  ("The State of Application Development", 2019)

In a latter survey in 2020 OutSystems also found out on what kind of projects LCDP are being used for. The simple biggest use case was “Employee facing portals and web applications” with the score of 49%. Figure 7 below demonstrates the results of the survey.

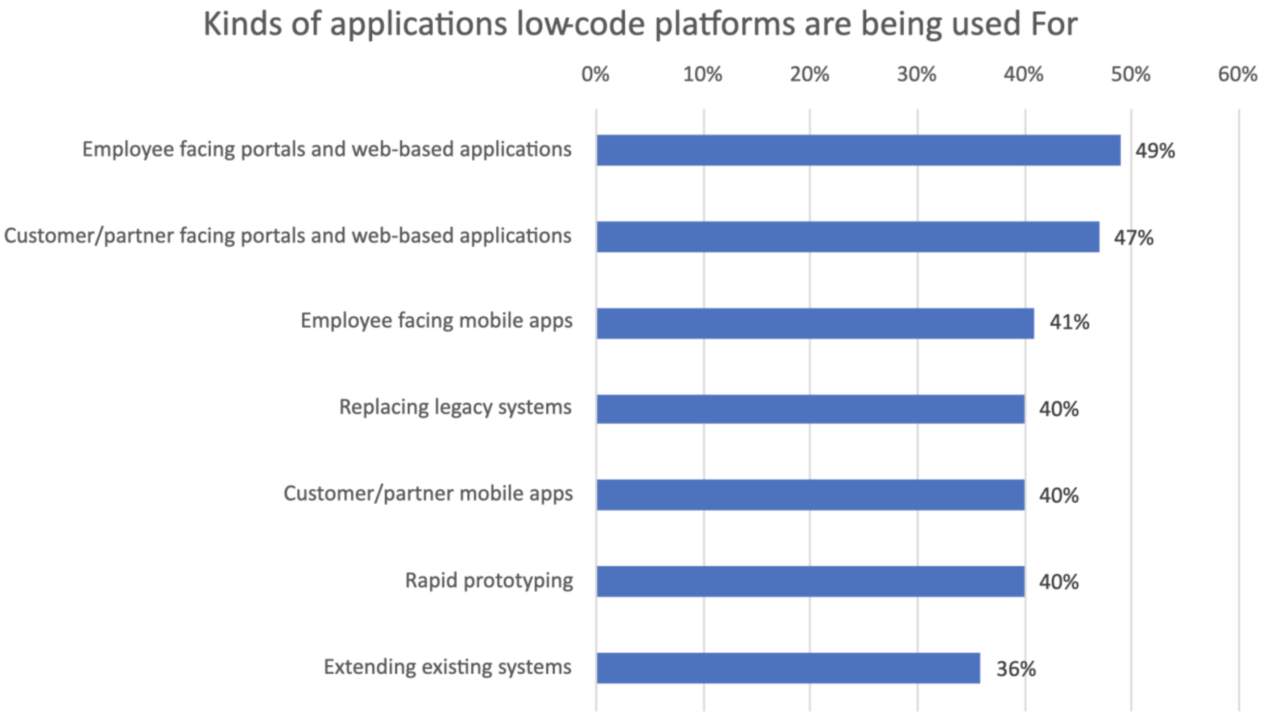


Figure 7. What are low-code used for? Based on the OutSystems 2020 report ("The State of Application Development”, 2020).

Another example of where LCNC technologies can be, and have been, used is supply chain management and supply chain digitalization. In their research Bhattacharyya and Kumar (2021) interviewed supply chain professionals in India. Their findings suggest that LCNC technologies could be used across the whole supply chain digitalization process and could be used as a leverage in the competition against bigger enterprises by small to mid-sized companies (Bhattacharyya, S., Kumar, S. 2021).

LCNC platforms are also marketed based on the speed of the development. In Creatios survey report from year 2021 surveyed adopters of LCNC technology. The responders estimated LCNC technologies to be faster than traditional development. A lot faster, somewhere between 40 to 60 percent ("The State of Low-Code/No-Code.", 2021). Following Figure 8 presents the results of the survey question.

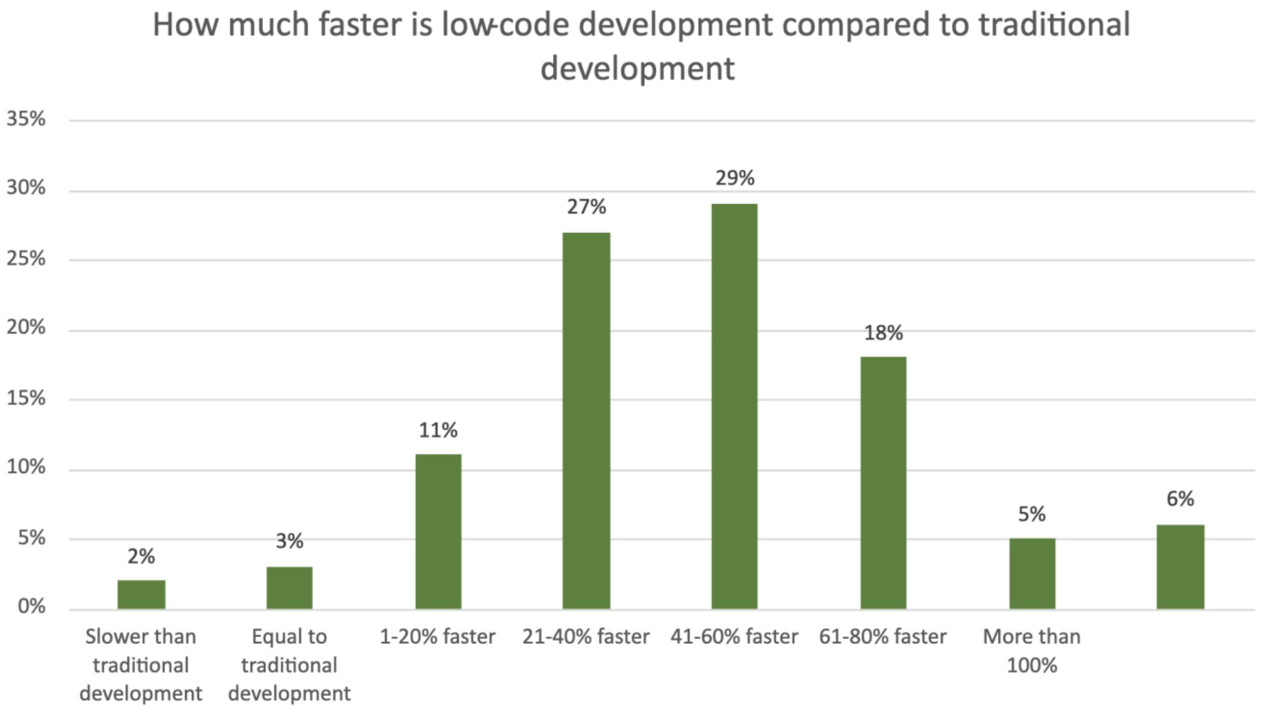


Figure 8. How much faster is low-code development compared to the traditional development based on data from Creatios report ("The State of Low-Code/No-Code.", 2021). Created by author.

Based on Gartners Peer Insights the five most reviewed LCNC platforms were OutSystems, Appian Low-Code Platform, Microsoft Power Apps, Quickbase and Salesforce Platform ("Enterprise Low-Code Application Reviews and Ratings.", n.d.).

LCNC platforms bridge the gap between high-level programming languages and user-friendly interfaces to empower a broader range of individuals to participate in software development, ultimately democratizing the field and fostering greater innovation across industries.

3.3 Adaptation and prospects of low-code/no-code

The acceptance of new technology can be presented with a sociological model called the technology adoption lifecycle. It is based on the theory of E.M. Rogers in his book Diffusion in Innovations. The theory divides the adopters in five categories: innovators, early adopters, early majority, late majority, and laggards (Rogers, E.M. 1995). Based on the percentual amount of these segments in population it is possible to draw a bell curve to present this theory, which can be seen in Figure 9 below.

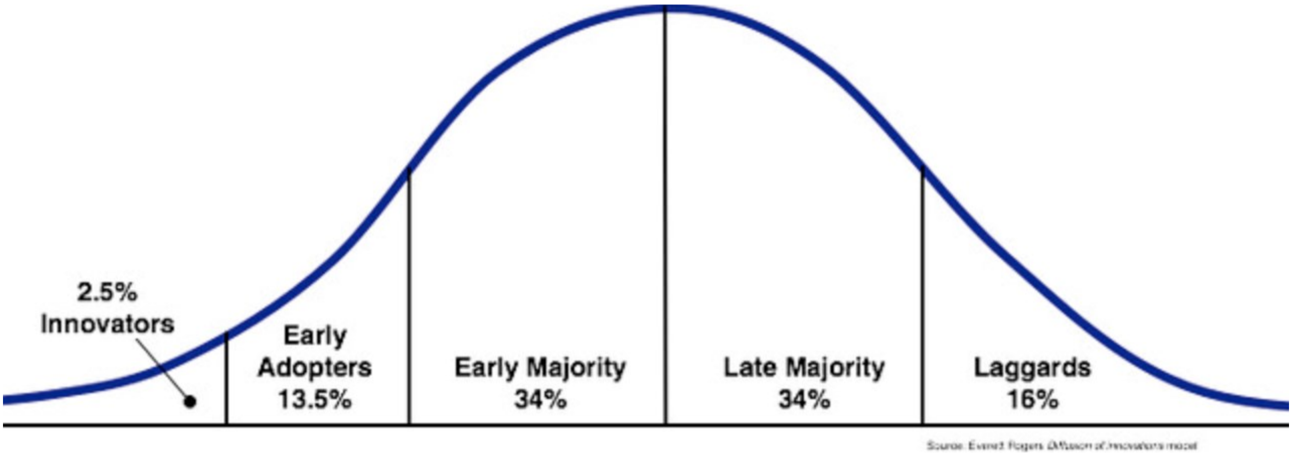


Figure 9. Technology adoption lifecycle. “Diffusion of Innovations” by Wesley, F. Flickr, licensed under CC BY-SA 2.0.

Although it’s currently hard to place LCNC technology adoption in any of these segments, there are several statistics available to estimate the adoption of these technologies.

One example of estimating the rate of adoption is to interpret the yearly revenue of LCNC technologies. Gartner studied the yearly revenues of LCNC technologies and found out that in 2021 the revenue was 18,497 billion (US) dollars, which grew to 22,462 billion dollars in 2022. Gartner predicts the revenue of 2023 to be around 26,869 billion dollars and by 2024 for it to grow as high as to 31,949 billion dollars (“Gartner Forecasts Worldwide Low-Code Development Technologies”, 2022). Research and Markets made the estimation of yearly revenue of these platforms and technologies reaching up to 187 billion dollars in year 2030 (“Low-Code Development Platform Market Research Report”. 2021). These revenue estimations are represented in Figure 10 below.

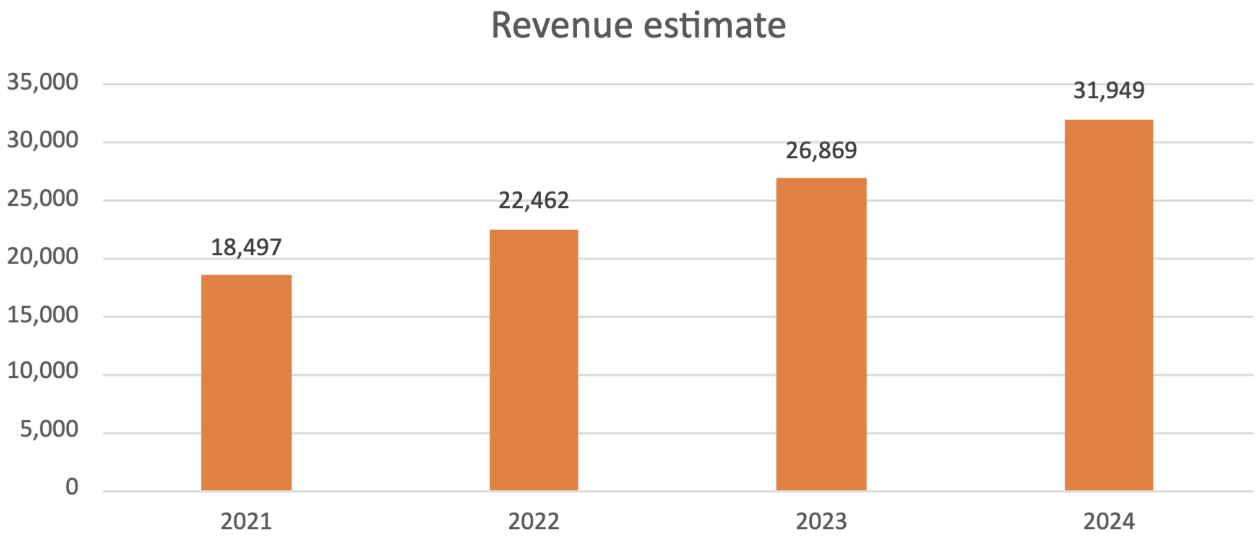


Figure 10. Low-code technology revenues in billion US dollars based on estimations of Gartner. Created by the author.

What are the reasons to adopt these technologies? In their research OutSystems (2020) found these six reasons for businesses to use low-code solutions:

- Speed up digital transformation and innovation.
- Reduce IT backlog and add IT responsiveness.
- Decrease or circumvent legacy debt.
- Lessen reliance on specialized technical skills that are challenging to hire.
- Protect technology from high turnover.
- Empower citizen developers to refine processes ("The State of Application Development", 2020).

The high demand and hardships of recruiting software developers right now across the industries drive the adoption of LCNC technologies. Appian found out in their survey that 82% of companies cannot attract or retain software developers based on their needs (Appian, 2018). This is where the previously mentioned “citizen developers” step into the play. LCNC technologies empower people outside from the traditional IT department to contribute to the application development. In Gartner's Hype Cycle for Digital Workplace estimation in 2020, they stated that citizen developer platforms will hit the mainstream in 2 - 5 years (”6 Trends on the Gartner Hype Cycle for the

Digital Workplace”, 2020). In another survey from Gartner, they found 41% of responding companies to have active citizen development initiatives and 21% of those who did not have, are planning to do so in future (“The Importance of Citizen Development and Citizen IT”, 2019).

But there are still some challenges in adopting these technologies. Tisi et al lists three perceived limitations of LCNC platforms in 2019:

- Scalability: Low-code development platforms (LCDPs) are currently favored for creating smaller applications, however, their ability to support large-scale, mission-critical enterprise applications is a desired next step in their development.
- Fragmentation: Different tool vendors each propose their unique low-code development paradigms, which are often associated with specific programming models.
- Domain-specific systems: While citizen developers may not have extensive programming knowledge, they are frequently experts in other engineering domains. These domain experts anticipate the ability to apply their expertise in applications at an appropriate level of abstraction and using familiar formalisms (Tisi et al. 2019).

In OutSystems report on 2019 they state several reasons why companies are still hesitant on LCNC technologies:

- Insufficient understanding of low-code platforms: 43%
- Apprehension about becoming locked into a specific platform or vendor: 37%
- Belief that low-code does not meet their requirements: 32%
- Doubts about the scalability of low-code applications: 28%
- Concerns regarding the security of low-code applications: 25% ("The State of Application Development", 2020).

LCNC platforms are usually cloud-based services provided by a vendor. This can be a bit daunting because of the possible vendor lock-in. Some of these LCNC technologies offer the possibility for self-hosting the platform in own cloud or even on-premises installations. Europe Commission in its H2020 Framework Programme funded a project called Virtual Factory Open Operating System (vf-OS) Platform. The project was conducted in 2016-2019 and the result was an open framework which can be deployed in-cloud or on-premises, ensuring better control over privacy of development (Sanchis, R. et al., 2019).

In 2016 Forrester published a report concerning LCNC platform vendors. In their report Richardson and Rymer figured out in 2015 that most of the vendors they kept track of were small businesses. This lowers the price of entry into the LCNC but does present a problem with big company strategies and standards in the long run. In the report they provided an assumption of large vendors acquiring LCNC platforms (Richardson & Rymer, 2016). That assumption proved to be correct, for example, Microsoft acquired a company called Zionsville in 2021 and implemented their low-code technologies into their Power Apps service (Mackie, K. 2021). In their report Richardson and Rymer also provided some recommended actions for companies trying to minimize possible risks and limitations concerning these platforms. They suggest, for example, prioritizing feature set to suit the needs of the enterprise, choosing a vendor able to sustain innovation and value, and setting up conventions and governing policies for LCNC development (Richardson & Rymer, 2016).

In their paper Overeem and Jansen (2021) studied the API maturity of four LCNC platforms. Authors used the API-m-FAMM API maturity model to pinpoint limitations in the API management of these platforms. They found out that these platforms supported only about 50% of the practices presented in the API-m-FAMM model. These limitations hinder the possibility for citizen developers to develop applications without the need of professional IT support. In their paper, they present suggestions for LCNC platform vendors on how to develop the API maturity further (Overeem & Jansen, 2021).

Even though LCNC platforms have some technical limitations and there are some challenges in adoption of these technologies, LCNC platforms are a growing market, and they are gaining popularity in various industries.

3.4 Security implications of low-code/no-code technologies

Before we investigate security implications of LCNC technologies, we need to be clear on terminology. In their article Galarita (2022) clarifies the difference between the terms of cyber and information security. To simplify, information security (also referred to as infosec) covers the security and authorization of data. Cyber security on the other hand focuses on protecting electronic communication services and devices (Galarita, B. 2022). The National Institute of Standards and Technology (NIST) maintain definitions for these terms. NIST definitions of ‘information security’ and ‘cyber security’ can be read from Table 2 below.

Table 2. Definitions of terminology by NIST. Adapted from National Institute of Standards and Technology.

Term	Definition
information security	"The protection of information and information systems from unauthorized access, use, disclosure, disruption, modification, or destruction in order to provide confidentiality, integrity, and availability." ("Glossary: information security", n.d.)
cyber security	"Prevention of damage to, protection of, and restoration of computers, electronic communications systems, electronic communications services, wire communication, and electronic communication, including information contained therein, to ensure its availability, integrity, authentication, confidentiality, and nonrepudiation." ("Glossary: cyber security", n.d.)

In this chapter both areas of security are addressed under the umbrella term 'security'.

In their review Cavelty summarizes cyber security to include all the practices we execute to mitigate the insecurities in our so called "cyber space" which we interact with in our daily lives (Cavelty, M. 2015). Wiley lists new innovations and technologies and describes some possible issues in cyber security perspective in their book. For example, current cryptographic algorithms are deemed unsafe when it comes to quantum computers, or how cyber attackers have developed better tools which make digital forensics more difficult than in the past. Computing power moving to cloud has numerous benefits, but this forces enterprises to measure their trust on the cloud service providers (Wiley, J. & Sons, 2022). This is relevant, since low-code platforms are usually cloud services, therefore cloud computation adds another layer to the consideration of security on these technologies.

LCDP streamlines processes and focuses on ease of use. While this accelerates and democratizes the development process, it may also inadvertently lead to increased risky behavior. In their white-paper Zenity lists 7 most commons concern for low-code applications and some tips to mitigate them.

- **Privilege Escalation:** LCDPs are often run with the personal identity of the developer. While accessing the application, developers might gain access to underlying credentials. This might grant access to resources not authorized for the developer. Developers should avoid running applications with user identities, but with least-privileged access permissions system roles.
- **Data Leakage:** LCDPs allow the user to transfer data through multiple points. If one such point is, for example, an unauthorized service or a personal hard drive data leakage is possible. This risk can be minimized by limiting connection to authorized connections only, or by utilizing data loss prevention policies of the platform in use.
- **Insecure authentication:** Since the users of LCDPs are often not professionals in data security, it is possible that, for example, weak protocols such as HTTP or weak encryption might be used. Zenity (2021) suggests checking possible FTP(s) and HTTP(s) connections and making sure they are using the more secure means of authentication.
- **Misconfigurations:** LCDPs offer a wide range of features and configuration options. Platforms often provide guidelines for the configuration which should be noted to minimize the risks.
- **Dependency injection:** LCDPs often allow users to create pre-made components and publish them in the marketplace. This creates a possible risk of dependency injection. Developers should carefully review any custom components fetched from such a marketplace, for example.
- **Oversharing:** Sharing of apps, components or data is made easy by design in LCDPs, sometimes organization-wide sharing is default option. Since oversharing is easy, Zenity (2021) suggests making sure if all the administrators are needed, if there’s more than two in an app in production. It is also crucial to set the security settings so that they grant the user only the access needed for the application or the data.
- **App impersonation:** Low-code apps developed by the organization might seem trustworthy for the users. LCDPs allow external deployment, for example, from the marketplace. By enforcing a consent process, it is possible to make these apps more secure. (“The 7 Deadly Sins of Low-Code Security and How to Avoid Them”, 2021).

The Open Worldwide Application Security Project, shortened OWASP, is an internationally recognized non-profit foundation which aims to improve the security of applications. OWASPs activities include community-led open-source projects, members and local chapters worldwide and educational and training conferences (“About the OWASP Foundation”, n.d.). OWASP community has produced a list of 10 most present security risks in LCNC technologies. In their list OWASP community also rates the prevalence, detectability, exploitability and technical impact of these potential risks. The list contains a lot of points included in the previous list by Zenity, but also a few additional concerns.

- **Injection handling failures:** Low-code apps consume data from users with various kinds of inputs. For example, from direct input or from retrieving it from other services. It is possible that such data could include a malicious payload. Input sanitization should be used for humans, but also for inputs performed by application.
- **Asset management failures:** LCDPs allow users to easily create a relatively low-cost app. This might lead it to be easy to abandon applications which are still running in production. It is also possible for single employee to create an internal app which gains a lot of users. When this person is no longer working with the company the application might not have anyone maintaining it. To prevent these concerns OWASP suggests keeping up an inventory of apps, components and users, and removing or disable anything unused.
- **Security logging and monitoring failures:** LCNC apps might not have a proper audit trail, produce sufficient logs or maybe even overshare logs deemed sensitive. It might be cumbersome to locate who produced a certain change in the logic. Developers should utilize the capabilities to collect user and platform audit logs, utilize logging mechanisms and ensure the logs do not contain any sensitive data (OWASP Low-Code/No-Code Top 10, n.d.).

The OWASPs Low-Code/No-Code Top 10 list is utilized in the survey study conducted for this thesis and it is referred to with greater detail in chapter 4.

Spets studied OutSystems LCDP by comparing it with Application Security Verification Standard (ASVS) security standard created by the OWASP community in their master's thesis. The standard provides organizations with means to develop and maintain the security in their applications, and to negotiate with LCDP vendors, for example. The ASVS standard can be divided into three levels, starting from the requirements considered a minimum in any application in level 1. Level 2 in the standard provides companies with means of defending against most of the modern threats. Level 3 is the most secure and is usually used to maintain a high security in critical infrastructure, for example, health care and military. Spets used Level 2 of the standard for the comparison. They noted that even though OutSystems is mostly compliant with the standard, there were some concerns in following areas:

- **Authentication:** OutSystems does not provide multifactor authentication (MFA) itself, but as a downloadable module. Also, U2F tokens used in physical authentications methods need to be developed by the user.
- **Input validation:** OutSystems LCDP warns developers from potential injections and provides them with sanitization tools, they do not enforce them (Spets, S. 2022).

In 2021 researchers from UpGuard (2021) found out that 38 million sensitive records were being exposed to the public internet containing Covid-19 contact tracing information. The data was stored on Microsoft Power Apps portal service, since the applications that the data was related to were created on the platform. This all comes down to misconfiguration of API privileges by the users. When users enabled Open Data Protocol (OData) APIs without setting Table Permissions it exposed the data to anonymous users by default. They found out that the possibility for the misconfiguration was documented but noted that a mere warning in the documentation of this possible misconfiguration is not sufficient ("By Design: How Default Permissions on Microsoft Power Apps Exposed Millions.", 2021). This is an unfortunate example of potential misconfiguration in low-code app development.

Governance is important when it comes to low-code application development. As stated earlier, citizen developers are a growing user base among LCNC platform users. These developers need more governing than regular software developers from the IT department. In their article Schwartz (2021) specifies it is crucial to create and enforce security guidelines but enable enough freedom to feed productivity. It is also important to be sure everyone working on the application is working through the low-code development platform. With citizen developers together with the IT department creating apps, there can be a lack of visibility on what is being created. One solution would be creating a sandbox for citizen developers where the resources and components are known. Another problem with citizen developers might be the forementioned misconfigurations. Having automatic processes identify exposed data and poorly set security protocols helps with this concern (Schwartz, K.D. 2021).

Most popular LCNC development platforms are cloud-based services, therefore it's important to address the security issues regarding cloud computing, some which Tissir (et al) (2020) describes in their work. Since cloud services gather masses of digital resources under the same infrastructure, it makes them intrigue for potential attackers. Cloud computing has risks under both categories, cyber security and information security (Tissir, N. et al. 2020). In 2013 Cloud Security Alliance (CSA) published an article called "The Notorious Nine", which lists nine top threats of cloud computing.

1. Data Breaches
2. Data Loss
3. Account Hijacking
4. Insecure APIs
5. Denial of Service
6. Malicious Insiders
7. Abuse and Nefarious Use

8. Insufficient Due Diligence
9. Shared Technology Issues ("Cloud Security Alliance Warns Providers of 'The Notorious Nine'", 2013).

These threats are in addition to LCNC platform specific threats and need to be addressed when using cloud-based services.

With all the possible risks aside, LCNC technologies can also bring benefits concerning security. In their article Newcomer (2022) argues that LCDPs can make it easier to implement secure authentication methods, since they can be pre-built components provided in the platform (Newcomer, E. 2022). Peruzzi argues that organizations can be more agile and react faster to newcoming problems. For example, a way of organization to respond to an emerging threat is to quickly create custom low-code solution if existing security software cannot support in preventing or stopping a threat (Peruzzi, J. 2023).

While LCNC development platforms and the introduction of citizen developers might bring some security concerns, the risks can be minimized with proper governing, security guidelines, sandboxing, automated security tests or by other means.

3.5 Summary of the literature review

The purpose of this literature review was to gather an insight into LCNC technologies based on the research objectives and questions. Definition, history, evolution and the core concepts of low-code and no-code technology and platforms were described. Insight was also gathered on a number of statistics about adoption and near prospects of this technology. Finally, the security implications were discussed in the last chapter.

Based on the findings about the prospects of LCNC technologies it is safe to say that the adaptation of these tools is growing fast. Concrete actions to minimize possible security concerns were also found.

The biggest uncertainty was in the security implications chapter. Proper academic research which had a strong focus on cyber or information security was scarce and therefore other less academic sources were used.

The literature review brought crucial information for the formation of the survey and interview studies. Especially the insight on cyber security implications helped to formulate the cyber security sections of the survey and the theme interviews. It also helped to form the categories for the qualitative content analysis technique performed on the qualitative data.

With the understanding that was gained it is safe to move to the studies conducted for the thesis.

4 Results

4.1 Survey

The survey study was conducted in the timeframe of 1.12.2022 – 18.12.2022. It was opened 352 times, 86 people have started the survey and 76 full and unique responses were received. The survey is structured to follow the research objectives.

The survey results can be logically divided into three sections following the research objectives, in addition to some basic information about the respondent and their company. The last section of cyber security implications was available only for those participants who expressed their experience on cyber security, hence it gathered less responses. In this chapter we will divide the findings as follows:

- Basic information of the responder
- Adaption of low-code/no-code technologies (RQ1, RQ2)
- Cyber security professionals’ perspective on the technology (RQ3)

Most of the questions were mandatory to respond. Some of them had options like “don’t know” or “don’t wish to answer”. Unless stated otherwise, all 76 participants answered the question in the following chapters.

4.1.1 Basic questions

Question 1: How old are you?

This question gives us some information about the age distribution of the participants. With this data we can assume possible student status or work experience in general, for example.

The question form was a text box which accepted only integers. 75 participants answered this question out of 76.

- Minimum age: 22
- Maximum age: 55
- Average: 33,7
- Median: 33
- Standard deviation: 6,7

Question 2: What is the level of your education?

This question gives us insight on the education level of low-code/no-code technology users. It was asked via multiple choice form where participants had to choose one from the following options: college or vocational level degree, bachelor's degree, master's degree and doctors' degree. The percentual results can be seen in Figure 11 below.

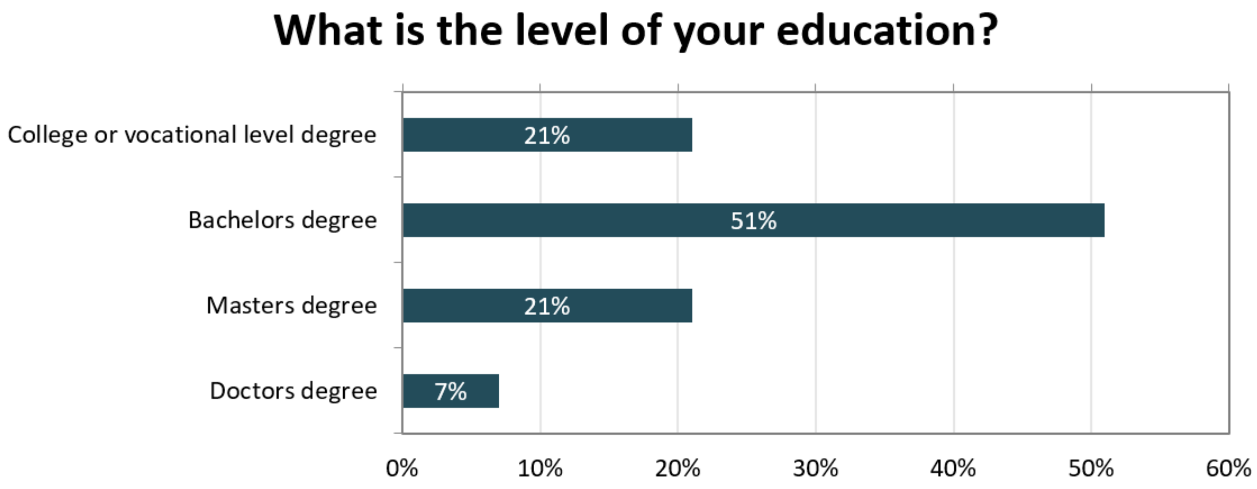


Figure 11. What is the level of your education?

This question gathered 76 answers. The largest group of responders had an educational level of bachelor’s degree (51%). 21% did not have a bachelor level of education. This might partly stem from bachelor level students responding to the survey.

<u>Question 3:</u> Where do you currently live?

This question aims to give us insight into where low-code/no-code is being utilized or adopted. It was asked via multiple choice form where participants had to choose one from the following options: Asia, Africa, Europe, North America, South America, Middle East and Australia/Oceania. The percentual results can be seen in Figure 12 below.

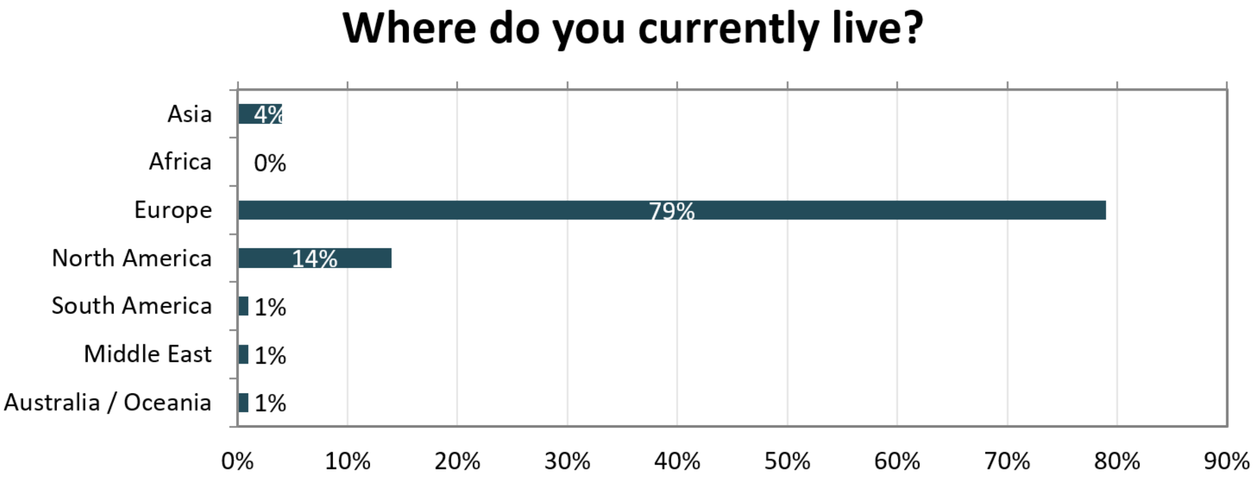


Figure 12. Where do you currently live?

This question gathered 76 responses. The answers came mostly from Europe. This was assumed based on the reach of the social media campaign and the network of the author. South America, Middle East and Australia/Oceania all had just one responder. Table 3 below presents the numerical results of this survey question.

Table 3. Where do you currently live?

Continent	Responses
Asia	3
Africa	0
Europe	59
North America	10
South America	1
Middle East	1
Australia / Oceania	1

<u>Question 4:</u> Which (of these) describes your department or company the best?

This question aims to gather insight into what kind of companies or departments LCNC technologies are used or considered in. With this information it is possible to compare results between software and cyber security departments or companies, for example.

It was asked via multiple choice form where participants had to choose one from the following options: software department/company, cyber security department/company and others. The option “Other” was provided for possible situations where job descriptions are hard to define. The percentual results can be seen in Figure 13 below.

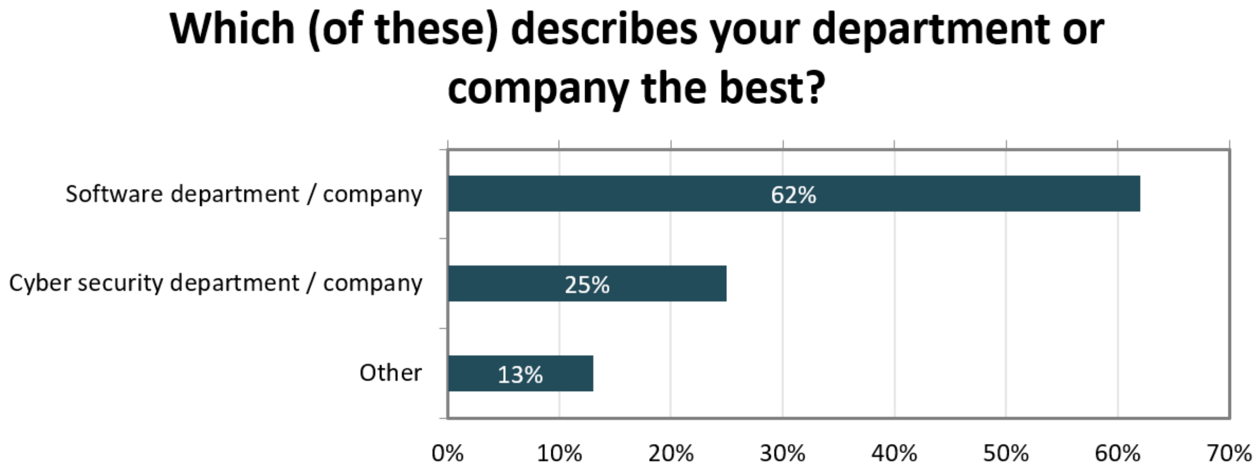


Figure 13. Which (of these) describes your department or company the best?

This question gathered 76 responses. Most of the respondents (62%) defined their department or company as a software department, or company. 13% of the participants chose the “Other” option.

The results divided as follows:

- Software department / company: 47
- Cyber security department / company: 19
- Other: 10

This question had a sub-question if the option “Other” was selected:

<u>Question 5:</u> If you answered 'other' to the previous question, would you like to describe your company?

This question aims to give us insight about what other IT / cyber security related fields are interested in these technologies. This was asked via open textbox form, and it gathered following descriptions:

- Education/University
- Executive management
- Education
- Industrial
- University
- Marketing
- JAMK
- Currently unemployed, but studying in a university of applied sciences
- Higher educational university
- I work in administration

Five (5) of these responses stated the participant to be working in an educational institute of some sort.

<u>Question 6:</u> How many people does your company employ?

This question gives us insight on the size of companies utilizing or considering LCNC technologies. It is possible to find out the differences in responses between small and big companies by filtering the results.

The division of the options are based on European Commissions definition of small and medium-sized enterprises (SME), which states that companies up to 250 employees fall under this category (“Commission Recommendation of 6 May 2003”, 2003).

The question was asked via multiple choice form where participants had to choose one from the following options: 0-10, 10-50, 50-250, 250-500 or “more than 500”. It was also possible for the participant to choose 'Do not wish to answer' if not willing to answer. All the options and the percentual results can be seen in Figure 14 below.

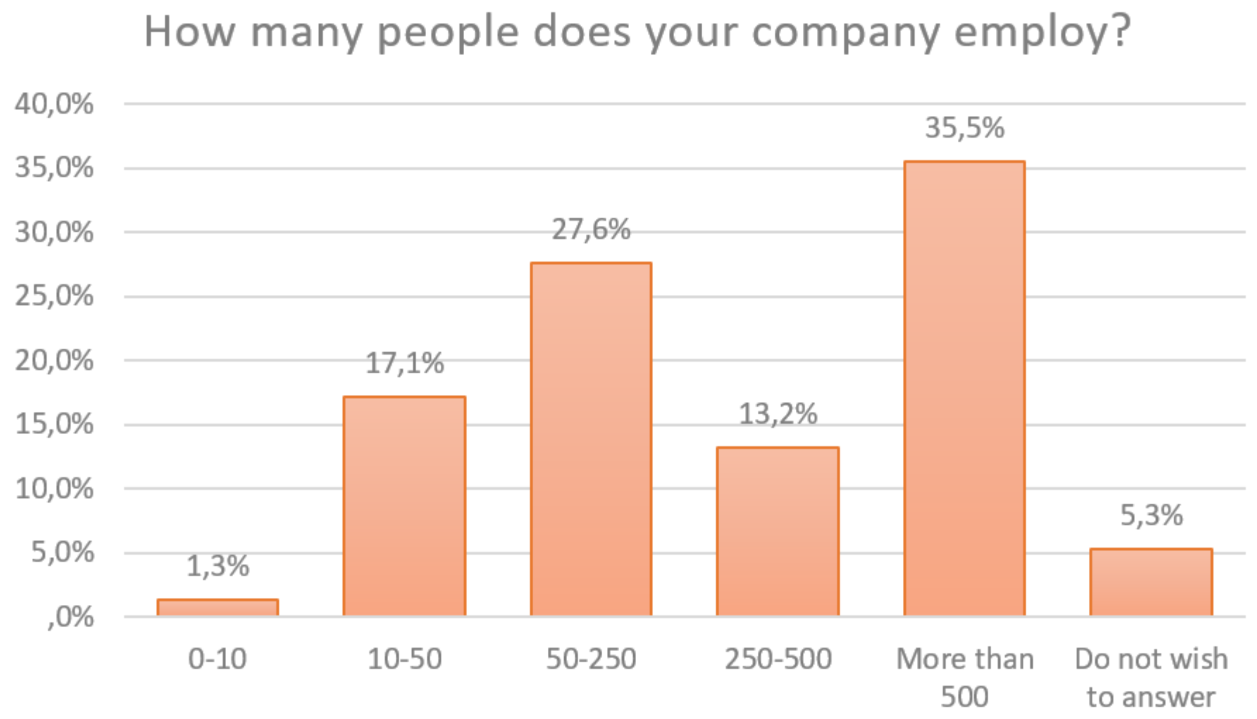


Figure 14. How many people does your company employ?

This question gathered 76 responses. Most participants (35,5%) stated their company employs over 500 employees. Table 4 below presents the survey results in numbers.

Table 4. How many people does your company employ?

1-10	10-50	50-250	250-500	More than 500	Do not wish to answer
1	13	21	10	27	4

<u>Question 7:</u> How familiar are you with software development in general?

In this question the participants estimated their level of general familiarity in software development. With this information it is possible to filter responses based on familiarity with software development.

Participants were asked to estimate their level of familiarity on a scale from one (1) to five (5) via multiple-choice form. In this scale option one (1) stands for "not familiar at all" and five (5) stands for "very familiar". The percentual results can be seen in Figure 15 below.

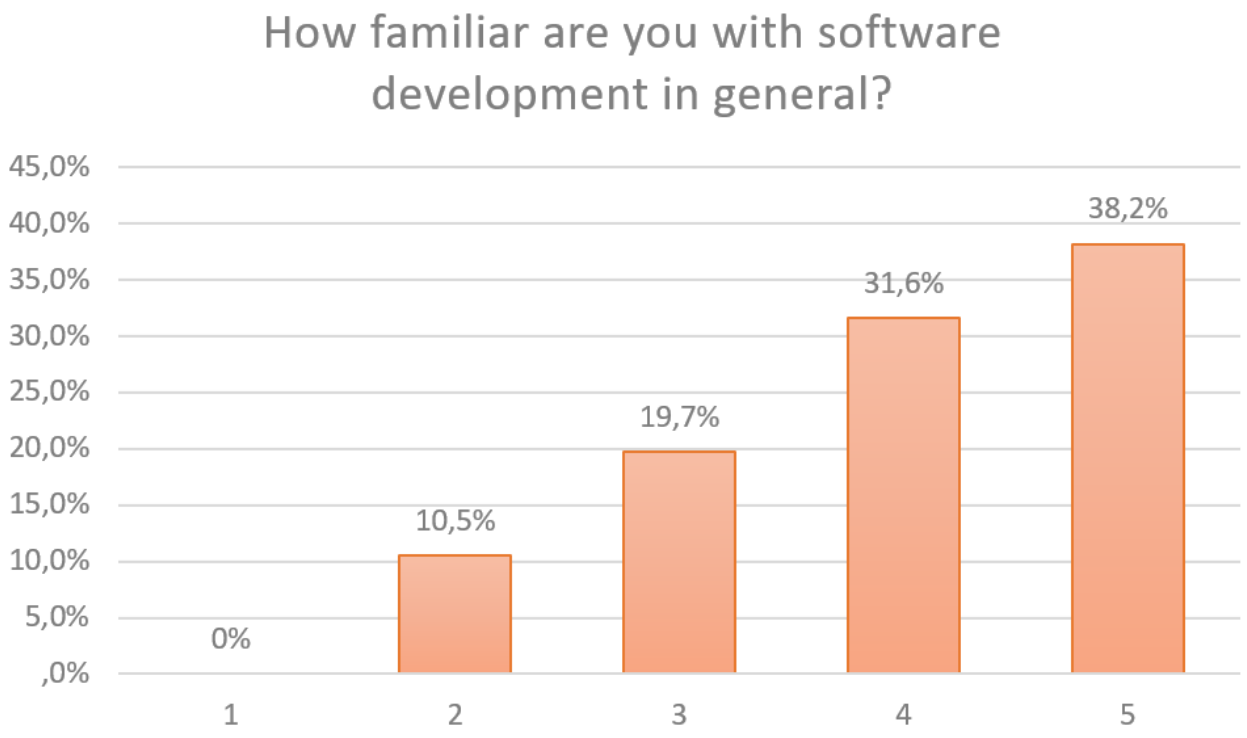


Figure 15. How familiar are you with software development in general? Option one (1) stands for "not familiar at all and five (5) for "very familiar".

This question gathered 76 responses. The average and the median values both were 4,0. The survey results in numbers can be read from Table 5 below.

Table 5. How familiar are you with software development in general?

1	2	3	4	5		Av	Median	Total
0	8	15	24	29		4,0	4,0	76

<u>Question 8</u>: Does your work involve software development?

This question clarifies if the responder is actively involved with software development. The purpose of this question is to gather data for filtering purposes. We can, for example, compare the responses of software developers and others.

It was asked via simple multiple-choice form, with options “yes” and “no”. The percentual results can be seen in Figure 16.

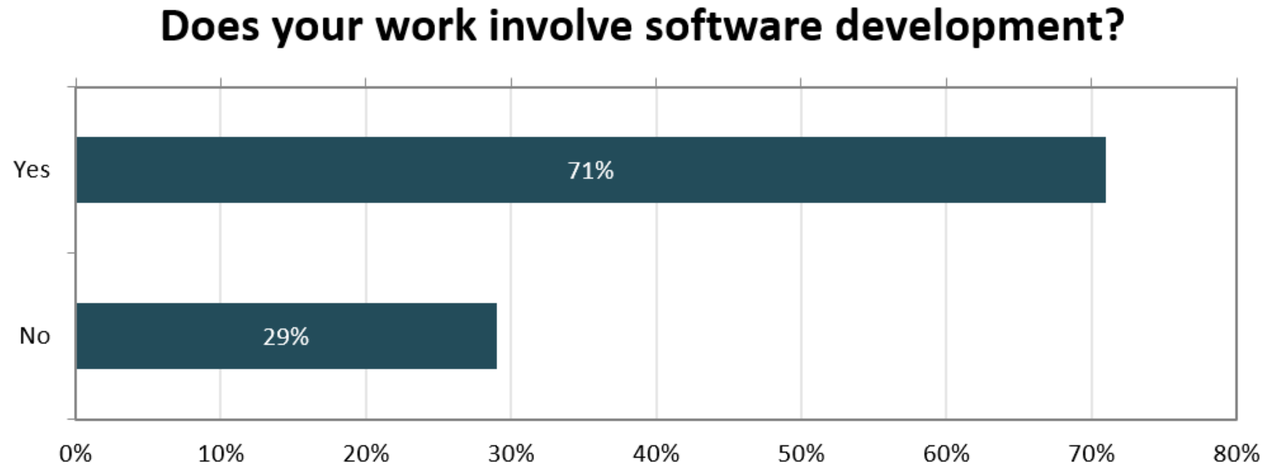


Figure 16. Does your work involve software development?

This question gathered 76 responses. Most of the participants (71%) are involved in software development. 29% of the participants are currently not involved in software development. Results of this survey question can be read in numbers from Table 6.

Table 6. Does your work involve software development?

Yes	No		Total
54	22		76

<u>Question 9</u>: How experienced are you as a software developer?

In this question the participants were asked to estimate their experience as a software developer in years. With this information it is possible to filter responses based on the experience level of the respondents.

The question was a multiple-choice form where the participant was asked to choose one of the options. All the possible options and the percentual results can be seen in the Figure 17 below.

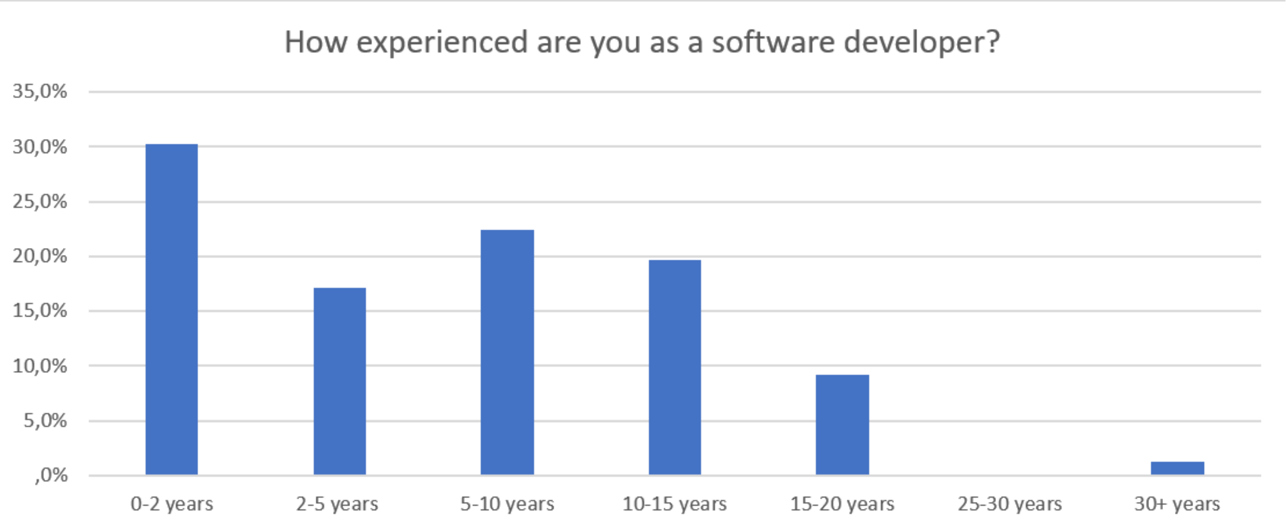


Figure 17. How experienced are you as a software developer?

This question gathered 76 responses. The average response was 2,7, so the average experience of a participant can be placed in the 5-10 years category. The median was 3,0.

4.1.2 Adaptation and prospects of low-code/no-code technology

Question 10: How familiar are you with low-code/no-code software development paradigm or technologies?

In this question the participants estimated their level of familiarity in low-code/no-code software development paradigm or technologies. The purpose is to find out how well technology is known in the field.

Participants were asked to estimate their level of familiarity on a scale from one (1) to five (5) via multiple-choice form. In this scale option one (1) stands for “not familiar at all” and five (5) stands for “very familiar”. The percentual results can be seen in Figure 18.

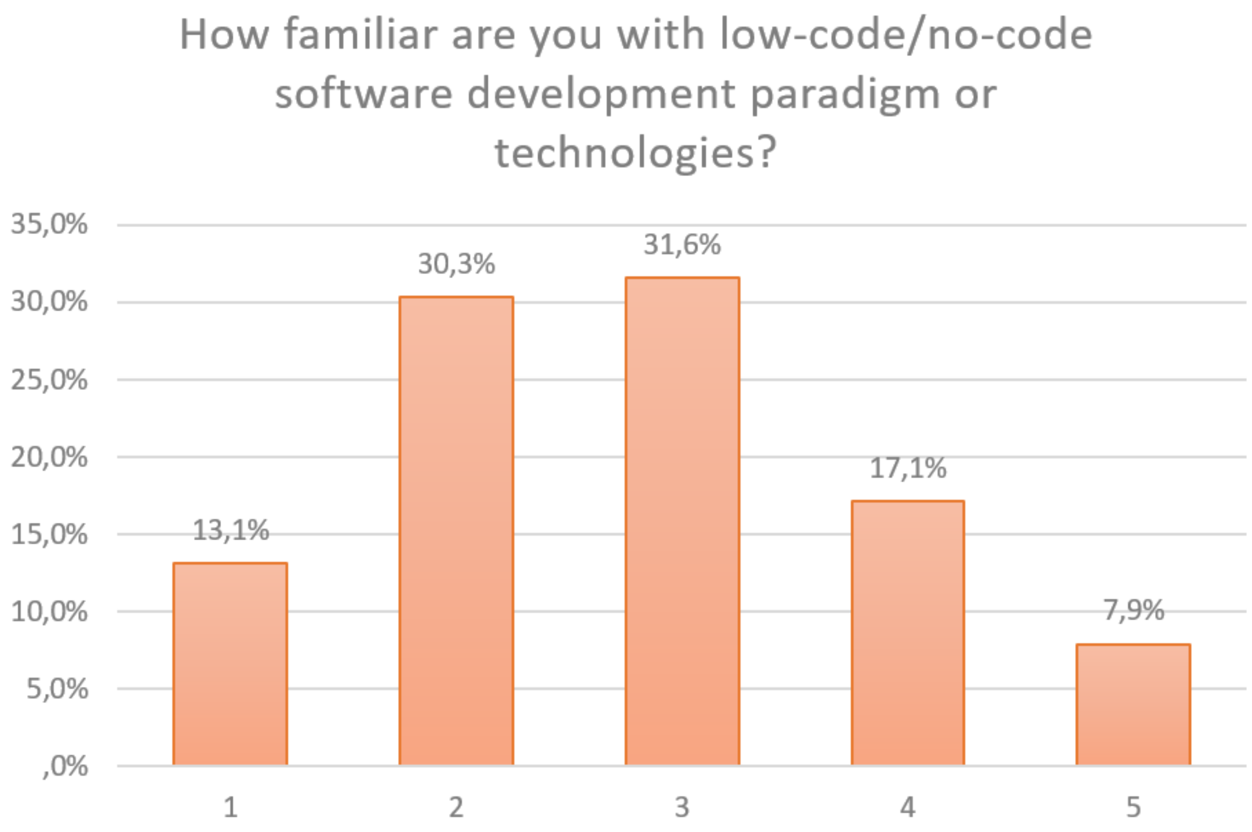


Figure 18. How familiar are you with low-code/no-code software development paradigm or technologies? Option one (1) stands for "not familiar at all" and five (5) for "very familiar".

This question gathered 76 responses. The two highest results in this question were option two (2) with 30,3% and option three (3) with 31,6%. 13,1% of the participants were not familiar at all with low-code/no-code technologies and 7,9% participants stated being very familiar with these technologies. See Table 7 below for results in numbers.

Table 7. How familiar are you with low-code/no-code software development paradigm or technologies?

1	2	3	4	5		Average	Median	Total
10	23	24	13	6		2,8	3,0	76

<u>Question 11:</u> Is your company already utilizing low-code/no-code software development?

In this question the participants were asked if their company already uses low-code/no-code technologies. The purpose of this question was to forward participants toward more relevant questions based on their use of low-code/no-code in software development.

Responses were asked via multiple-choice form with options “yes”, “no” and “don’t know”. The participant was forwarded based on the option they chose. If the participant chose option “yes” they will skip forward to question number 14. If they chose option “no” they continued to question number 12. And if they chose option “don’t know” they were forwarded to question number 15. The percentual results can be seen in Figure 19.

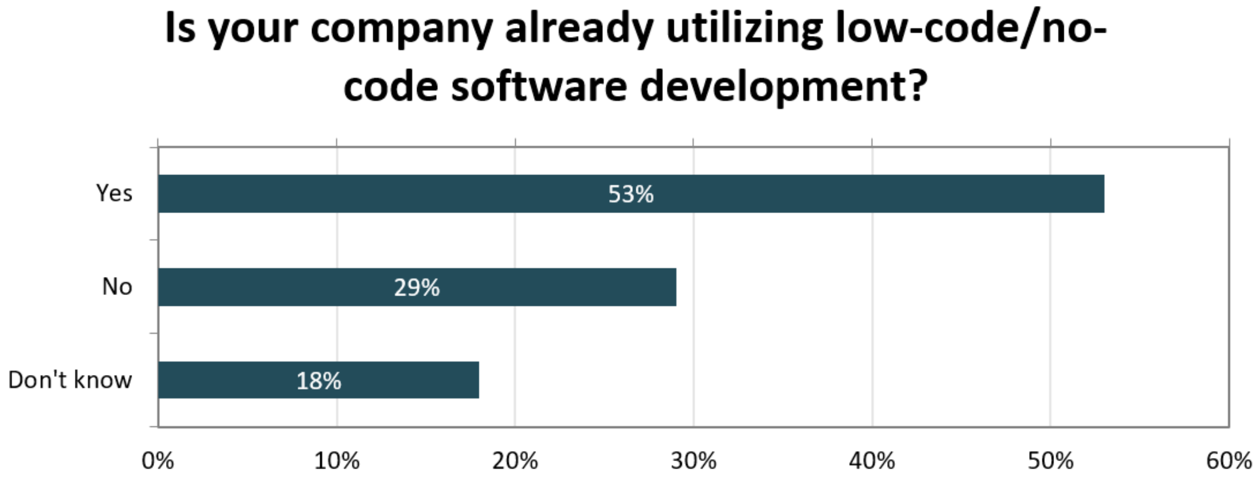


Figure 19. Is your company already utilizing low-code/no-code software development?

This question gathered 76 responses. More than half of the participants stated their company was already utilizing low-code/no-code software development, 29% knew that their company did not utilize these technologies and 18% did not know if they were utilizing them. See Table 8 below for the results in numbers.

Table 8. Is your company already utilizing low-code/no-code software development?

Yes	No	[redacted]	Don’t know
40	22	[redacted]	14

<u>Question 12:</u> When would you roughly estimate your company start utilizing low-code/no-code technology?

This question was asked if the participant selected “No” in previous question 11. The purpose of this question is to find out if the participants company might be adopting low-code/no-code technologies in future.

In this question the participants were asked to roughly estimate when they could see their company starting to utilize low-code/no-code technologies via multiple-choice form. Options in this form were: “1-5 years”, “5-10 years”, “10-15 years”, “later than 15 years”, “never” and “not sure”. In Figure 20 below you can see the percentual results, excluding the options which gained no responses.

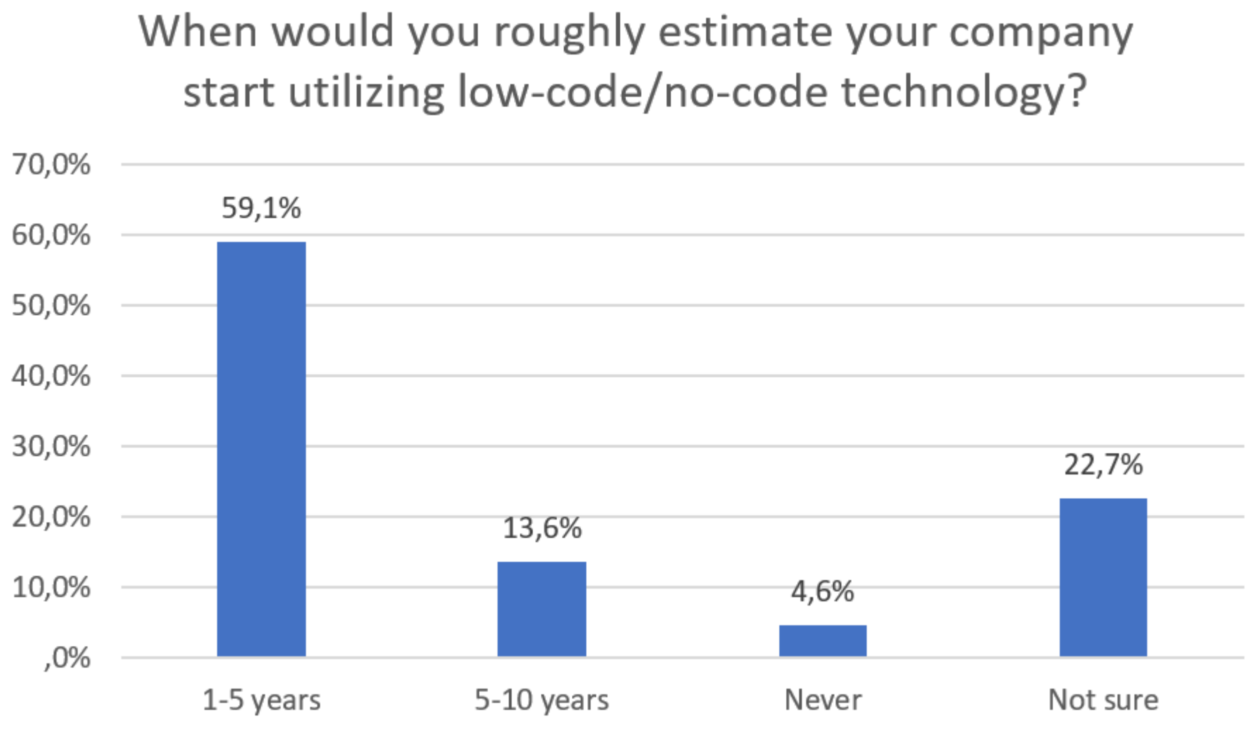


Figure 20. When would you roughly estimate your company start utilizing low-code/no-code technology?

This question gathered 22 responses. Over half (59%) of the participants chose the option "1-5 years". There were also options for 10-15 and later than 15 years, but they gathered no responses.

This question had a sub-question if the options "Never" or "Not sure" were selected:

<u>Question 13:</u> If you answered 'Never' or 'Not sure' in the previous question, would you like to explain why?

The purpose of this sub-question is to gather reasoning behind the hesitancy of using these technologies. It can also provide us with possible barriers to adopting LCNC in software development. The open-text question gathered following six (6) justifications:

- Not suitable for our purposes
- Our core development tasks focus on the single service our company provides, so we always have to take legacy code into account on some level and hence don't adopt new technologies very often.
- We need flexibility in our products that low-code/no-code can't provide. With low-code/no-code debugging and changing things on the fly is a no go.

- I've never heard of low-code/no-code platforms where you could do serious programming, delivering complex applications for customers. My initial feeling is that these platforms would be suitable for more lightweight development, but my initial judgement may very well be wrong.
- At first glance I can see how low-code/no-code could be useful for our development teams, but LC/NC has not really been seriously evaluated as a tool and I'm not familiar enough with the technology to make an estimate.
- We build integrations that needs coding

Question 14: How well does low-code/no-code technology serve your company's needs?

This question was open only to those participants who stated their company already utilizing low-code/no-code technologies. The purpose of this question is to figure out how well low-code/no-code technologies serve the need of the participant’s company.

In this question the participants estimated how well the low-code/no-code technologies serve their company’s needs on a scale from one (1) to five (5). The option one (1) stands for “not at all” and option five (5) for “very well”. The question was asked via multiple-choice form and the participant could choose only one option. This question gathered a total of 40 responses. See Figure 21 for results.

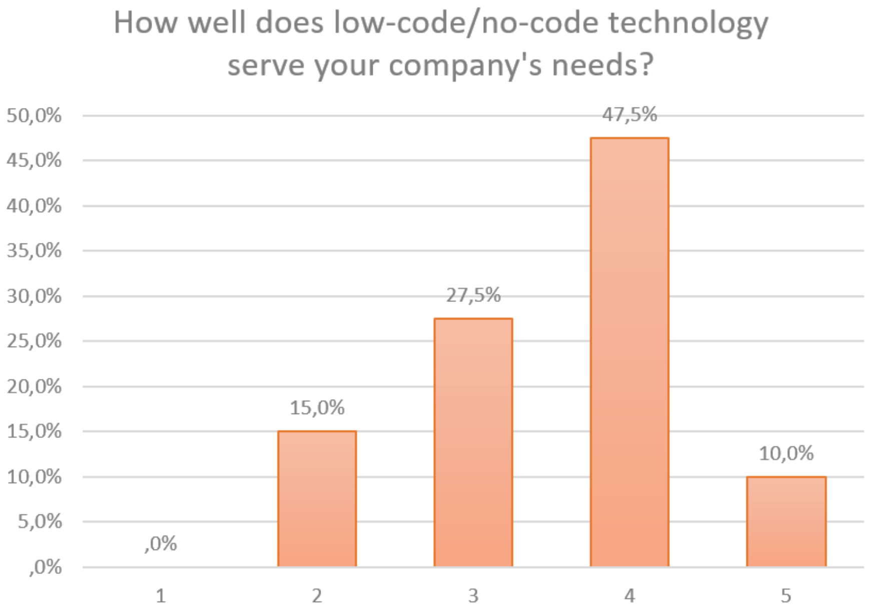


Figure 21. How well does low-code/no-code technology serve your company's needs? Option one (1) stands for “not at all” and five (5) for “very well”.

This question gathered 40 responses. The option with the most responses (47,5%) was option four (4). Not a single participant chose option one (1) which stands for “not at all”. The results in numbers can be seen in Table 9 below.

Table 9. How well does low-code/no-code technology serve your company's needs?

1	2	3	4	5		Average	Median	Total
0	6	11	19	4		3,5	4,0	40

This question had a sub-question:

Question 16: Please describe how does your company utilize low-code technologies.

The purpose of this sub-question is to figure out how low-code/no-code technologies are being utilized in companies. This question gathered 26 responses. The response was given via open-text form and the results can be read from Appendix 2.

Question 15: How well would low-code/no-code technologies serve your company's needs?

This question was open only for those participants who stated their company is not utilizing, or they don’t know if it’s utilizing low-code/no-code technologies currently. The purpose of this question is to find out if the participant might see benefits or use for these technologies in future.

In this question the participants estimated how well the low-code/no-code technologies would serve their company’s needs on a scale from one (1) to five (5). Option one (1) stands for “not at all" and option five (5) stands for "very well". This question gathered a total of 36 responses. The results are presented in Figure 22 below.

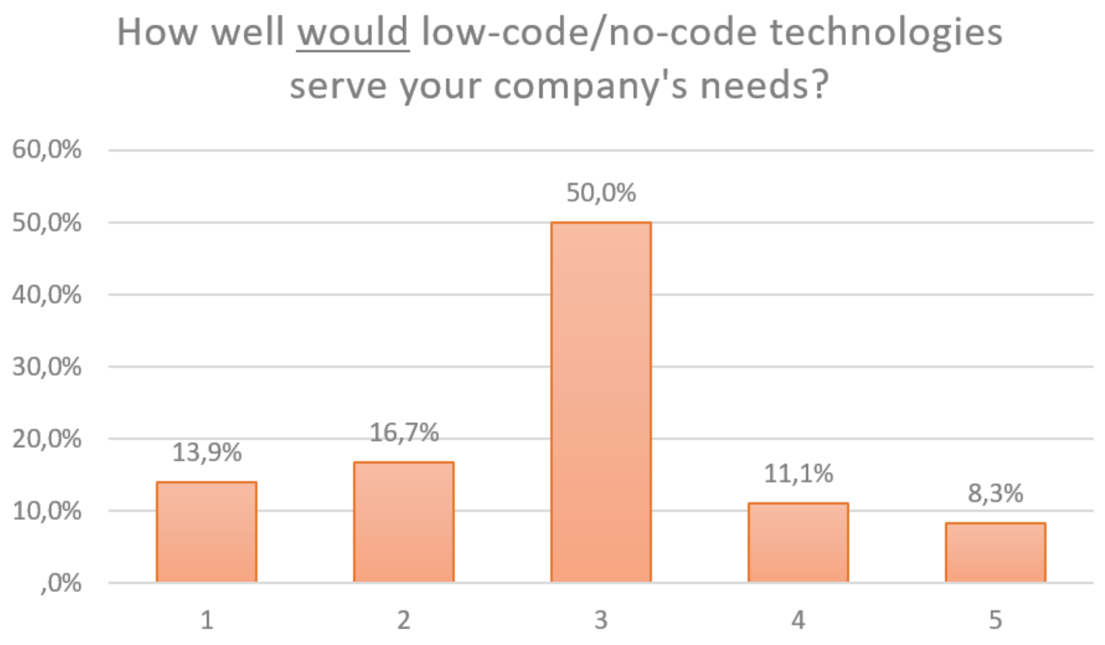


Figure 22. How well would low-code/no-code technologies serve your company's needs? Option one (1) stands for "not at all" and five (5) for "very well".

This question gathered 36 responses. The largest group of participants (50,0%) chose option number three (3). A clear difference can be seen in these results compared to the results of participants already using LCNC technologies in the previous question (14). The data is presented in numbers in Table 10 below.

Table 10. How well would low-code/no-code technologies serve your company's needs?

1	2	3	4	5		Average	Median	Total
5	6	18	4	3		2,8	3,0	36

This question had a sub-question:

Question 17: Please describe how could your company utilize low-code technologies.

The purpose of this sub-question is to figure out how low-code/no-code technologies could be utilized in companies. The answers might provide possible use cases where low-code/no-code technologies are seen beneficial. The response was given via open-text form and the results can be read from Appendix 3. The question gathered 18 responses.

4.1.3 Cyber security implications

The last section of this survey was based on cyber security professionals' perception on the LC/NC technologies. The participants who stated their knowledge on cyber security were presented with a link to the OWASP Low-code/No-code Top 10 list.

The participants were also presented with the formula describing how cyber risk is calculated:
likelihood * impact = risk

When addressing the security implications based on the OWASPs list, the participants were asked to estimate the **likelihood** of the following risks.

Participants were also provided with brief explanations of the risks they are estimating. These explanations were direct quotes from the OWASP Low-code/No-code Top 10 list, so that the results are as comparable as possible.

Question 18: Do you have any knowledge on cyber security?

This question led to the final section of the questionnaire. The purpose of this question was to forward people with experience on cyber security to the last section. For the participants with no experience on cyber security the survey was finished.

The response was provided via multiple-choice form with option "yes" and "no". The percentual results can be seen in Figure 23 below.

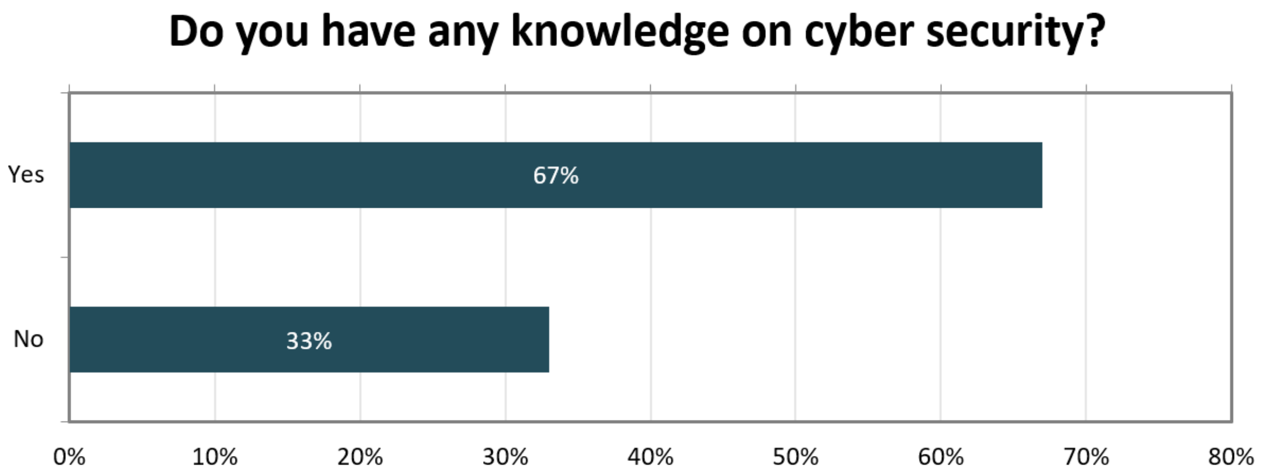


Figure 23. Do you have any knowledge on cyber security?

Out of 76 responses 51 participants had at least some level of knowledge on cyber security while 25 did not. For those 25 participants the survey was over after this question.

<u>Question 19:</u> How experienced are you in cyber security?

In this question the participants were asked to evaluate their experience on cyber security in years. The purpose of this question was to provide the level of experience of the participant which can be used for filtering and analysis.

The response was provided via multiple-choice form with options: “1-5 years”, “5-10 years”, “10-15 years”, “15-20 years” and “more than 20 years”. The percentual results can be seen in Figure 24 below.

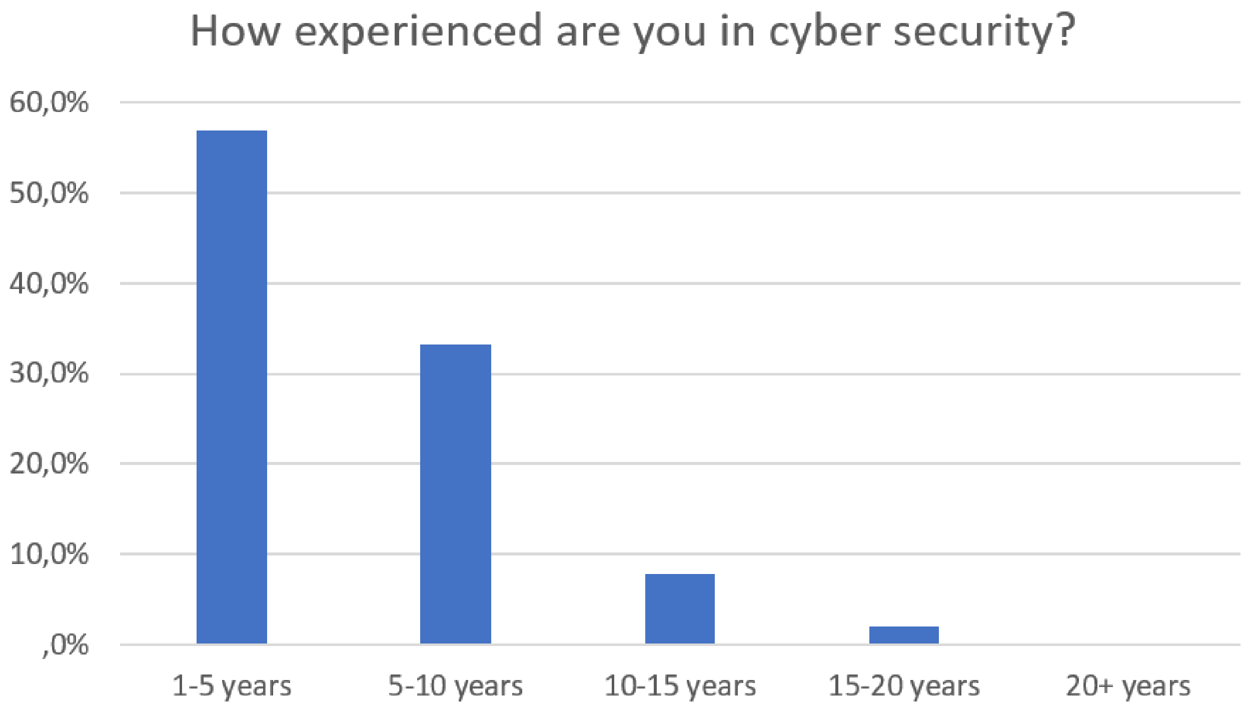


Figure 24. How experienced are you in cyber security?

This question gathered 51 responses. Not a single participant estimated their experience to be over 20 years. The average response was 1,5 and the median 1,0.

<u>Question 20:</u> Account Impersonation

The purpose of this question is to find out the estimate likelihood of the risk of account impersonation. The response was provided via multiple-choice form where the users estimated the likelihood by selecting one option on the scale of one (1) to five (5). The option one (1) stands for “least likely” and the option five (5) stands for “most likely”. Participant were also provided with the following description:

> *“No-code/Low-code applications can be embedded with a developer account which is used implicitly by any application user. This creates a direct path towards Privilege Escalation, allows an attacker to hide behind another user's identity, and circumvents traditional security controls” (“LCNC-SEC-01: Account Impersonation Risk Rating”, n.d., para. 1).*

The percentual results can be seen in Figure 25 below.

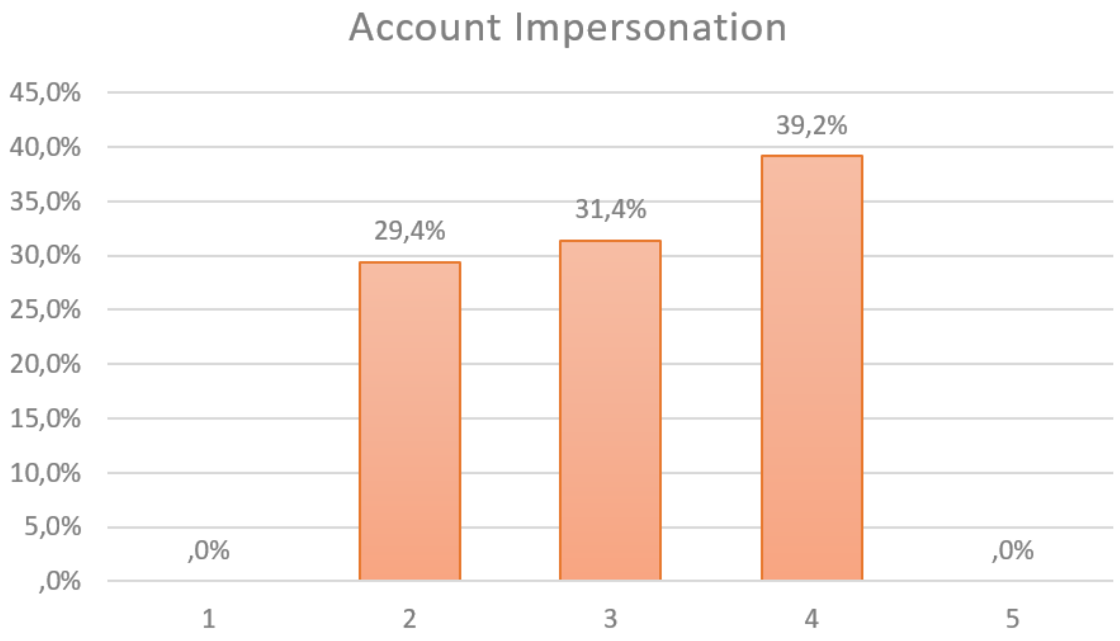


Figure 25. Account impersonation. Option one (1) stands for "least likely" and five (5) for "most likely".

This question gathered 51 responses. The largest group of participants (39,2%) responded with option four (4). Options one (1) and five (5) were not selected by any participant. The results in numbers can be seen in Table 11 below.

Table 11. Account Impersonation.

1	2	3	4	5		Average	Median	Total
0	15	16	20	0		3,1	3,0	51

<u>Question 21:</u> Authorization Misuse

The purpose of this question is to find out the estimate likelihood of the risk of authorization misuse. The response was provided via multiple-choice form where the users estimated the likelihood by selecting one option on the scale of one (1) to five (5). The option one (1) stands for "least likely" and the option five (5) stands for "most likely". Participant were also provided with the following description:

> *"Connections are first-class objects in most no-code/low-code platforms. This means connections between applications, other users, or entire organizations. Applications can also be shared with users who should not have access to their underlying data" ("LCNC-SEC-02: Authorization Misuse", n.d., para. 1).*

The percentual results can be seen in Figure 26 below.

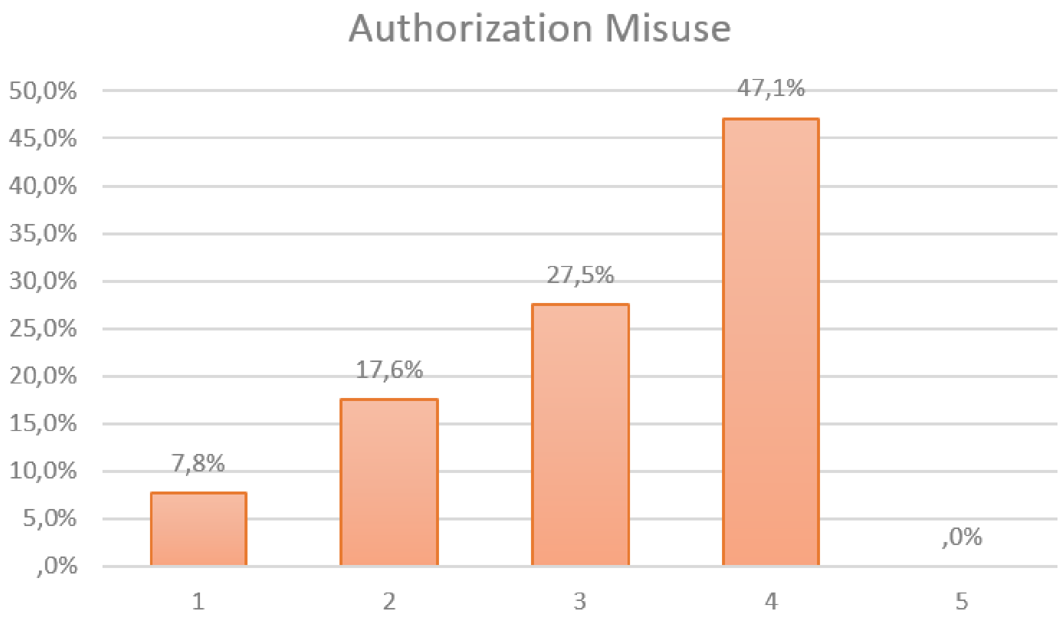


Figure 26. Authorization Misuse. Option one (1) stands for "least likely" and five (5) for "most likely".

This question gathered 51 responses. Option four (4) was the most chosen option (47,1%). Not a single participant estimated the risk to be "most likely". See Table 12 below for results in numbers.

Table 12. Authorization misuse.

1	2	3	4	5		Average	Median	Total
4	9	14	24	0		3,1	3,0	51

<u>Question 22:</u> Data Leakage and Unexpected Consequences

The purpose of this question is to find out the estimate likelihood of the risk of data leakage and unexpected consequences. The response was provided via multiple-choice form where the users estimated the likelihood by selecting one option on the scale of one (1) to five (5). The option one (1) stands for “least likely” and the option five (5) stands for “most likely”. Participant were also provided with the following description:

> *”No-code/low-code applications legitimately access data from underlying services but can also serve as a conduit to those backend systems for actions that were not anticipated or approved of. This includes unintended side effects such as data leakage beyond the application/security boundary; triggering create, read, update or delete operations on the data; or accidental/malicious data exfiltration” (”LCNC-SEC-03: Data Leakage and Unexpected Consequences”, n.d., para. 1).*

The percentual results can be seen in Figure 27 below.

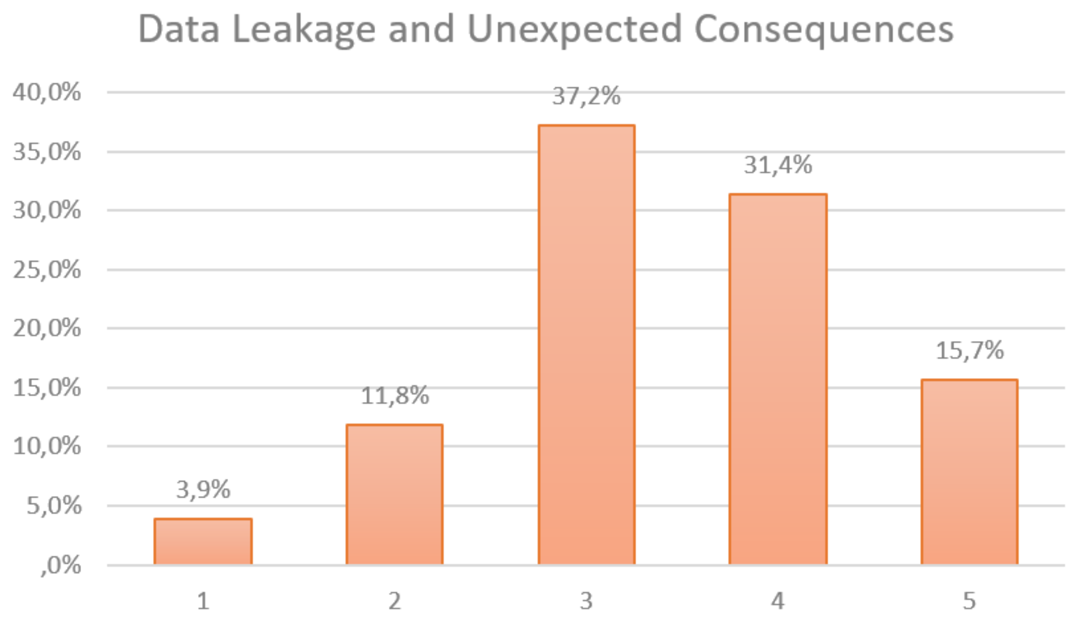


Figure 27. Data Leakage and Unexpected Consequences. Option one (1) stands for “least likely” and five (5) for “most likely”.

This question gathered 51 responses. The option number three gathered the most (37,2%) responses. See Table 13 below for results in numbers.

Table 13. Data Leakage and Unexpected Consequences.

1	2	3	4	5		Average	Median	Total
2	6	19	16	8		3,4	3,0	51

<u>Question 23:</u> Authentication and Secure Communication Failures

The purpose of this question is to find out the estimate likelihood of the risk of authentication and secure communication failures. The response was provided via multiple-choice form where the users estimated the likelihood by selecting one option on the scale of one (1) to five (5). The option one (1) stands for “least likely” and the option five (5) stands for “most likely”. Participant were also provided with the following description:

> *”No-code/low-code applications typically connect to business-critical data via connections set up by business users, which can often result in insecure communication” (”LCNC-SEC-04: Authentication and Secure Communication”, n.d., para. 1).*

The percentual results can be seen in Figure 28 below.

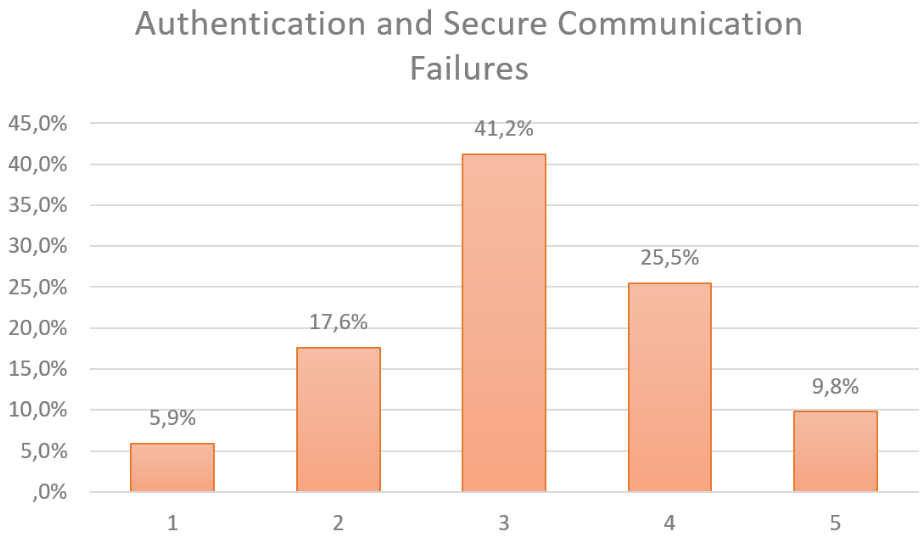


Figure 28. Authentication and Secure Communication Failures. Option one (1) stands for “least likely” and five (5) for “most likely”.

This question gathered 51 responses. Option number three (3) got the most responses (41,2%) among the participants. See Table 14 below for results in numbers.

Table 14. Authentication and Secure Communication Failures.

1	2	3	4	5		Average	Median	Total
3	9	21	13	5		3,2	3,0	51

<u>Question 24:</u> Security Misconfiguration

The purpose of this question is to find out the estimate likelihood of the risk of security misconfiguration. The response was provided via multiple-choice form where the users estimated the likelihood by selecting one option on the scale of one (1) to five (5). The option one (1) stands for "least likely" and the option five (5) stands for "most likely". Participant were also provided with the following description:

> *"Misconfigurations can often result in anonymous access to sensitive data or operations, unprotected public endpoints, secrets, and oversharing" ("LCNC-SEC-05: Security Misconfiguration", n.d., para. 1).*

The percentual results can be seen in Figure 29 below.

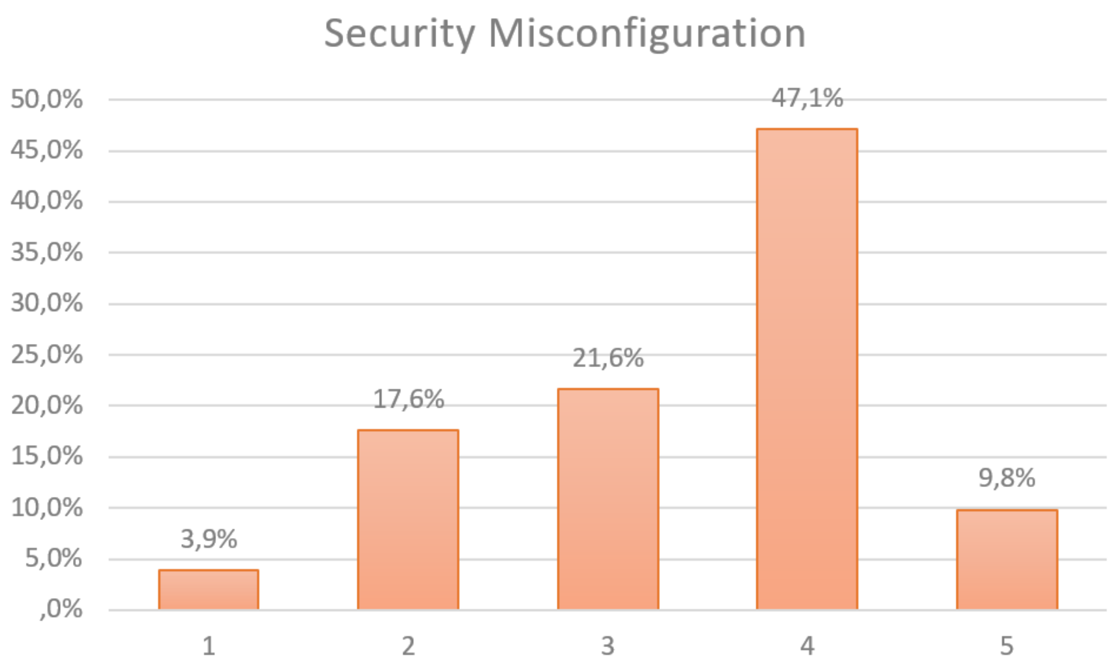


Figure 29. Security Misconfiguration. Option one (1) stands for “least likely” and five (5) for “most likely”.

This question gathered 51 responses. Most of the participants (47,1%) chose the option number four (4). See the Table 15 below for results in numbers.

Table 15. Security Misconfiguration.

1	2	3	4	5		Average	Median	Total
2	9	11	24	5		3,4	4,0	51

<u>Question 25:</u> Injection Handling Failures

The purpose of this question is to find out the estimate likelihood of the risk of injection handling failures. The response was provided via multiple-choice form where the users estimated the likelihood by selecting one option on the scale of one (1) to five (5). The option one (1) stands for “least likely” and the option five (5) stands for “most likely”. Participant were also provided with the following description:

> *”No-code/low-code applications ingest user-provided data in multiple ways, including direct input or retrieving user-provided content from various services. Such data can contain malicious payloads that may introduce risk to the application” (”LCNC-SEC-06: Injection Handling Failures”, n.d., para. 1).*

The percentual results can be seen in Figure 30 below.

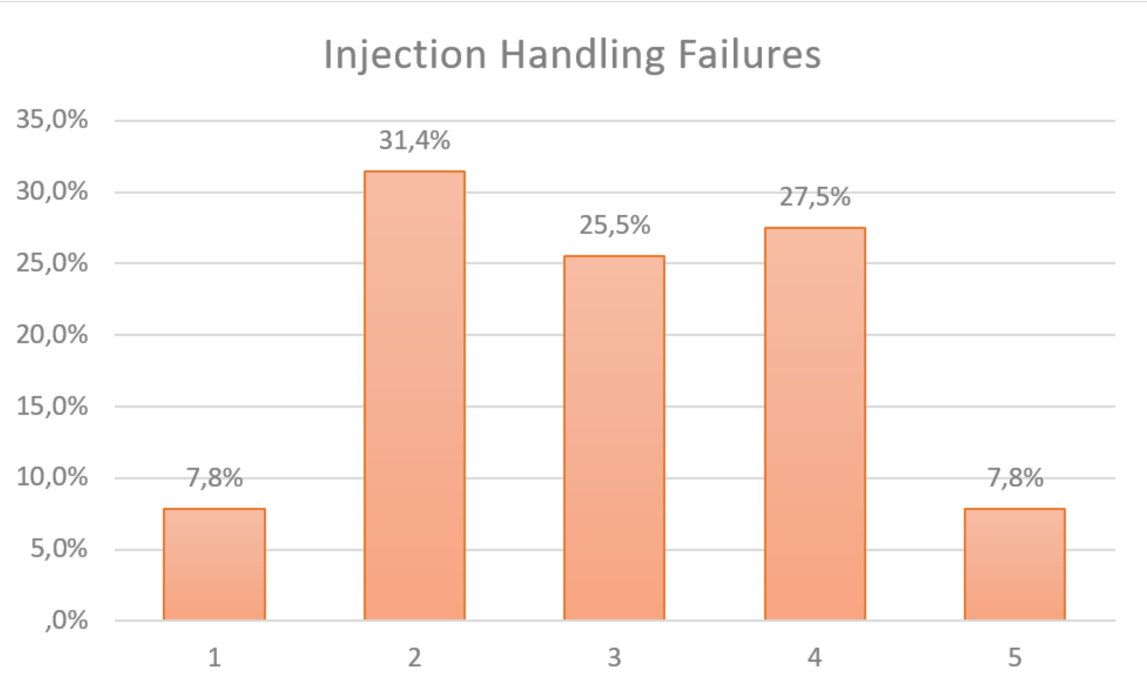


Figure 30. Injection Handling Failures. Option one (1) stands for “least likely” and five (5) for “most likely”.

This question gathered 51 responses. Option number two (2) gained the most responses, although the average and the median are 3,0. See Table 16 below for results in numbers.

Table 16. Injection Handling Failures.

1	2	3	4	5		Average	Median	Total
4	16	13	14	4		3,0	3,0	51

<u>Question 26:</u> Vulnerable and Untrusted Components

The purpose of this question is to find out the estimate likelihood of the risk of vulnerable and untrusted components. The response was provided via multiple-choice form where the users estimated the likelihood by selecting one option on the scale of one (1) to five (5). The option one (1) stands for "least likely" and the option five (5) stands for "most likely". Participant were also provided with the following description:

> *"No-code/low-code applications rely heavily on ready-made components out of the marketplace, the web, or custom connectors built by developers. These components are often unmanaged, lack visibility, and expose applications to supply chain-based risks" ("LCNC-SEC-07: Vulnerable and Untrusted Components", n.d., para. 1).*

The percentual results can be seen in Figure 31 below.

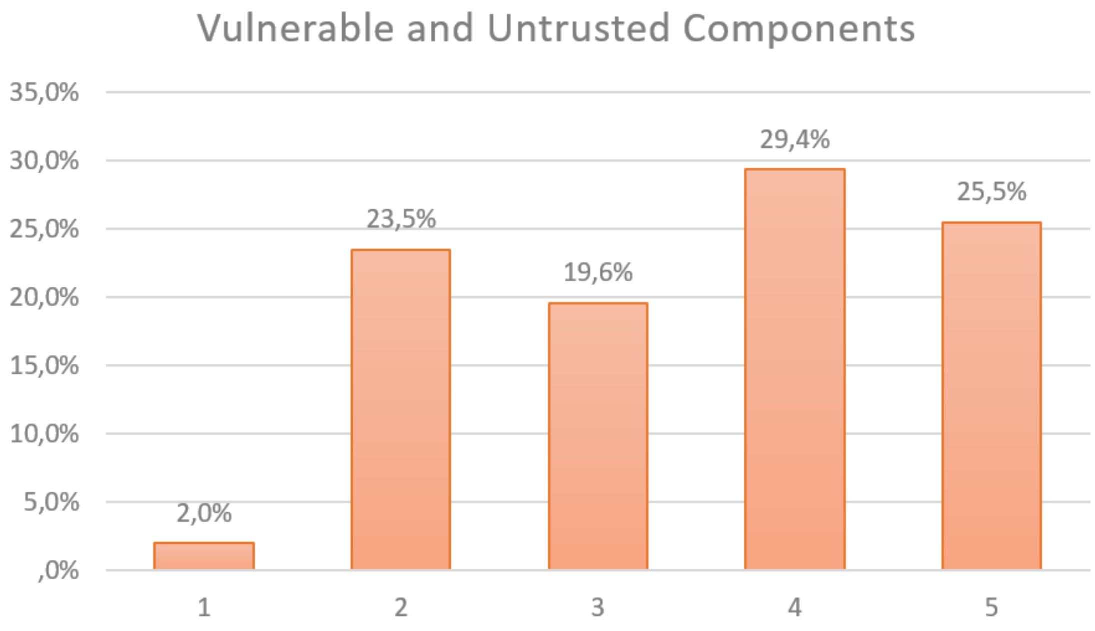


Figure 31. Vulnerable and Untrusted Components. Option one (1) stands for "least likely" and five (5) for "most likely".

This question gathered 51 responses. Option number four (4) gained the most (29,4%) responses. See Table 17 below for results in numbers.

Table 17. Vulnerable and Untrusted Components.

1	2	3	4	5		Average	Median	Total
1	12	10	15	13		3,5	4,0	51

<u>Question 27:</u> Data and Secret Handling Failures

The purpose of this question is to find out the estimated likelihood of the risk of data and secret handling failures. The response was provided via multiple-choice form where the users estimated the likelihood by selecting one option on the scale of one (1) to five (5). The option one (1) stands for “least likely” and the option five (5) stands for “most likely”. Participant were also provided with the following description:

> *”No-code/low-code applications often store data or secrets as part of their "code" or on managed databases offered by the platform, which must be stored adequately in compliance with regulation and security requirements.*
>
> *Furthermore, applications often lack a comprehensive audit trail, preventing change management processes and inquiries. Finding out who introduced a change becomes an intractable challenge” (”LCNC-SEC-08: Data and Secret Handling Failures”, n.d., para. 1).*

The percentual results can be seen in Figure 32 below.

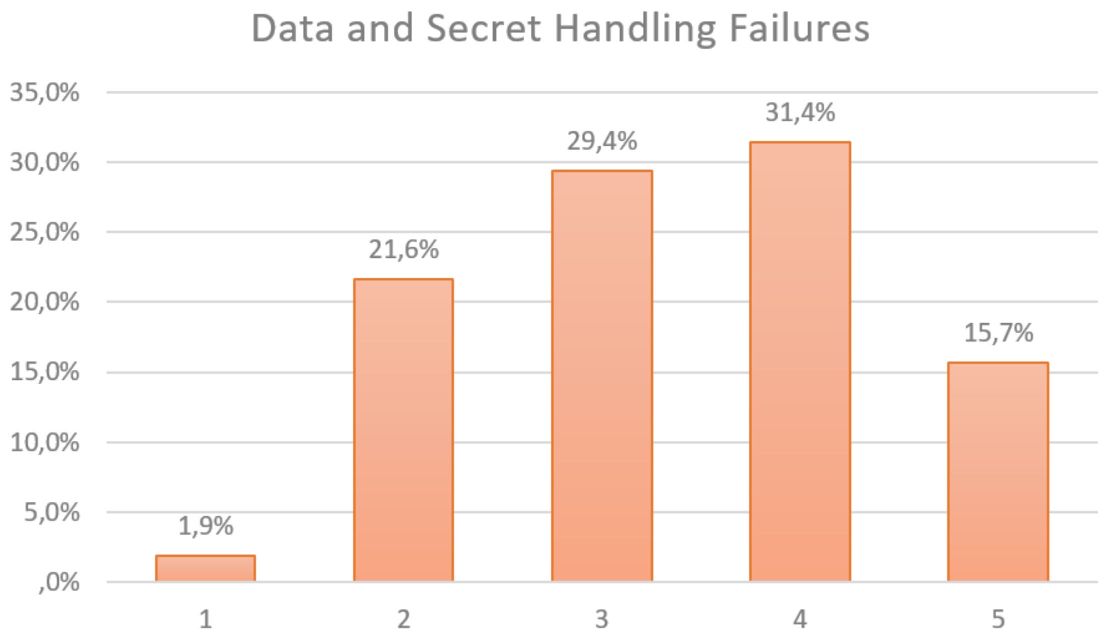


Figure 32. Data and Secret Handling Failures. Option one (1) stands for "least likely" and five (5) for "most likely".

This question gathered 51 responses. The largest (31,4%) group of participants selected the option four (4). See Table 18 below for results in numbers.

Table 18. Data and Secret Handling Failures.

1	2	3	4	5		Average	Median	Total
1	11	15	16	8		3,4	3,0	51

Question 28: Asset Management Failures

The purpose of this question is to find out the estimate likelihood of the risk of asset management failures. The response was provided via multiple-choice form where the users estimated the likelihood by selecting one option on the scale of one (1) to five (5). The option one (1) stands for "least likely" and the option five (5) stands for "most likely". Participant were also provided with the following description:

> *”No-code/low-code applications are easy to create and have relatively low maintenance costs, making them prone to abandonment while remaining active. Furthermore, internal applications can gain popularity rapidly without addressing business continuity concerns” (”LCNC-SEC-09: Asset Management Failures”, n.d., para. 1).*

The percentual results can be seen in Figure 33 below.

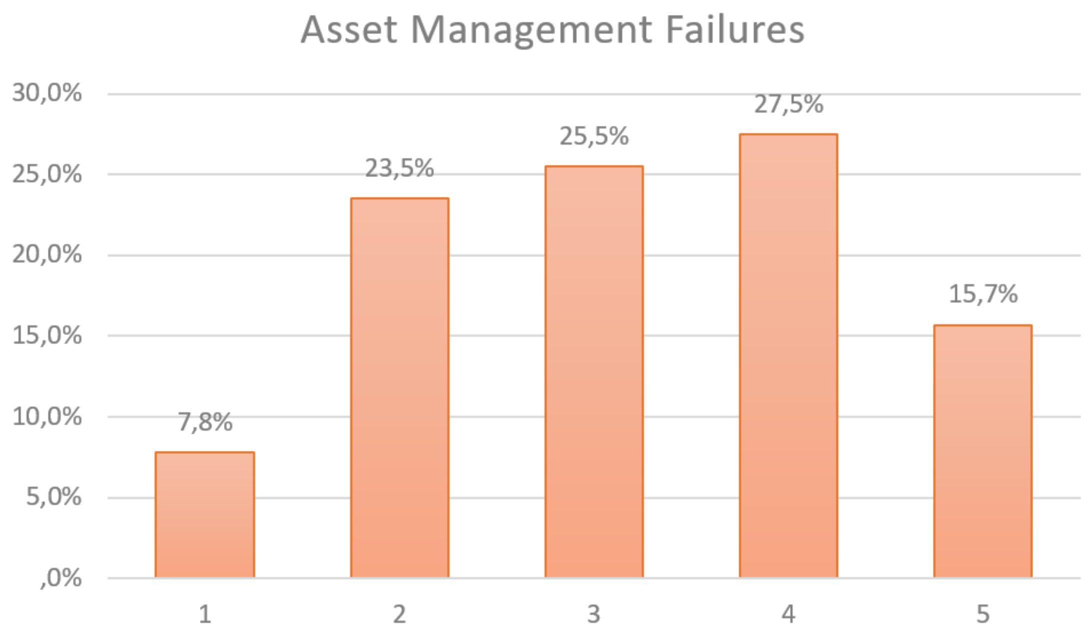


Figure 33. Asset Management Failures. Option one (1) stands for “least likely” and five (5) for “most likely”.

This question gathered 51 responses. The option four (4) was the most (27,5%) chosen option. The median response was 3,0. See Table 19 below for results in numbers.

Table 19. Asset Management Failures.

1	2	3	4	5		Average	Median	Total
4	12	13	14	8		3,2	3,0	51

<u>Question 29:</u> Security Logging and Monitoring Failures

The purpose of this question is to find out the estimate likelihood of the risk of security logging and monitoring failures. The response was provided via multiple-choice form where the users estimated the likelihood by selecting one option on the scale of one (1) to five (5). The option one (1) stands for “least likely” and the option five (5) stands for “most likely”. Participant were also provided with the following description:

> *“No-code/low-code applications often lack a comprehensive audit trail, produce none or insufficient logs, or overshare access to sensitive logs“ (“LCNC-SEC-10: Security Logging and Monitoring Failures”, n.d., para. 1).*

The percentual results can be seen in Figure 34 below.

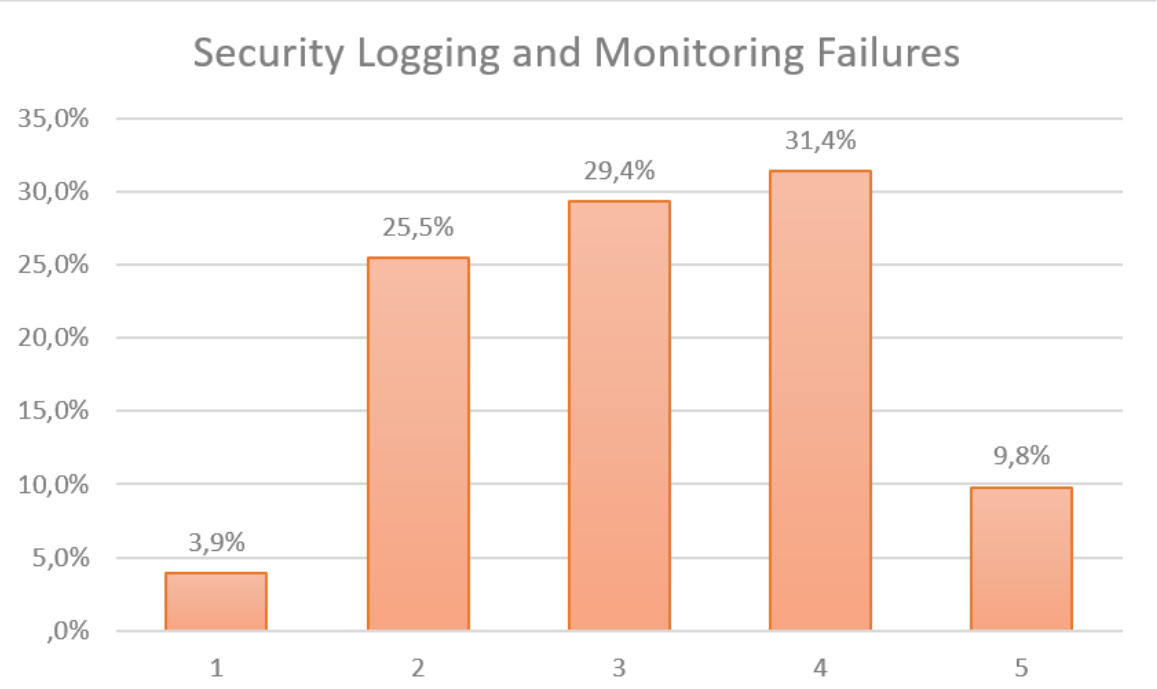


Figure 34. Security Logging and Monitoring Failures. Option one (1) stands for “least likely” and five (5) for “most likely”.

This question gathered 51 responses. The largest group (31,4%) of participants chose option four (4). See Table 20 below for results in numbers.

Table 20. Security Logging and Monitoring Failures.

1	2	3	4	5		Average	Median	Total
2	13	15	16	5		3,2	3,0	51

Question 30: Do you have any comments concerning the list of risks from OWASP in previous page? Or do you suspect any other possible risks concerning low-code/no-code technologies?

The purpose of this question was to gather details on the previous topics or any other possible cyber or information security implications not listed in this section. This question gathered 21 responses. The response was provided via an open textbox and the results can be read from Appendix 4.

4.1.4 Survey analysis

The analysis of the data is based on comparison the results of certain groups based on the responses. It is possible to form a group like “companies using low-code”, “companies not using low-code”, “has no experience on cyber security” or “large sized company” from the data, for example. The choice of the groups to be compared was made based on the topic of survey question and the significance in terms of analysis, keeping in mind the research objectives. The chosen groups, their descriptions and the rules can be seen in Table 21 below.

Table 21. Survey analysis groups

Group name	Description	Ruling
Companies using LCNC	Responses from participants who stated their company utilizing LCNC technologies	Option ‘yes’ selected on ‘Question 11: Is your company already utilizing low-code/no-code software development?’
Companies not using LCNC	Responses from participants who stated their company to not utilize LCNC technologies	Option ‘no’ selected on ‘Question 11: Is your company already utilizing low-code/no-code software development?’
Software department or company	Responses from participants who stated their department or company being focused on software development. In some of the questions the focus is only on the company.	Option ‘software department / company’ selected on ‘Question 4: Which (of these) describes your department or company the best?’
Cyber security department or company	Responses from participants who stated their department or company being focused on cyber security. In some of the questions the focus is only on the company.	Option ‘cyber security department / company’ selected on ‘Question 4: Which (of these) describes your department or company the best?’
Companies based on the size: ‘0-10’, ‘10-50’, ‘50-250’, ‘250-500’ and ‘more than 500’	Responses from participants who stated their company’s size. In some questions these options can be	Options ‘0-10’, ’10-50’, ’50-250’, ‘250-500’ or ‘more than 500’ selected on ‘Question 6:

	grouped by size based on the European Commission’s definition on small to mid-sized enterprises.	How many people does your company employ?’

Questions 5, 13, 16, 17, and 30 were asked via open textbox forms and so the nature of gathered data was qualitative. Therefore, a simplified qualitative content analysis was performed for some of these questions.

In this analysis meaningful observations based on the data and research objectives are highlighted. Some of the questions did not provide observations with significance in terms of research objectives, and therefore were left out of the analysis.

Question 6: How many people does your company employ?

It was possible to make an observation on the difference in company sizes between the companies who used LCNC technologies and those that did not. It seems that larger companies have greater adoption of LCNC technologies than small to mid-sized companies based on the data. The proportional difference can be seen in Figure 35 below.

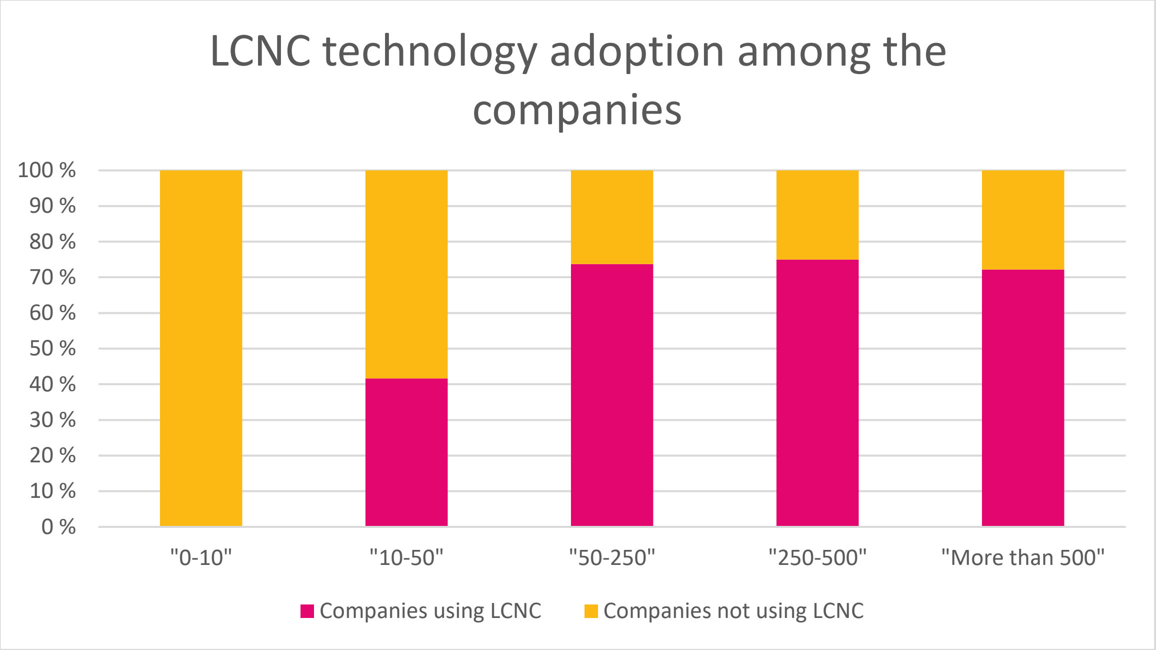


Figure 35. LCNC technology adoption compared to the company sizes.

In the size group of 0-10 there were no LCNC users. The companies sized from 10 to 50 there were five (5) LCNC users compared to seven (7) non-users. In the next size group, more than half of the participants stated their companies using LCNC technologies, and so did all the other larger companies after it.

Question 7: How familiar are you with software development in general?

In this question the respondents estimated their familiarity on a scale from one (1) to five (5), one being ‘not familiar at all’ and five ‘very familiar’.

When the familiarity with software development was compared between the companies using LCNC and those that did not use, it was possible to perceive a difference in the results. Based on the data the participants working in companies which are not using LCNC technologies estimated their familiarity in software development somewhat higher than those in companies utilizing LCNC. The difference can be perceived in Figure 36 below.

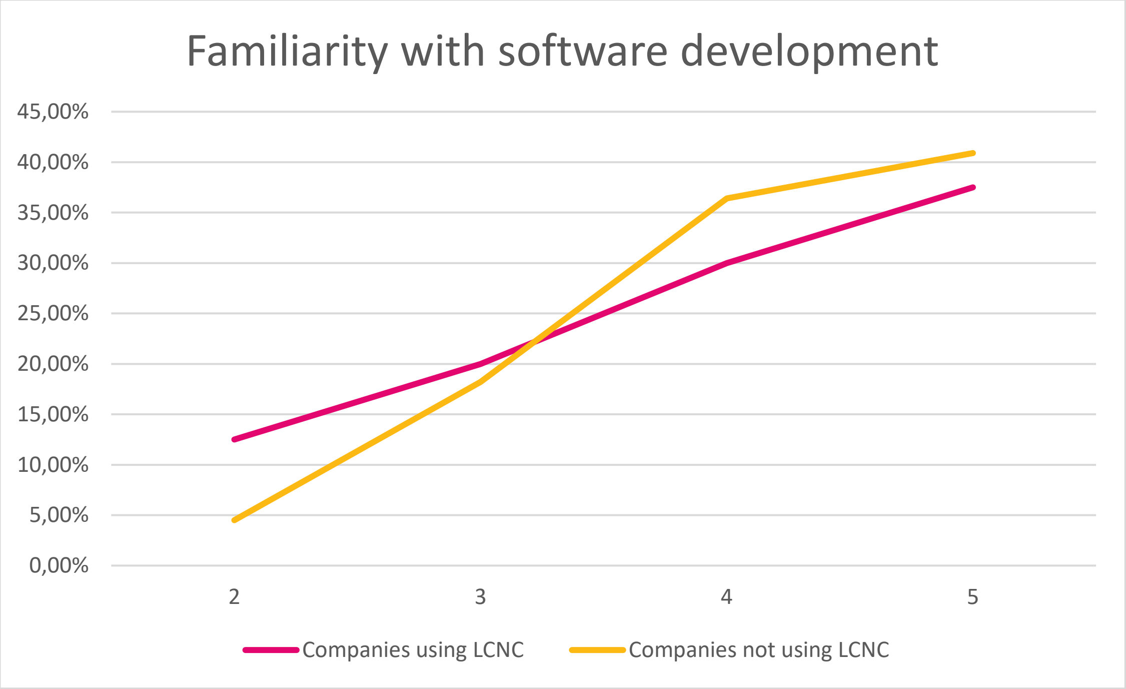


Figure 36. Familiarity on software development between companies using and not using LCNC technologies.

Question 10: How familiar are you with low-code/no-code software development paradigm or technologies?

In this question the respondents estimated their familiarity with LCNC on a scale from one (1) to five (5), one being ‘not familiar at all’ and five ‘very familiar’.

The results of companies using LCNC, companies not using LCNC, cyber security department/companies and software department/companies were compared. The results can be seen in Figure 37 below.

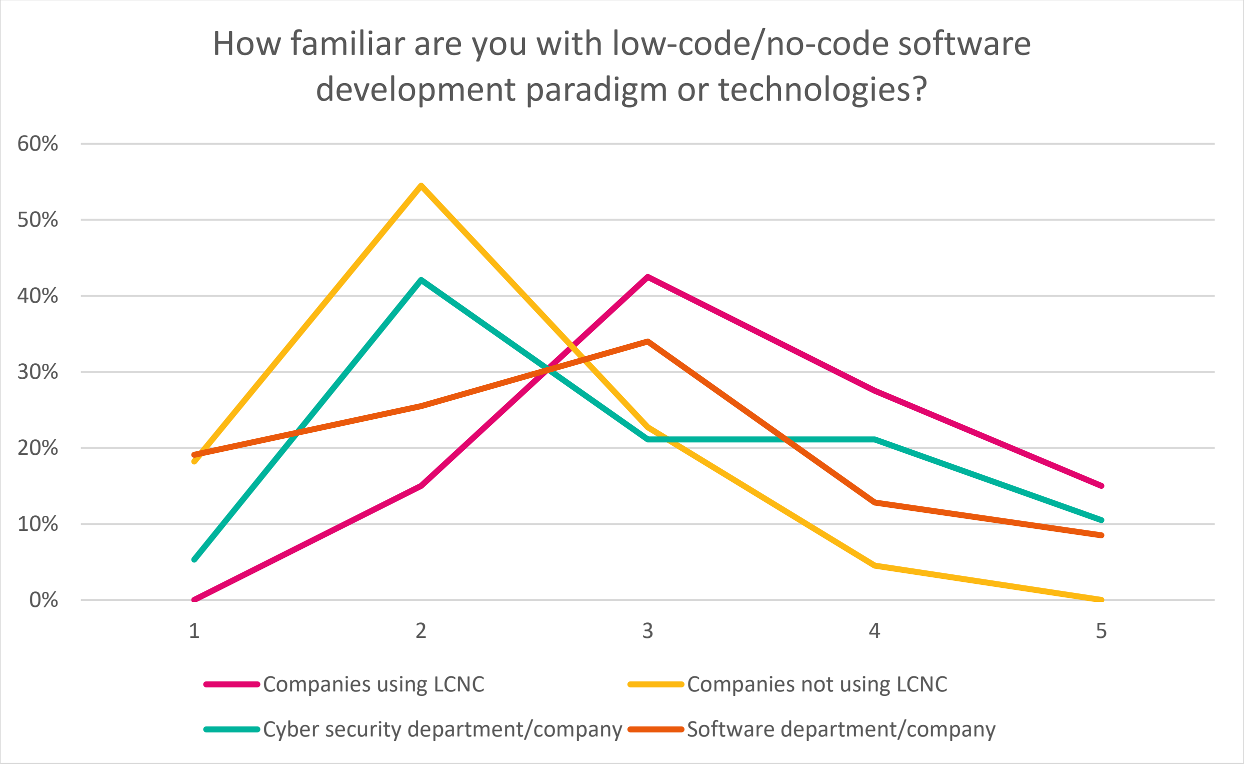


Figure 37. Familiarity with LCNC technologies among the companies

There are a couple of observations that can be made based on the data. First, participants from software departments or companies estimate themselves more familiar with the technology than participants from cyber security departments or companies. The participants who are most familiar with these technologies are the participants whose companies are utilizing it, and on the other hand, participants from companies not using LCNC technologies are the least familiar with these technologies.

Question 11: Is your company already utilizing low-code/no-code software development?

For this question participants stated if their company is already using LCNC technologies, or if they don't know if it is. When the groups of cyber security and software department/companies were compared the adoption of LCNC technologies in cyber security departments/companies can be observed utilizing LCNC relatively more. The proportional results are presented in Figure 38 below.

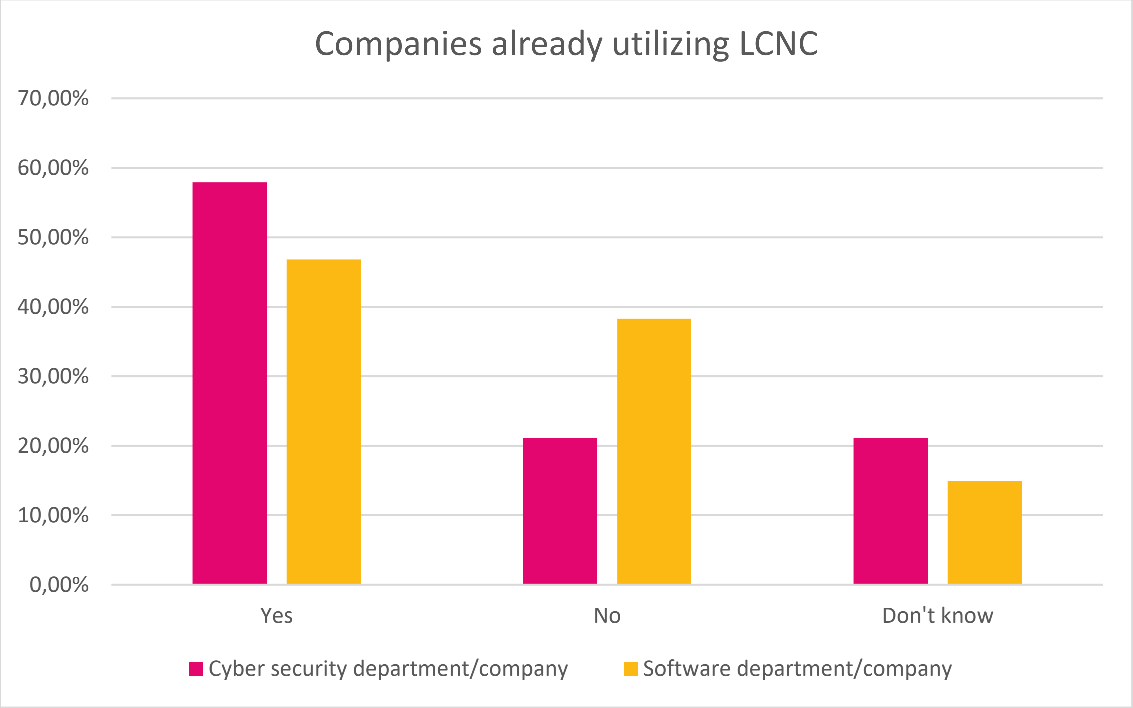


Figure 38. Companies utilizing LCNC.

Question 12: When would you roughly estimate your company start utilizing low-code/no-code technology?

In this question the participants whose companies did not yet use LCNC technologies estimated when their company would adopt these technologies. Most estimated the adoption to happen in one to five years, although participants from software departments/companies had the most uncertainty in their responses, which can be seen in Figure 39 below. The reasoning behind the uncertainty is analyzed in the next question.

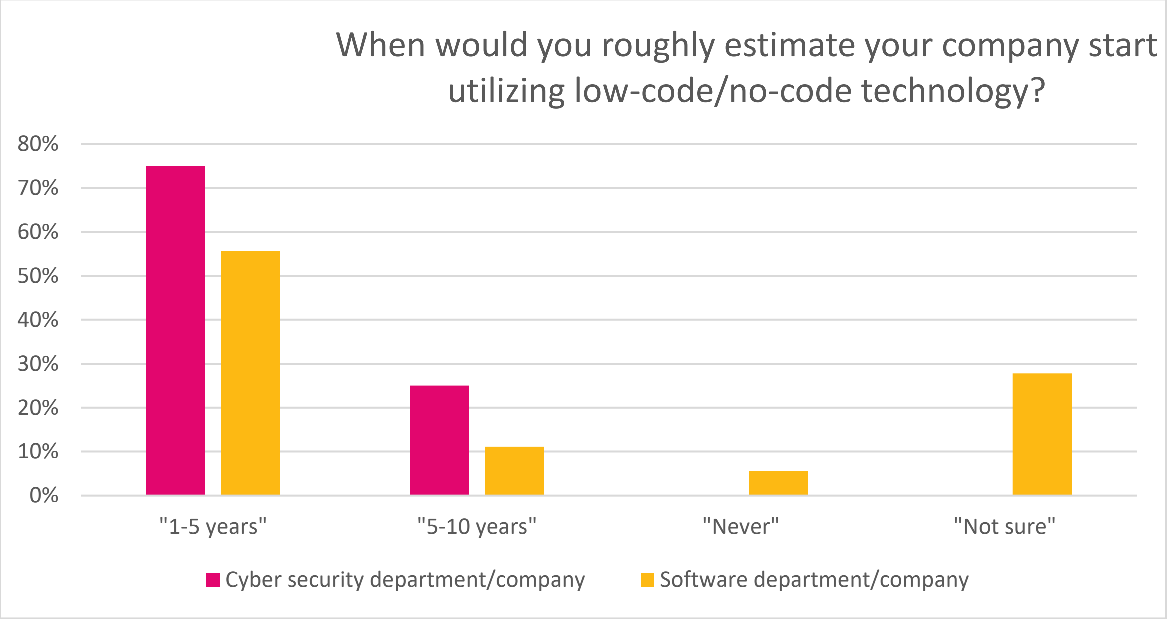


Figure 39. Comparison of estimated adoption of LCNC technologies.

Question 13: If you answered 'never' or 'not sure' in the previous question, would you like to explain why? The participants who selected the ‘never’ or ‘not sure’ option in the previous question, stating they do not know when or if they would adopt LCNC technologies in their companies, described their reasoning by responding to this question. Out of the six (6) responses an occurring theme could be recognized: four responses stated the technology is not suitable for their needs. For example, a lack of flexibility, possibility to create complex applications and the lack of serious evaluation were stated.

> *“I've never heard of low-code/no-code platforms where you could do serious programming, delivering complex applications for customers. My initial feeling is that these platforms would be suitable for more light weight development, but my initial judgement may very well be wrong.” Anonymous participant.*

Question 14: How well does low-code/no-code technology serve your company's needs? Here the participants estimated how well the low-code/no-code technologies serve their company’s needs on a scale from one (1) to five (5). The option one (1) stands for “not at all” and option five (5) for “very well”.

First the groups of cyber security and software companies were compared. From the comparison it is possible to observe software companies having more even spread among the responses, whereas cyber security companies have strong emphasis on the option ‘4’. This can be perceived in Figure 40 below.

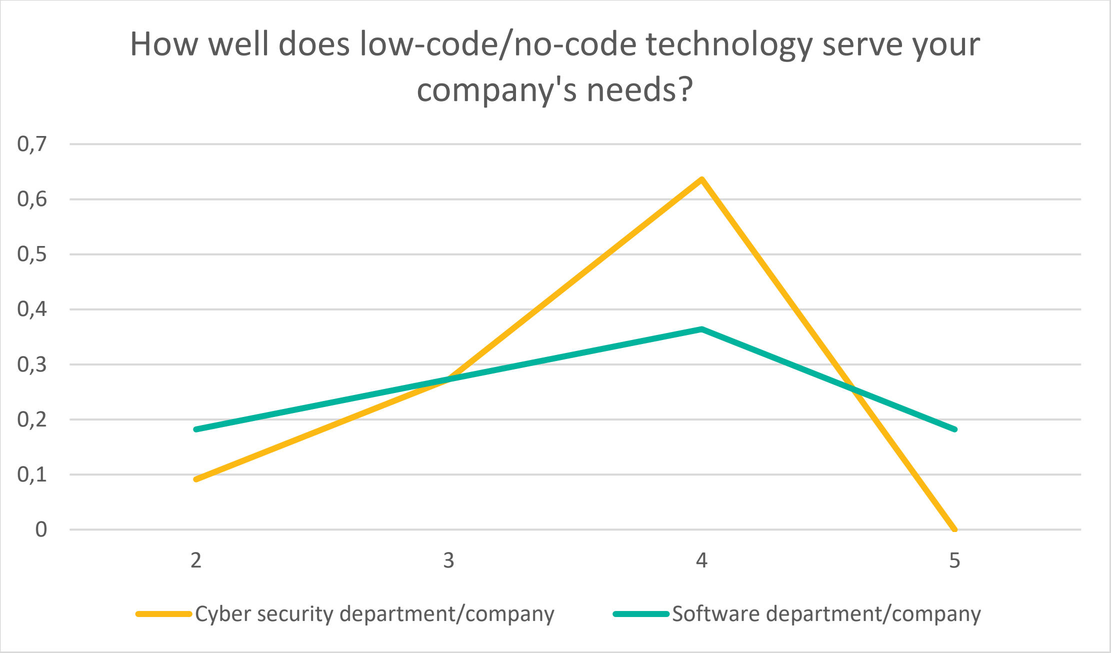


Figure 40. Comparison of cyber security and software companies on the question how well does LCNC technologies server their needs.

When the groups based on the company sizes were compared it was possible to perceive the group of company size from 50 to 250 employees being the most satisfied with the LCNC technologies. The comparison between the groups can be seen in Figure 41 below.

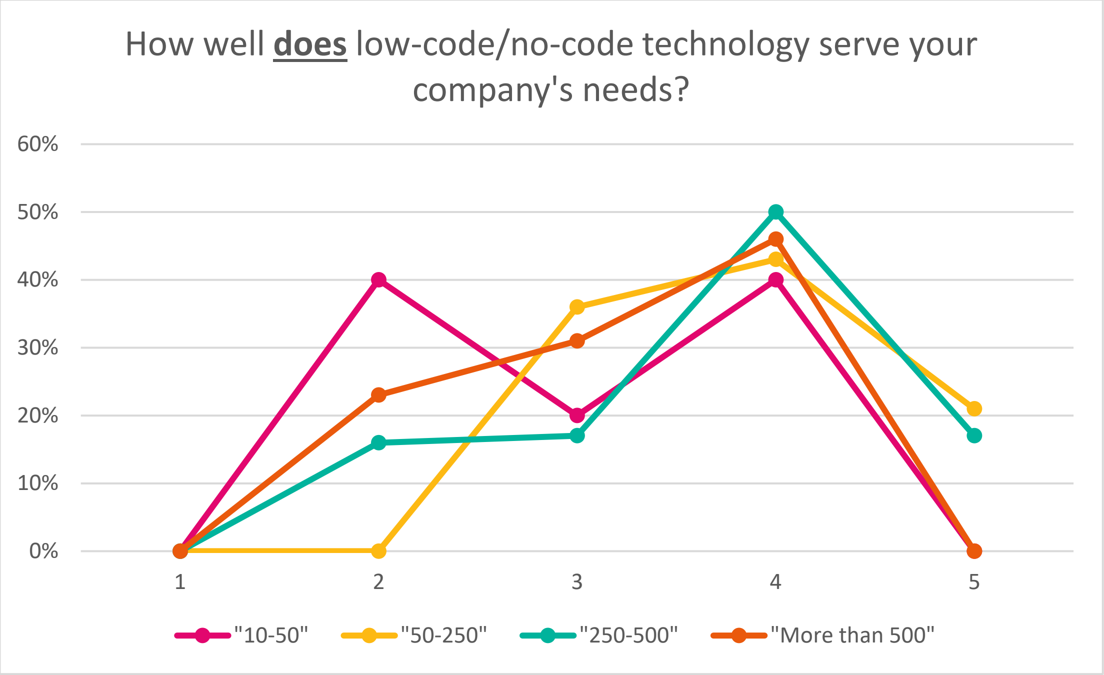


Figure 41. Company size comparison on how well the LCNC technology server the company's needs

If the average from the small to mid-sized enterprises (less than 250) were compared against average from large enterprises (more than 250) the differences were minor, as can be seen on the Figure 42 below.

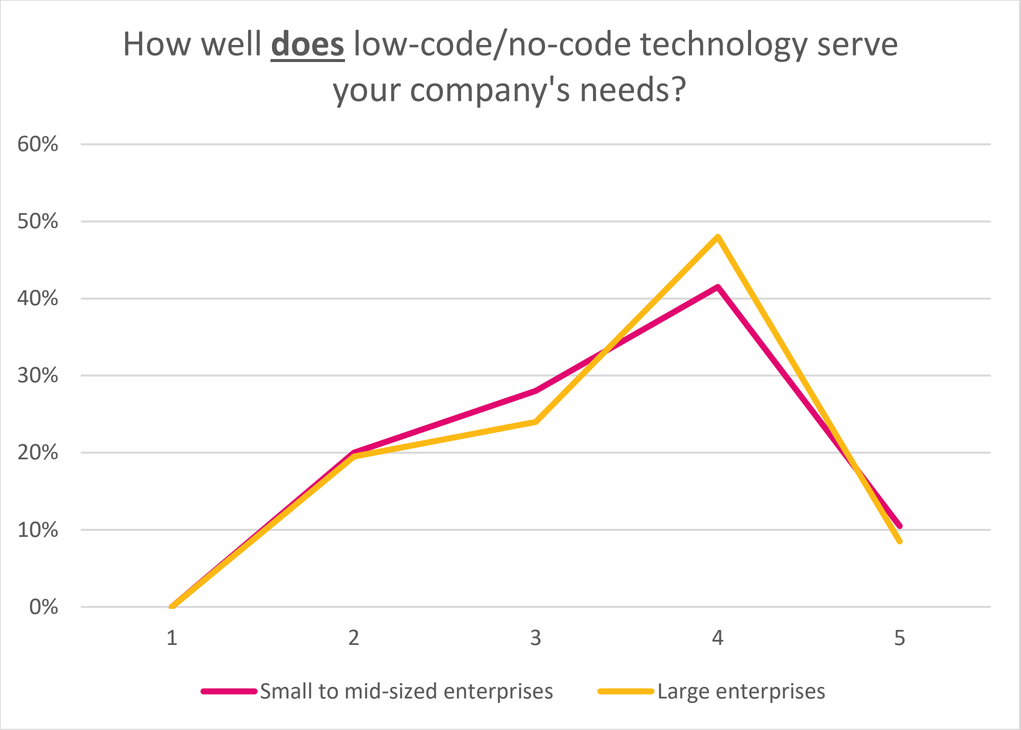


Figure 42. Differences between small to mid-sized enterprises against large enterprises.

Question 15: How well would low-code/no-code technologies serve your company's needs? For this question groups of cyber security or software companies, and the groups based on company sizes were compared. No trends or meaningful observations could be recognized from this data. The expectations of how these companies could utilize these technologies are analyzed in the next question.

Question 16: Please describe how does your company utilize low-code technologies. Within the responses to this question three themes transpired: application development, automation and integrations. Out of 26 responses 18 responses were related to one or two of these themes.

Application development was mentioned in 12 responses. Enterprise systems, mobile applications and data store apps, for example, were discussed.

> *”Graphics team utilizes low-code on a daily basis to achieve their results. Low-code has been quite paramount in bridging the gap between software developers and artists.” Anonymous participant.*

Six (6) responses were related to the theme of *integrations*. For example, data integrations and integrating internal software were described.

> *”My company is a software consultancy and in some of tour customer projects our employees develop business applications for our customers using MS PowerApps on top of cloud data stores and train the customer end-users to create their own applications. This usually extends our on-prem to cloud data integration offering in situations where the customers lack their own capabilities to utilize their data stores to produce business value.” Anonymous Participant*

There were also six (6) related to the theme of *automation*. Participant described using LCNC to robot process automation for business and finance processes, security automation pipelines and static content creation.

> *”My company utilizes low-code technologies in automation and generating static content from dynamic sources.” Anonymous participant.*

Among the respondents LCNC technologies were also used for creating websites and prototyping. The rest of the responses were hard to connect to any theme.

Question 17: Please describe how could your company utilize low-code technologies. This question was asked if the respondent’s company was not yet using it, but they gave some kind of estimation when they would possibly start utilizing LCNC technologies. Participants mainly described the possible reason for use as application development in general, although there were a couple of themes worth mentioning. Among of those respondents who described their expectations of possible adoption of LCNC technologies in greater detail hopes of speeding up the application development, allowing more customization and enabling non-technical people to co-create applications were discussed.

> *”Simple or repetitive apps could be developed quicker and devs could focus on more complex tasks” Anonymous participant.*

> *” One possibility would be involving and empowering non-technical employees to prototype the kinds of internal tools that they need in their daily duties. Currently, developing these tools involves a lot of guesswork and reiteration, because non-technical employees don't feel confident in communicating their technical requirements and technical employees don't have a clear understanding of user scenarios.” Anonymous participant.*

Question 18: Do you have any knowledge on cyber security?

In this question the participant was asked if they have any knowledge on cyber security. Based on the responses it is possible to notice the people from companies utilizing LCNC technologies to have more initial knowledge on cyber security than those whose company does not yet utilize these technologies. The difference can be observed in Figure 43 below.

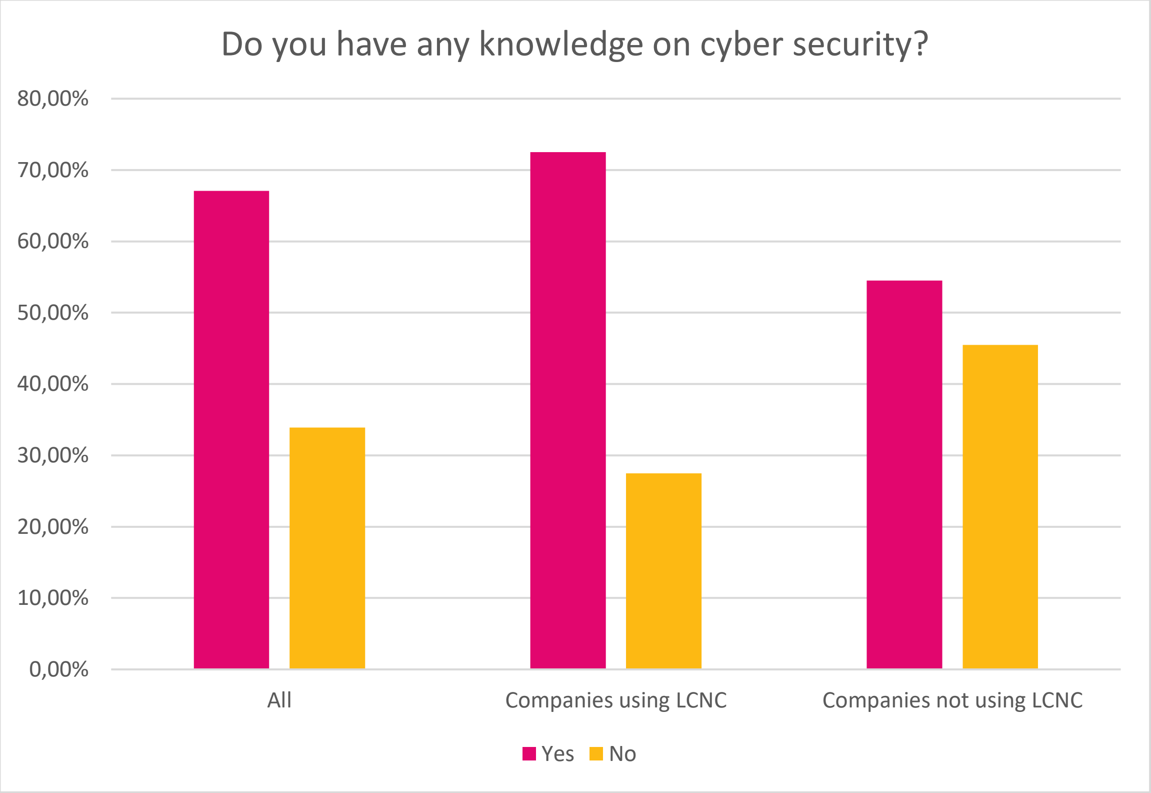


Figure 43. Comparison of groups based on knowledge on cyber security.

The survey was now over for those who had no experience on cyber security and those who did have continued to the cyber security section of the survey. It is relevant to note that the sample size decreases from here. 51 participants continued the survey where 25 were finished.

Question 19: How experienced are you in cyber security?

In this question the participants stated their experience on cyber security in years. Related to the findings in previous question, it is possible to observe companies using LCNC technologies employing people with longer experience on cyber security. See Figure 44 below for the comparison.

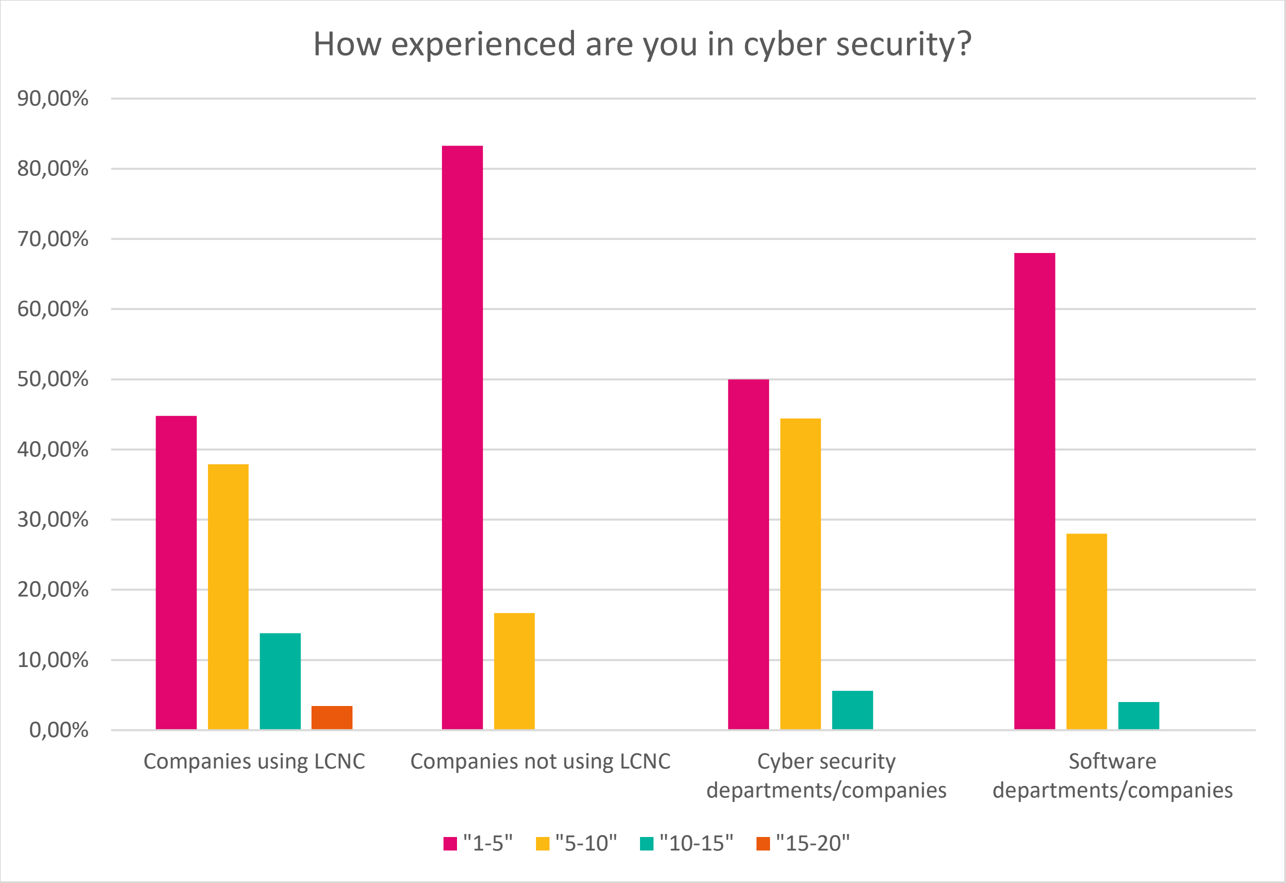


Figure 44. Comparison of groups in the years of experience on cyber security.

Question 20: Account Impersonation.

In this question the participants were asked to estimate the likelihood of potential risk of account impersonation. Option one (1) being the least and option five (5) being the most likely.

Although the differences are small it can be noticed that companies who do not use LCNC technologies estimated the likelihood the most probable in comparison. This can be seen in Figure 45 below.

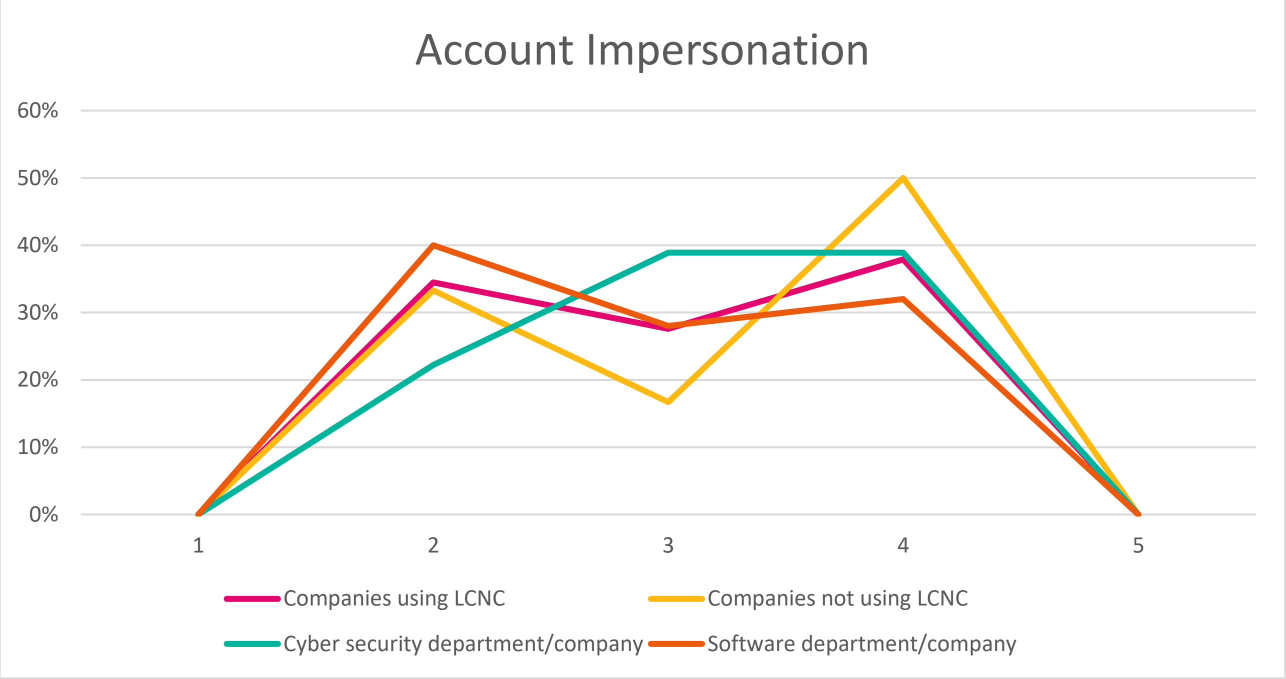


Figure 45. Account impersonation comparison.

When we compare the company sizes, we can also notice that larger companies tend to estimate the risk bigger than smaller ones. Group of “0-10” only got one response of option four (4) and was left out from the graph. See the comparison in Figure 46 below.

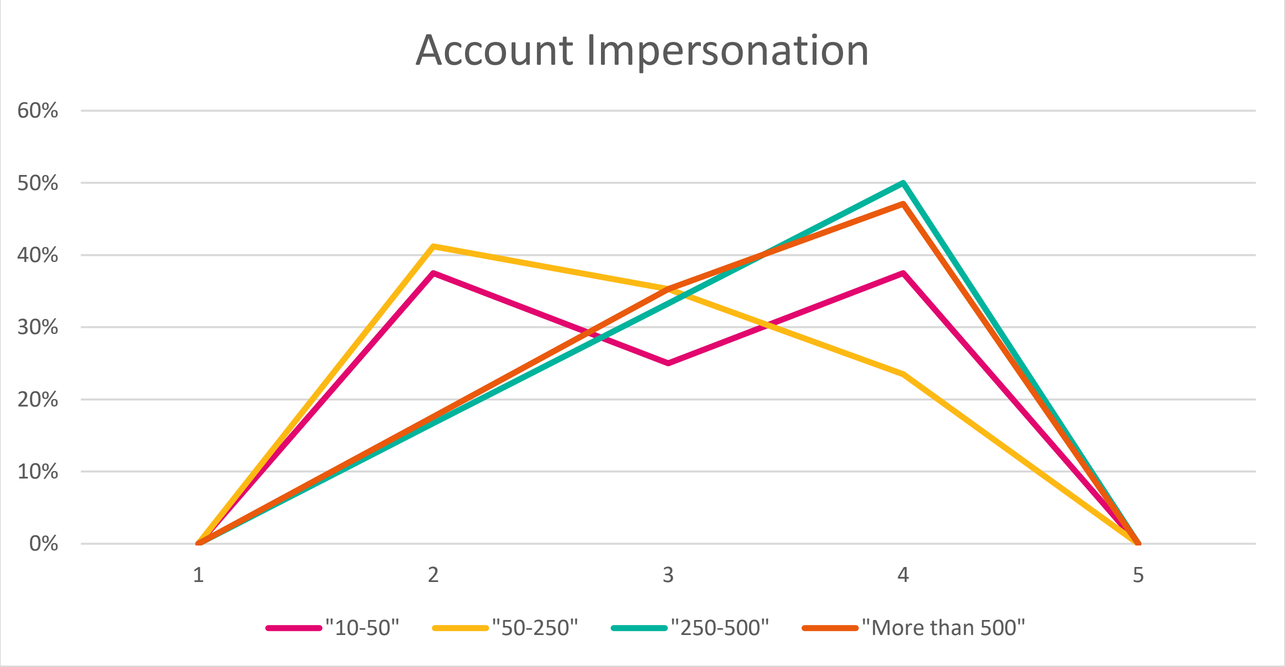


Figure 46. Account impersonation comparison based on company sizes.

Question 21: Authorization Misuse.

In this question the participants were asked to estimate the likelihood of potential risk of authorization misuse. Option one (1) being the least and option five (5) being the most likely.

Initially groups of “companies using LCNC”, “companies not using LCNC”, “cyber security department/company” and the “software department/company” were compared, but no significant differences could be found in the comparison.

When the groups based on company sizes were compared it was observed that the companies sized 10 to 50 estimated the likelihood lower than the other groups. Group of “0-10” had one response of option four (4) and was left out from the graph. See the results in Figure 47 below.

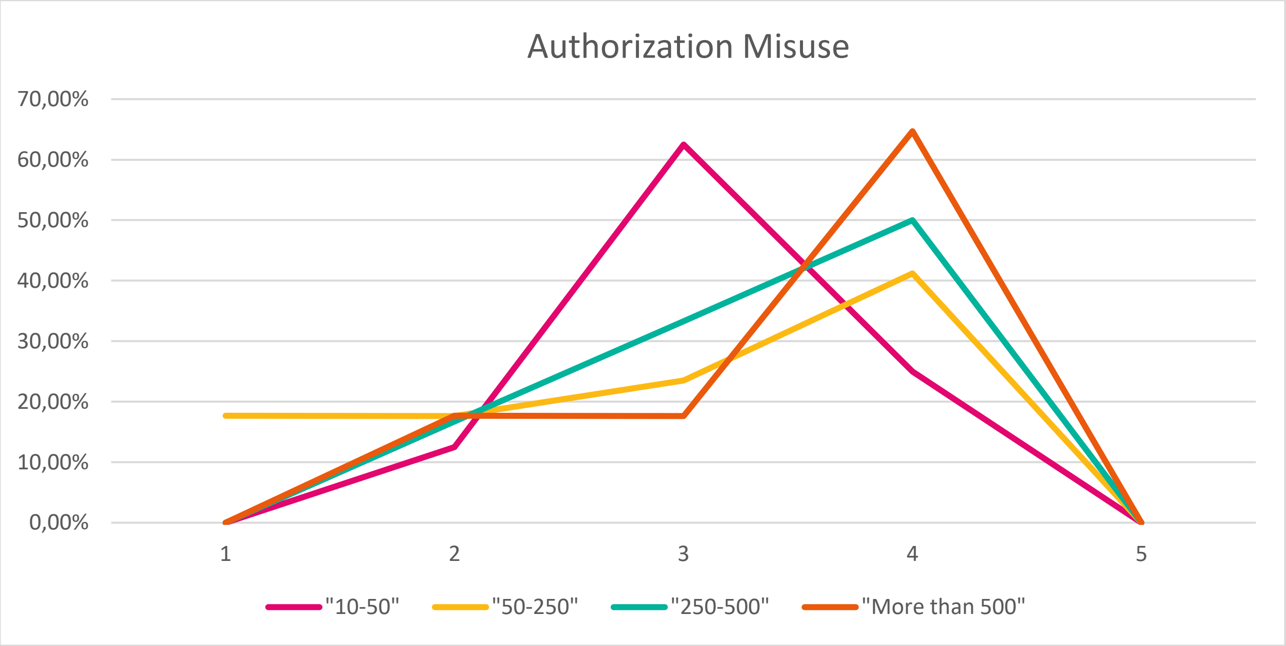


Figure 47. Authorization misuse comparison between company sizes.

Question 22: Data Leakage and Unexpected Consequences.

In this question the participants were asked to estimate the likelihood of potential risk of data leakage and unexpected consequences. Option one (1) being the least and option five (5) being the most likely.

Like in question 20 about account impersonation, here the companies not using LCNC estimated the likelihood the highest in comparison. This can be seen in Figure 48 below.

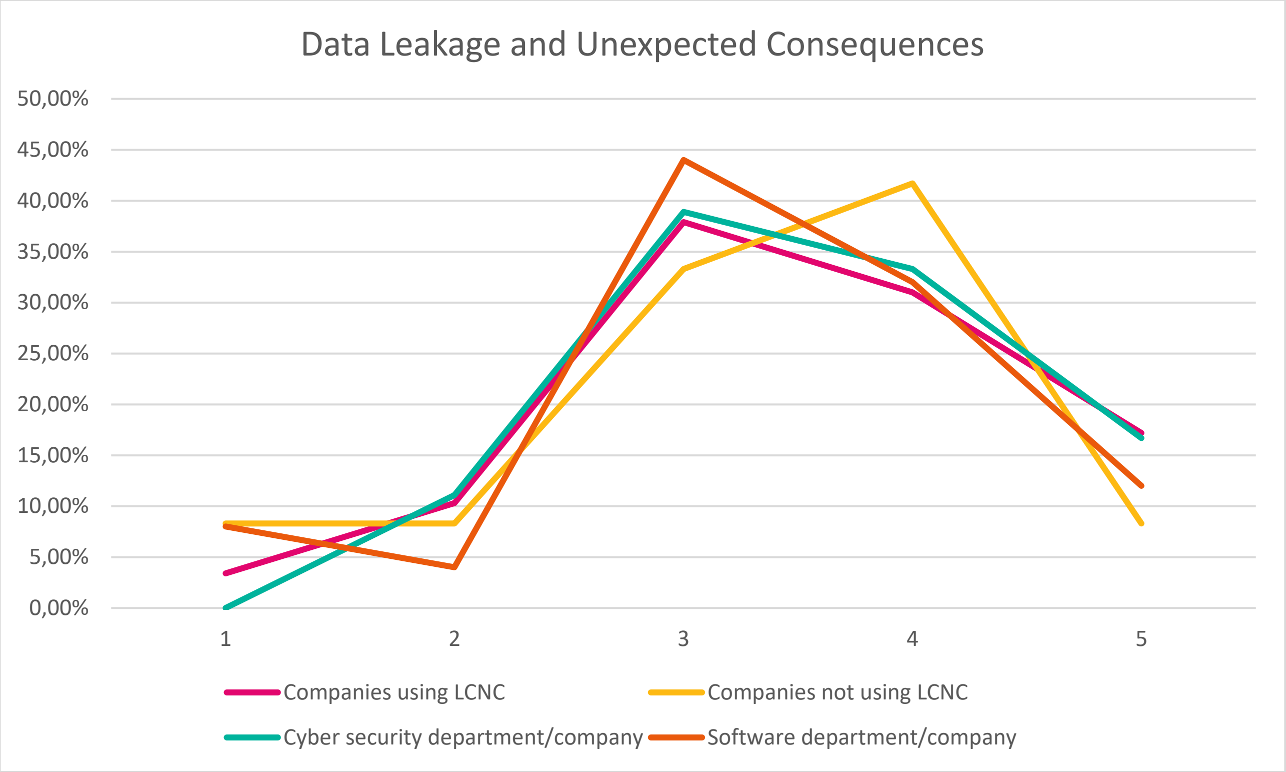


Figure 48. Data leakage and unexpected consequences comparison.

Groups based on company sizes were also compared, but no significant differences could be found in the comparison.

Question 23: Authentication and Secure Communication Failures

While comparing the groups and company sizes no significant trends or differences could be observed under this question.

Question 24: Security Misconfiguration.

In this question the participants were asked to estimate the likelihood of potential risk of security misconfiguration. Option one (1) being the least and option five (5) being the most likely.

Participants working in cyber security department or companies can be observed estimating this risk the highest in comparison. See Figure 49 below for the differences.

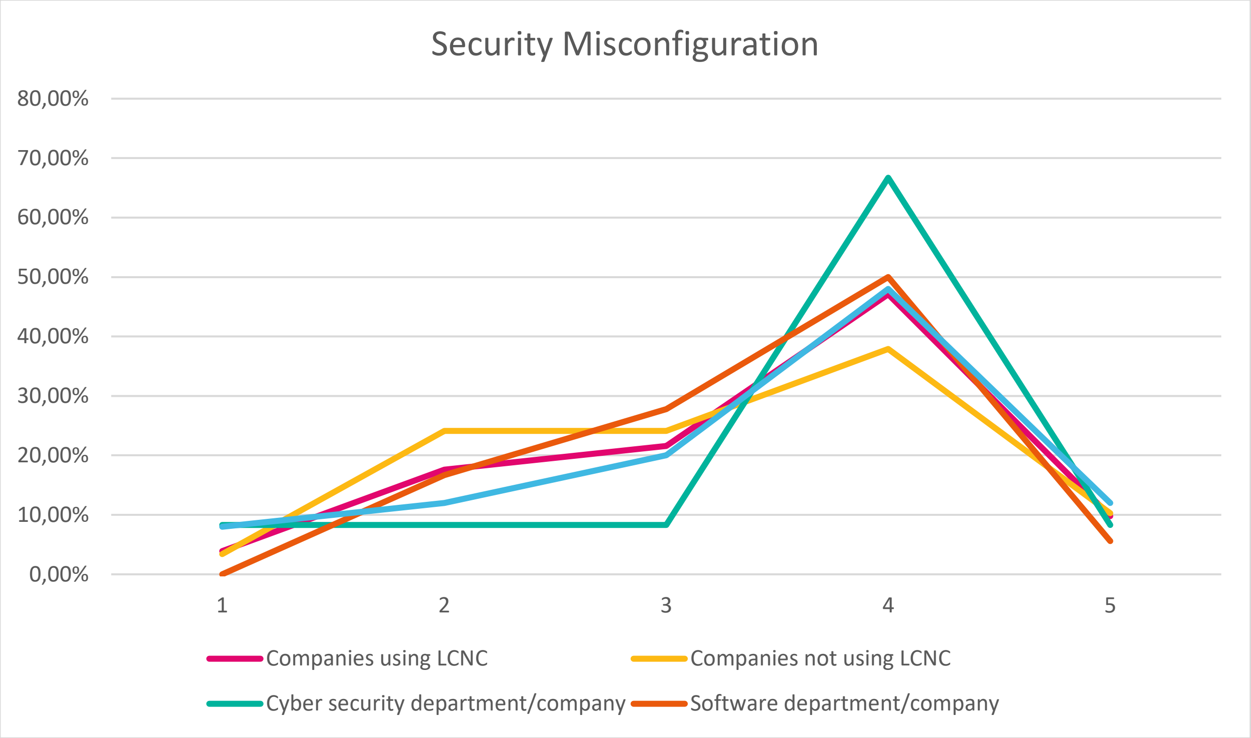


Figure 49. Security misconfiguration comparison.

Question 25: Injection Handling Failures and the **Question 26: Vulnerable and Untrusted Components.** While comparing the groups and company sizes no significant trends or differences could be observed under these questions.

Question 27: Data and Secret Handling Failures.

In this question the participants were asked to estimate the likelihood of potential risk of data and secret handling failures. Option one (1) being the least and option five (5) being the most likely.

Based on the data participants from cyber security departments or companies rated the likelihood of this risk with a bit higher emphasis on the likely end of the scale, in comparison. See Figure 50 below for the comparison.

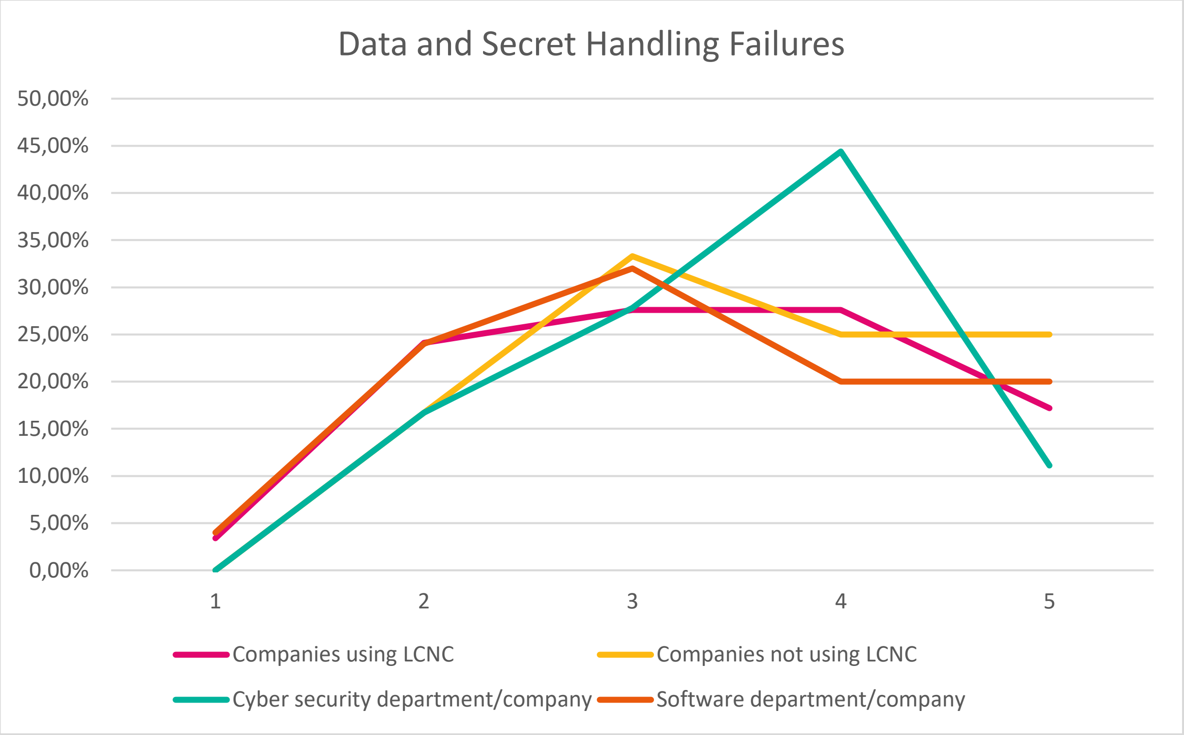


Figure 50. Comparison on data and secret handling failures risk.

Question 28: Asset Management Failures.

In this question the participants were asked to estimate the likelihood of potential risk of asset management failures. Option one (1) being the least and option five (5) being the most likely.

As in the previous question an observation can be made from the estimation of the likelihood of an asset management failure. Participants working in cyber security department or company estimated the likelihood to be relatively more likely than other groups. See the difference in Figure 51 below.

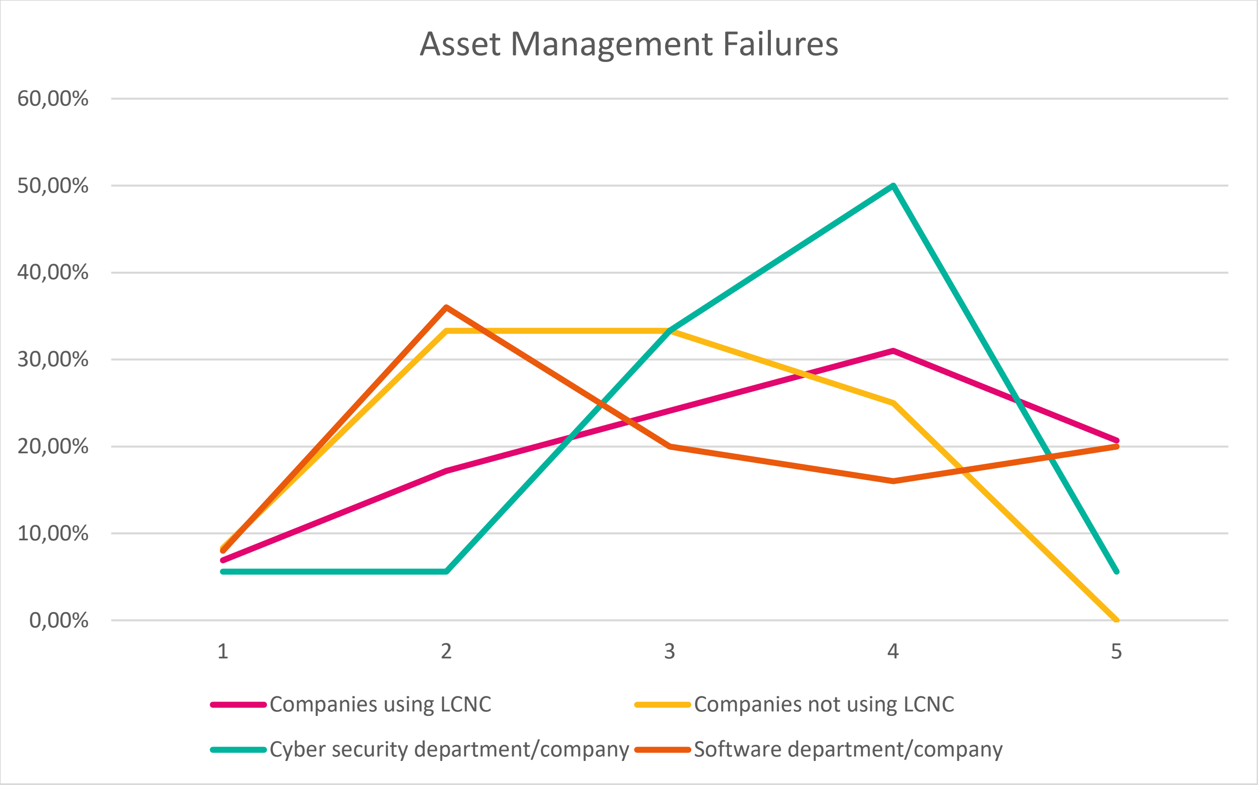


Figure 51. Comparison on asset management failures.

Question 29: Security Logging and Monitoring Failures.

In this question the participants were asked to estimate the likelihood of potential risk of security logging and monitoring failures. Option one (1) being the least and option five (5) being the most likely.

And like in the two previous questions, a minor emphasis can be noted in the group of cyber security department/company in this comparison as well. See Figure 52 below for the results.

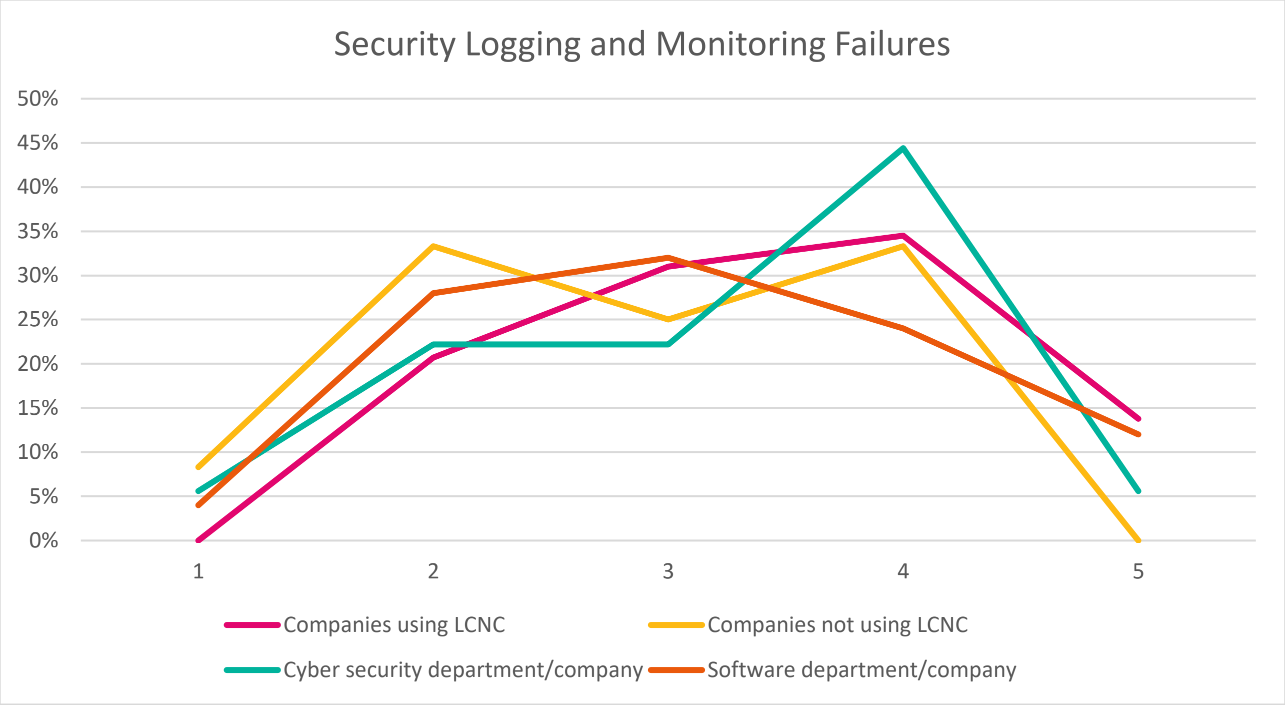


Figure 52. Comparison on security logging and monitoring failures.

Question 30: Do you have any comments concerning the list of risks from OWASP in previous page? Or do you suspect any other possible risks concerning low-code/no-code technologies?
This question about cyber security risks was the most open-ended question of the survey. Responses represented a great variety in the concerns. From the responses two themes could be recognized: governing and lack of transparency.

Out of the 15 responses five (5) are related to governing the developers. Participants discussed how the so-called ”citizen developers” should always be governed by the more senior software developers and how their powers and credentials should be limited.

> *”From my perspective, low-code/no-code staff should always have a senior "hard code" staff member supervising their work in order to prevent technical mistakes and control the overall technical design. Low-code staff has been rarely qualified to analyze technical approaches in design, compared to junior programmers who might be more familiar with the hard technical concepts behind the technologies.” Anonymous participant.*

At least three (3) responses fell under the theme of lack of transparency. Selecting the chosen service provider based on their security scrutinies and the nature of outsourced platforms were discussed.

> *”Complexity and opaqueness in troubleshooting situations. Hard time to get response when platform is outsourced to another company. If I have an incident with their code the chances are all their customers will which will result in a huge risk in case of disaster.” Anonymous participant.*

The remaining responses which could not easily be categorized included discussion on lack of control, usual risks related to cloud-hosted services and the lowered commitment levels. Even some benefits of cyber security were recognized.

> *”This might solve itself in the future if the approach matures, but it might also cause a shift to more predatory development standards when the commitment level drops lower and mitigates the long term risks.” Anonymous participant.*
>
> *”On the other hand no code can be very declarative, which could enable it to know fairly precisely what rights it actually needs and why. This could allow very precisely crafted access rights; a good thing.” Anonymous participant.*

4.2 Semi-structured theme interviews

To support the survey study, a set of five (5) semi-structured interviews were also conducted in February and March of 2023. Participants were found using a LinkedIn post and direct messages on the said platform. All the participants were familiar with software development and used low-code/no-code (LCNC) and some also model-driven engineering (MDE) tools in their work.

Note: Since these tools have so much overlap in practice among the participants, it was decided to include both LCNC and MDE in the analysis in later chapter.

The interview questions were structured around four (4) themes and an open discussion section. An interview guide was produced including questions around the theme to guide the interview.

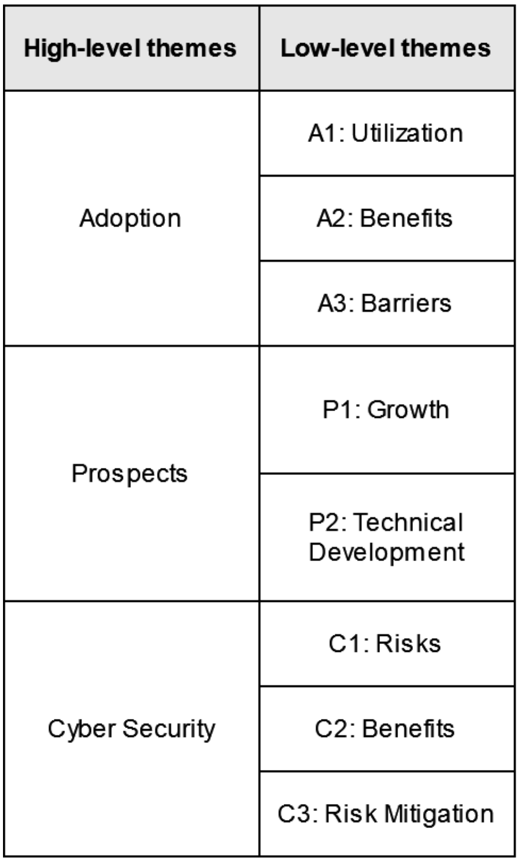
You can find the interview guide in appendices (Appendix 1). The themes in question were the following:

1. How long work experience do you have in software development or/and cyber security?
2. How have you become familiar with LC/NC technologies?
3. What is your perspective on the future developments of LC/NC technologies?
4. Have you considered the potential impact of LC/NC technologies on cyber security?
5. Free speech on or around the topic

A single interview lasted for about 30 minutes and was held in MS Teams meeting. All the interviews were recorded and transcribed. Most of the interviews were held in Finnish language and are not translated – translations are only made for the quoted statements in the next chapter. Interview participants all have a unique code with the letter P and a number afterwards.

Qualitative content analysis

Although the interviews have more of a supportive role for the previous study, it was decided to be analyzed with a systematic approach. To analyze the data produced by the semi-structured interviews a qualitative content analysis was used. Transliterations were read through, and three high-level themes and several lower-level themes were identified. After the first categorization phase a second iteration was conducted to remove unnecessary lower-level themes, renaming and adjusting the remaining themes and finally coding them. See Figure 53 below for the final themes and coding.



High-level themes	Low-level themes
Adoption	A1: Utilization
	A2: Benefits
	A3: Barriers
Prospects	P1: Growth
	P2: Technical Development
Cyber Security	C1: Risks
	C2: Benefits
	C3: Risk Mitigation

Figure 53. Qualitative content analysis theming for the interviews

The high-level themes follow the research objectives. They can be linked to the research questions RQ2 and RQ3, but the emphasis in the adoption theme differs a bit from the RQ1. The findings also provide some insight on other questions for research mentioned in chapter 1.2. Lower-level themes are derived from the higher-level themes. Some minor coding rules were set, for example, since all the participants have firsthand experience on LCNC+MDE technologies the benefits of the adoption were based only on experience. Categories, descriptions, and coding rules are described in Table 22 below.

Table 22. Categories, descriptions and coding rules of the qualitative content analysis

Category	Description	Coding rule
A1: Utilization	Current utilization of the LCNC+MDE technology. Describes the technology and its features. Describes how it could or should be used.	Current use, no past of future.
A2: Benefits of adoption	Perceived benefits after adoption of these technologies.	Perceived, measured, based on experience. No estimations or guessing.
A3: Barriers to adoption	Possible barriers hindering adoption.	Can be based on experience or estimation.
P1: Growth of the technology	Insight on the growth of LCNC+MDE technologies	Estimations of how technology will grow in future. Can be based on current developments
P2: Development of the technology	Insight on the possible future development of LCNC+MDE technologies	Estimations of how technology will develop in future. Can be based on current developments
C1: Cyber security risks	Possible risks related to LCNC+MDE technologies	Perceived, measured or estimated risks. Can include information security.

C2: Benefits on cyber security	Estimated benefits of these technologies on cyber security	Perceived, measured or estimated benefits. Can include information security.
C3: Cyber security risk mitigation	Ideas on how to mitigate possible risks related to these technologies.	Proposed measures of mitigation.

All the participants were working in software development companies. They had varying experience on the field, one newcomer with couple of years' experience and one with a 30-year experience in multiple big companies. They all had some knowledge on cyber security, and one participant was working in the security operations center (SOC). All of them had experience on LCNC platforms, but three of them had also experience on MDE along with the LCNC technologies. In many cases these two technologies overlapped so it was decided to include the MDE in the analysis as well. In this analysis cyber security category includes information security as well.

Adoption

52 mentions that it can be associated with the high-level theme of adoption was identified. In this theme lower-level themes of utilization, benefits of use and the possible barriers hindering the adoption of LCNC+MDE technologies were discussed. This was easily the most discussed category.

First lower-level theme (A1) under adoption was the current utilization of these technologies. This lower-level theme gathered 13 identified mentions. Participants discussed how they currently use these technologies in their companies, what kind of features these technologies have and how they should be implemented to serve their needs in the best way possible.

> *We have a cyber security automation software where you can use both low/no-code and then also write the same things using Python. So, we basically use a little bit of both. We use LCNC in the situation where it can be used, where it speeds up the matter and, in the situations, where, let's say, things are bit more complex then we might use code. P1*

> *For what I have and for what I understand it has been used is for creating various kinds of integrations. P3*

The second lower-level theme (A2) was about the benefits of LCNC+MDE technologies. 22 mentions were identified related to the theme. This theme discussed the benefits perceived and maybe even measured, based on experience. Observations such as significant rise in productivity, company-wide readability, better quality and the reduction on the costs were discussed.

> *So, if we see like when the customer has some requirements, we can see the system here from the sales department already what we are selling based on the conversation of the requirements. So, we know what we will most likely be delivered. We can estimate the size, schedule and budget very well because of the iterative nature. P5*

> *Third example is the homogeneousness and the rise of quality, when the best developer can create the code that the model generates, then in a way junior developers can develop such top-quality code as well. P4*

> *These kind of low-code solutions are also really good at creating data lineage. P2*

The final lower-level theme (A3) is about the possible barriers hindering the adoption of these technologies. 17 mentions could be identified to be related to this theme. The discussion included the experience from the software developer’s perspective, learning curve of a new set of tools, possible limitations and lack of control, for example. Most of barriers were presumed, since all the participants have been using these technologies and report the benefits from these technologies.

> *The company might have one or two people who understand what this is about and how it works, but they have not managed to spread the word company-wide, so the use has been scarce or completely decayed. So, it on the other hand also needs a change in the culture inside the company to start utilizing these and not do everything by the hand like previously. P4*

> *It is possible that the amount of generated code grows explosively. I was thinking that developers might not like if a simple thing creates tens of thousands rows of code, it might grow the codebase needlessly. P4*

Prospects

Only 12 mentions can be identified in the higher-level category of prospects. It is divided equally between the themes of growth of technology and future development of technology.

The theme of growth of the technology gained six (6) related mentions. It was discussed a bit less, but all the participants saw a potential growth for LCNC+MDE technologies.

> *I believe the low-code/no-code technologies will just expand and become better, conquering more of the markets and will change the field overall. P2*
>
> *In my own work I'm assuming the utilization is growing based on the benefits said earlier. Productivity and the quality increases, for example. I believe the use is going to grow, at my work its increasing and I'm myself more in the development of these technologies. P4*

The second lower-level theme under the category was the development of technology, which also gained six (6) related mentions. This was also discussed a bit less than the themes in the previous category. It was mainly discussed if the artificial intelligence (AI) tools might be connected to LCNC+MDE technologies in future. It was hard to predict the possibility, but the participants could see some potential in the unison of these technologies.

> *Could it be so that in the future it's not so pre-defined how, for example, the user interface is modeled, but you could discuss it with the AI how to create and model it. P4*

Cyber Security

Last higher-level category was the category about cyber security implications. This higher-level theme gathered 18 related mentions and it includes the lower-level themes of cyber security risks, benefits on cyber security and cyber security risk mitigation. Information security was included in this discussion.

First lower-lever theme of cyber security risks gained seven (7) related mentions. In the discussion participants discussed poorly maintained code, lack of transparency and the possibility of over-sharing as potential risks related to the technology, for example.

> *And then again when people who can’t code or have no knowledge about release cycles, for example, and they use those blocks then of course there might be bigger problems when the person using them cannot maybe evaluate is it safe or not. P1*

> *Thinking about* ****tool x**** *or* ****tool y**** *where you could create apps or reports and publish them with public access easily then it’s possible too much data is visible, or you can use the public access app to modify the data. That is a clear risk. P4*

The second lower-level theme under this category was benefits on cyber security. I was discussed briefly and gained only three (3) mentions. Quality of code and lack of human interference were mentioned as possible benefits of LCNC+MDE technologies on cyber security.

> *I would put it in a way that it has like pros and cons regarding cyber security. Like on the pro side, there's no human interference in in code. P2*

> *It’s the same with the quality. So, now the person who knows cyber security the best, like, from software development perspective and knows how to create safe code, creates it… and develops the generator creating the code. Then the junior developer who is not as skilled in security aspects the code they generate is still the safest possible that can be currently made. In this case the safety of the application increases. P4*

The final lower-level theme was about mitigation of possible cyber security risks. It gained eight (8) mentions. A lot was discussed about governing and training the users, limiting the use of “code blocks” and creating controlled sandboxed areas for use.

> *Then the back-end developers create some sort of back-end what the block queries, so the block never has a straight connection to the database but there is some sort of secure solution in between which makes sure to drop the connection immediately if there is something weird happening. P1*

> *Least privilege principle applies to these technologies as well. P4*

> *Should somehow be able to prevent them doing applications that someone else can use for malicious purposes. To brief the person and limit the access so that they have as limited set of APIs and data sources as possible to use. P4*

Two themes left out from the analysis were history and the discussion about these technologies among software developers. Some participants described the base principles and technology of

LCNC being already quite old. The theme of discussion about the topic was also left out. Participants discussed how these technologies are marketed, what kind of terminology is used and how some software developers may have strong prejudices against these technologies, for example.

5 Discussion

5.1 Summary of research questions

The research objectives of this study were to find out the current extent of LCNC technology adoption, their prospects in near or semi-near future and their possible cyber security implications. Primary research questions aimed to answer the questions of:

- RQ1: How far the adaptation of LCNC software development technology has reached in the IT field?
- RQ2: What are the prospects of the LCNC technology in the IT field in near future?

Based on the results 40 participants stated their company is already utilizing LCNC technologies and 14 participants were not sure about it. 22 participants informed their company not yet using these technologies. 59 participants out of 76 were located in Europe, 10 participants in North America and the rest of the respondents were spread across the globe.

Based on the results 53% of the respondents stated their company is already utilizing LCNC technologies. If we take this percentage as a best possible scenario and compare this result to the technology adaption lifecycle curve briefed in the literature review, we could argue LCNC technologies to have passed the early majority and already be in the late majority adaptor group. Although, it is likely that two or more participants from the same company have responded to the survey, so a safer estimate would still be somewhere in the early majority sector. These estimates are visualized in Figure 54 below.

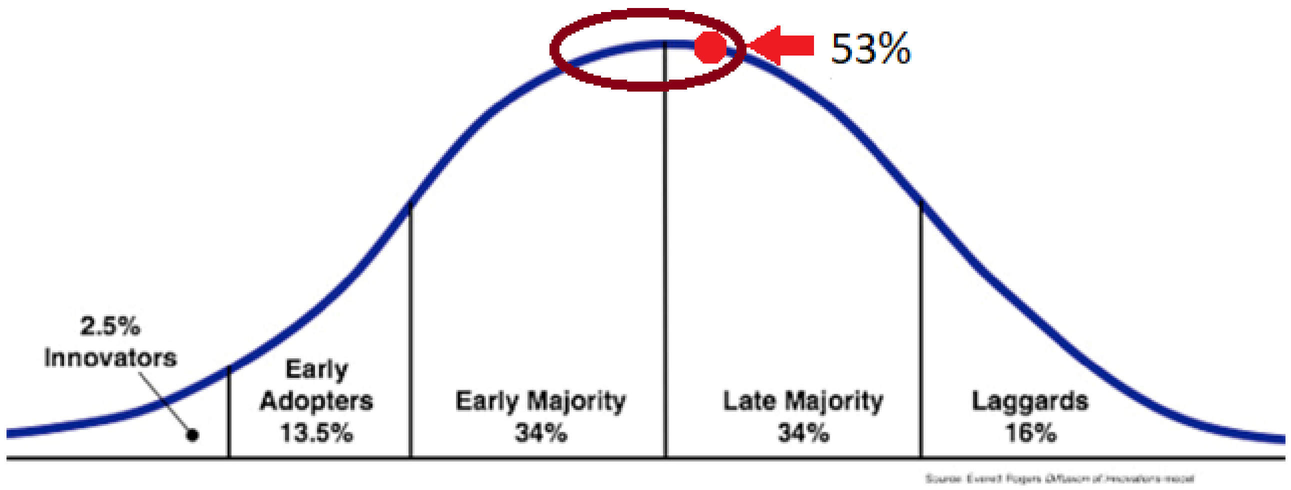


Figure 54. Technology adoption lifecycle. Adapted from “Diffusion of Innovations” by Wesley, F. Flickr, licensed under CC BY-SA 2.0.

Out of those 22 participants who stated their company is not yet utilizing LCNC technologies 59% estimated their company to adopt these technologies in one to five (1-5) years. 13,6% expected their company to adopt LCNC technologies later, but no more than in ten (10) years. Some of the respondents were not sure about this question.

The secondary research question was:

- RQ3: What are the potential cyber security threats concerning the LCNC technologies?

In addition to the estimations on the likelihood of potential cyber security risks the participants also provided insight to this question via open-text forms in the survey and in the theme interviews. Typical topics discussed related to this question were about governing the developer with less technical know-how, lack of transparency of LCDPs and poorly maintained code, for example. Means of risk mitigation were also discussed and even some benefits of LCNC technologies on cyber security.

5.2 Interpretation of findings

Multimethodological study provided diverse set of results which was analyzed with couple of techniques. A number of interesting findings were made in the analysis of the data.

When the participants who stated their company already utilizing LCNC technologies were divided into groups based on their company size it was possible to conclude that the larger company was the more they had adopted these technologies. One possible explanation is the costs – these LCNC platforms can be pricey and therefore smaller companies might not be able to utilize them at the same level. Adopting these technologies can also been seen as a risk which smaller companies are not as willing or capable to take.

It was also found out that large enterprises rated LCNC technologies to fit their needs better than small to mid-sized companies, although the differences were minor. The most satisfied group was the companies with 250-500 employees.

The familiarity with software development was compared between the participants who used LCNC in their departments/companies and those who did not. The results imply that participants in departments/companies who use LCNC technologies are not as familiar with software development as the participants who did not use LCNC in their department/companies. The possible reason for this might be the nature of LCNC technologies, since they aim to enable citizen developers with less experience on software development and therefore these companies most likely have more people with lesser technical know-how for developing software.

Plenty of interesting findings could be made when comparing the responses from participants working in cyber security departments and/or company against the responses from those who worked in software department and/or company. When the familiarity with LCNC technologies or paradigm was compared between these groups the software department/company group was recognized as being more familiar with these technologies. Nevertheless, when the adoption rate was compared it was found out that cyber security companies were using these technologies more than software companies. Participants from cyber security departments/companies even estimated their company adopting LCNC technologies faster than those from software departments/companies. Even though the study did provide some explanations how cyber security departments and companies were using these technologies the reason why cyber security companies utilized these technologies more than software companies was not recognized from the data.

In the beginning of the cyber security section of the survey the participants who had experience on cyber security stated their experience measured in years. It was noticed that companies using LCNC technologies were employing people with longer experience in cyber security. A possible reason for this might be the fact that among the respondents of this survey LCNC technologies were most utilized in the cyber security departments and/or companies.

In the second part of the cyber security section the participants estimated the likelihood of potential cyber security risks adapted from the OWASP Low-Code/No-Code Top 10 list. In many of these questions the spread of responses were pretty even, and no significant observations were made.

When estimating the likelihood of the risk of account impersonation, it was noted that smaller companies estimated the risk to be lower than larger companies. In the data leakage and possible consequences question companies who did not use LCNC technologies estimated the likelihood higher than the companies using LCNC. In both of these cases the difference might stem from the lack of experience with these technologies.

Overall, it seemed like participants from cyber security departments/companies estimated the likelihood of many of these risks higher than the groups of ‘companies using LCNC’, ‘companies not using LCNC’ and the ‘software department/company’. This is most likely because of the experience in the field and possibly even experiences with these risks.

Some observations were made on the qualitative data produced by open-text forms in the survey. When participants described how their companies were using LCNC technologies it was found out that in addition to application development there were many of participants describing their company to use LCNC to create integrations and automation. This was a bit surprising since these use cases did not pop up that much in the literary review. A possible reason for this might be that some of the participants were using model derived engineering (MDE) to automate the software generation.

In the end of the cyber security section there was a question with an open-text form where the participants were able to describe cyber security risks in greater detail. Governing the LCNC developers, especially those who are so called “citizen developers” or developers with less experience on software development, was deemed to be an important means of mitigating the possible risks. Participants described creating a “sandbox” for these developers with limited power and access to data. This was a theme which was also recognized in the literary review.

When the semi-structured theme interviews were analyzed a lot of findings were found supporting the results from the survey, but also some new themes and observations were figured out. When participants discussed their current use of LCNC technologies, it was found out that the LCNC development was used together with more traditional programming languages. The usual use case was to create custom LCNC “logic blocks” with programming languages like Python, for example.

All participants stated there being clear benefits to LCNC technologies. The increase in quality, productivity and readability were discussed. MDE was described to have exclusive benefit of estimating the price and delivery times of the software that was not recognized in non-MDE LCNC solutions. This is based on the automated generation of business logic which is in the core of MDE technology.

In the interviews the social aspects of adopting novel technologies were discussed more than in the survey. To adapt to these kind of technologies, it sometimes requires a broader change in company culture to fully support the change. This should apply to any other disruptive novel technologies as well.

The development of LCNC technology itself was discussed in the interviews - mainly the fusion of LCNC/MDE technologies with artificial intelligence (AI) solutions. Participants saw potential in the unison of these technologies and figured out some possible use cases where AI could be used to generate logic with less pre-defined manner by discussion with the AI. There was some hesitation to give estimations to this subject which is understandable. AI is developing so fast it’s hard to imagine all the possibilities related to it and LCNC technologies.

The interesting attribute of the interview discussions about the LCNC technologies and cyber security implications was the potential benefits of these technologies. One of the reasons was lack of human interference. One participant described how it's possible for the most experienced cyber security professional to create and maintain the "logic blocks" of the LCNC solution, so that when a developer with little experience uses them, they are still able to create really secure logic.

The mitigation of cyber security risks related to the LCNC technologies governing the developers were discussed like it was in the survey. In the discussion the governing was associated with the changes in the company culture discussed before. Without governing the developers, they are able to create non-secure logic and possibly overshare data.

5.3 Integration of findings with the literature

In Virtas (2018) masters' thesis research they interviewed the employees a Salesforce consulting company called Biits in Finland. Participants were divided into three groups based on their work descriptions: consultants, developers, and architects. In Virta's research there were similar observations made than in the current study. For example, the popular use case for LCNC technologies is automation. Other benefits similar to the current study were readability and the speed of development, which in the current study was referred to as increased productivity. In their results LCNC technologies were deemed inefficient with major performance issues, which was not observed in the current study. Virta's research is from 2018 so it might be possible that these technologies have developed since then (Virta, T. 2018).

In their master's thesis Alyousef (2021) conducted interviews in a large consulting firm in Netherlands. They interviewed six professionals who had at least a few years of experience in LCNC technologies. There are similarities in the findings of this study and Alyousef's research. For example, in both studies the potential barriers to the adoption of these technologies included the possible limitations and lack of control. On the other hand, in their study code collaboration, version control issues and testing were discussed, which was not observed in the results of the current study. Overall, Alyousef's research has a greater emphasis on the technical side of LCNC technologies from the software developer's perspective (Alyousef, D. 2021).

There were some similarities in the study design of the current study and Turunen’s (2022) bachelor thesis research. In their research employees from small to mid-sized Finnish information technology companies were interviewed. They found out the adoption rate to be 33% in these companies, which has increased 13% compared to previous study conducted by Arrow ECS Finland Oy (2021). In the current study the adoption rate was 53%, but it is not directly comparable, since the current study included large and global enterprises (Turunen, O. 2022).

In their master’s thesis Spets’ (2022) studied how compliant Outsystems LCDP was to the Application Security Verification Standard (ASVS). They found out that the requirements in the categories of authentication and session management were not fully met, and provided means to address these problems. Although the findings in the current research do not directly relate to the findings in Spets’ research, these studies do have a strong similarity in the focus on the cyber security implications. In their research the service provider was evaluated rather than the LCNC technologies itself, like in the current study (Spets, 2022).

The current study had some common findings compared to the former research, but was also able to produce new insight, especially on the cyber security implications related to LCNC technologies.

5.4 Reliability, limitations and ethics

Reliability

The reliability of these studies is assessed in this chapter. The multimethodological approach to the research can benefit the reliability of the results but should still be critically evaluated.

For the survey results the margin of error needed to be figured out. It was calculated with the following formula:

$$z * (sqrt(p * \frac{(1-p)}{n}))$$

- Z stands for Z-score, meaning the confidence level
- p is the estimated proportion of population with a particular attribute
- n is the sample size

It was hard to estimate the population of potential low-code/no-code developers, people considering adopting and pondering about the security aspects of these technologies in the information technology field. Therefore, the most conservative margin of error was calculated by using 0.5 as the value of p, which is the infinite value for population. The confidence level chosen for the equation was 95%, for which the value of 1.96 is used in the formula. The sample size (n) is 76.

$$1.96 * (sqrt(0.5 * \left(\frac{(1-0.5)}{76}\right))$$

With this formula the error margin of our study can be calculated to be 11,2%. Since the population was unknown, it was calculated conservatively and therefore it depicts the worst possible scenario.

76 responses are not a large sample and do affect the reliability of the survey results. Most of the responses came from Europe and because of the use of social media campaign as a data collection method, there is assumed to be a bias towards Finnish people in responses. This should be taken into account when evaluating the results with the global population.

The final section of the semi-structured theme interviews was for open discussion. When the interviewer participates freely in the discussion it is possible to cause biases.

Limitations

Self-selection sampling was selected for the survey. It was chosen as it seemed the most effective method of gathering responses. The limitation of this method is a possible selection bias which might lead to less trustworthy findings, Berndt argues in their article about sampling methods (Berndt, A. E. 2020).

The social media marketing campaign is assumed to have a bias towards people working in Finland, based on the network of the author and near colleagues who helped to share the study.

Conducting a raffle alongside the survey might be a double-edged sword. It can gather more interest to the survey among the potential respondents, but it might tempt people to skim through the survey to get to participate the raffle.

The division between information technology and cyber security professionals is a bit artificial. It can be argued that a cyber security professional is always an IT professional as well. It is also possible that the work description of responder is hard to define. For example, university lector who lecture on computer science might see themselves working in the IT field. Self-selection sampling moved this burden of consideration to the participant.

A major limitation for the data analysis was the lacking sample size of 76 responses. For this reason, it was not possible to use proper statistical analysis techniques.

Only five (5) theme interviews were conducted. The sample is too small to be considered a proper study itself and must be taken rather as a supportive addition for the survey study.

Ethical considerations

For the survey study a privacy policy was written and presented at the beginning of the survey. The anonymity of responders was assured technically within the survey service. Open text answers were checked for identifying information and anonymized if found. Respondents were given a possibility to not answer some questions, for example, the size of the company currently working for. Responding to the study was voluntary.

In the interview transliterations any names of people or companies were anonymized to protect the participants anonymity. The means of communication was chosen based on the wishes of the participant. The participants were informed that the interview is being recorded.

Data management plan was required by the university of applied sciences and was conducted for the study. A research permit was needed for sharing the study via the chat service of the university of applied sciences, and the permission was granted by the ethical committee.

6 Conclusion

6.1 Implications for practice

The study provides insight on how far the adoption of LCNC technologies is as well as what are the prospects in near future. Companies in information technology or other industries can use this information for following the trend of LCNC movement. A variety of benefits and use cases for LCNC technologies were described in the results.

A number of barriers to adopting LCNC technologies were discussed and can be used as a base for evaluating and choosing LCDP service providers. Also, suggestions on how to alter the company culture to tackle some of the possible barriers to adopting these technologies were also presented.

The data from the cyber security implication section of the study is assumed to be valuable for companies concerned on the security of these technologies. Multiple cyber security risks were listed, and the likelihood of these risks was estimated. The results provide means to mitigate these risks and present secure convention for the utilization of LCNC technologies, such as governing the developers using these development platforms and limiting the use.

6.2 Future research

The initial steps for this thesis work were taken already in late 2021 and since then there has been an increasing number of new publications on the topic of LCNC. A lot of the research is industry driven, but also novel academic work has been published in the past couple of years.

There were a few notable differences in the survey responses of participants working in small companies or in large enterprises. Based on the results smaller companies are slower to adopt LCNC technologies. Research on these findings could provide smaller companies crucial insight and possibly help them to bridge the gap in the rate of adoption.

The topic of LCNC technology security implications is still lacking and does require more research. LCNC technologies are usually provided as cloud-based services. One of the findings of this thesis research was that citizen developers or less skilled developers require governing and limitations. How should they be governed? What are the best practices for governing and limiting the use of LCDPs while keeping productivity as high as possible? This study did not provide significant insight into proper conventions for governing.

Based on the results of this study cyber security departments or companies are adopting LCND technology faster than software development departments or companies. It is possible to make an assumption why this might be, but no obvious reasons were perceived in the findings of the current study.

6.3 Final thoughts

In this master's thesis a mixed method study was designed and conducted to provide results based on the research objectives. A set of quantitative and qualitative data was produced, presented and analyzed. The study accomplished answering the research questions and also brought additional insight on the topic of low-code/no-code technologies, platforms and the cyber security implications of said technologies. The research on the topic of cyber security and low-code/no-code technologies was deemed lacking, and this study was able to produce new information on the matter. The study has its limitations which are presented in earlier chapter and should be considered when utilizing the data.

The topic and design of this thesis required a lot of iteration. Initially a more practical approach was considered: adopting and utilizing a low-code tool to create a simple CRUD application to evaluate the benefits and limitations of the technology. Although this would have served the needs of the commissioner it would most likely have less value for the professional and academic community. When the current research on the topic of low-code/no-code was explored, it was easy to find lacking areas and to adjust the study to better benefit the academic and professional community, in addition to commissioner.

The author of this thesis found the work put into the research and thesis report to be an invaluable learning experience for his professional and academic future.

References

2018 Digital Transformation Readiness Survey. (2018). Appian. https://appian.com/blog/2018/are-companies-truly-prepared-for-digital-transformation.html

The 7 Deadly Sins of Low-Code Security and How to Avoid Them. (2021). Zenity. https://www.zenity.io/the-7-deadly-sins-of-low-code-security-and-how-to-avoid-them/

About the OWASP Foundation. (n.d.). OWASP. https://owasp.org/about/

About Us. (n.d.). JYVSECTEC. https://jyvsectec.fi/about/overview/

Alyousef, Z. (2021). Challenges Development Teams Face in Low-code Development Process. https://research.ou.nl/ws/portalfiles/portal/45505497/Alyousef_Z_IM9906_AF_scriptie_PURE.pdf

An Overview of CDI. (n.d.). ExLibris. https://knowledge.exlibrisgroup.com/Primo/Content_Corner/Central_Discovery_Index/Documentation_and_Training/Documentation_and_Training_(English)/CDI_-_The_Central_Discovery_Index/010An_Overview_of_the_Ex_Libris_Central_Discovery_Index_(CDI)

Bay, J. (n.d.). What is visual scripting, and how is it used to make video games? https://www.gameindustrycareerguide.com/how-is-visual-scripting-used-in-games/

Berndt, A. E. (2020). Sampling Methods. https://doi.org/10.1177/0890334420906850

Bhattacharyya, S., Kumar, S. (2021). Study of deployment of “low code no code” applications toward improving digitization of supply chain management. https://doi.org/10.1108/JSTPM-06-2021-0084

Bloomberg, J. (2017). The Low-Code/No-Code Movement: More Disruptive Than You Realize. Forbes. https://www.forbes.com/sites/jasonbloomberg/2017/07/20/the-low-codeno-code-movement-more-disruptive-than-you-realize/?sh=29ca2975722a

Cavelty, M. (2015). Cyber-security. https://www.researchgate.net/profile/Myriam-Dunn-Cavelty/publication/281631032_Cyber-security/links/55f1426408ae199d47c243b1/Cyber-security.pdf

Cloud Security Alliance Warns Providers of 'The Notorious Nine' Cloud Computing Top Threats in 2013. (2013). Cloud Security Alliance [CSA]. https://cloudsecurityalliance.org/press-releases/2013/02/25/ca-warns-providers-of-the-notorious-nine-cloud-computing-top-threats-in-2013/

Commission Recommendation of 6 May 2003. (2003). European Commission. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32003H0361

Dillman, D. A., Smyth, J. D., Christian, L. M., & Dillman, D. A. (2014). Internet, phone, mail, and mixed-mode surveys: The tailored design method (4th ed.). Wiley.

Enterprise Low-Code Application Reviews and Ratings. (n.d.). Gartner. https://www.gartner.com/reviews/market/enterprise-low-code-application-platform

Feilzer, M. Y. (2009). Doing Mixed Methods Research Pragmatically: Implications for the Rediscovery of Pragmatism as a Research Paradigm. DOI: 10.1177/1558689809349691

Galarita, B. (2022). Information Security Vs. Cybersecurity: What's The Difference? https://www.forbes.com/advisor/education/information-security-vs-cyber-security/

Gartner Forecasts Worldwide Low-Code Development Technologies Market to Grow 20% in 2023. (2022). Gartner. https://www.gartner.com/en/newsroom/press-releases/2022-12-13-gartner-forecasts-worldwide-low-code-development-technologies-market-to-grow-20-percent-in-2023

Glossary: cyber security. (n.d.). National Institute of Standards and Technology (NIST). https://csrc.nist.gov/glossary/term/cybersecurity

Glossary: information security. (n.d.). National Institute of Standards and Technology (NIST). https://csrc.nist.gov/glossary/term/information_security

Google Scholar, About. (n.d.) Google. https://scholar.google.com/intl/en/scholar/about.html

Hurlburt, G. F. (2021). Low-Code, No-Code, What's Under the Hood? IEEE. https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9655393

Home. (n.d.). PRISMA. http://www.prisma-statement.org

The Importance of Citizen Development and Citizen IT. (2019). Gartner. https://blogs.gartner.com/jason-wong/importance-citizen-development-citizen/

Kallio, H., Pietilä, A-M., Johnson, M., Kangasniemi, M. (2016). Systematic methodological review: developing a framework for a qualitative semi-structured interview guide. Journal of Advanced Nursing. DOI: 10.1111/jan.13031

Kermanchi, A. (2022). Developer Experience in Low-Code Versus Traditional Development Platforms - A Comparative Experiment. https://aaltodoc.aalto.fi/bitstream/handle/123456789/118413/master_Kermanchi_Arian_2022.pdf?sequence=1&isAllowed=y

Kolade, C. (2022). Is WordPress a Code or No-code Tool? https://www.freecodecamp.org/news/is-wordpress-a-code-or-no-code-tool/

LCNC-SEC-01: Account Impersonation. (n.d.). OWASP. https://owasp.org/www-project-top-10-low-code-no-code-security-risks/content/2022/en/LCNC-SEC-01-Account-Impersonation

LCNC-SEC-02: Authorization Misuse. (n.d.). OWASP. https://owasp.org/www-project-top-10-low-code-no-code-security-risks/content/2022/en/LCNC-SEC-02-Authorization-Misuse

LCNC-SEC-03: Data Leakage and Unexpected Consequences. (n.d.). OWASP. https://owasp.org/www-project-top-10-low-code-no-code-security-risks/content/2022/en/LCNC-SEC-03-Data-Leakage-and-Unexpected-Consequences

LCNC-SEC-04: Authentication and Secure Communication Failures. (n.d.). OWASP. https://owasp.org/www-project-top-10-low-code-no-code-security-risks/content/2022/en/LCNC-SEC-04-Authentication-and-Secure-Communication-Failures

LCNC-SEC-05: Security Misconfiguration. (n.d.). OWASP. https://owasp.org/www-project-top-10-low-code-no-code-security-risks/content/2022/en/LCNC-SEC-05-Security-Misconfiguration

LCNC-SEC-06: Injection Handling Failures. (n.d.). OWASP. https://owasp.org/www-project-top-10-low-code-no-code-security-risks/content/2022/en/LCNC-SEC-06-Injection-Handling-Failures

LCNC-SEC-07: Vulnerable and Untrusted Components. (n.d.). OWASP. https://owasp.org/www-project-top-10-low-code-no-code-security-risks/content/2022/en/LCNC-SEC-07-Vulnerable-and-Untrusted-Components

LCNC-SEC-08: Data and Secret Handling Failures. (n.d.). OWASP. https://owasp.org/www-project-top-10-low-code-no-code-security-risks/content/2022/en/LCNC-SEC-08-Data-and-Secret-Handling-Failures

LCNC-SEC-09: Asset Management Failures. (n.d.). OWASP. https://owasp.org/www-project-top-10-low-code-no-code-security-risks/content/2022/en/LCNC-SEC-09-Asset-Management-Failures

LCNC-SEC-10: Security Logging and Monitoring Failures. (n.d.). OWASP. https://owasp.org/www-project-top-10-low-code-no-code-security-risks/content/2022/en/LCNC-SEC-10-Security-Logging-and-Monitoring-Failures

Low-Code Development Platform Market Research Report - Global Industry Analysis, Trends and Growth Forecast to 2030. (2021). Research and Markets. https://www.globenewswire.com/news-release/2020/11/10/2123468/0/en/Global-187-Billion-Low-Code-Development-Platform-Market-to-2030.html

Low-code vs. no-code app development. (n.d.) Microsoft. https://powerapps.microsoft.com/en-us/low-code-no-code-development-platforms/

Low-Code vs. No-Code: What’s the Difference? (2022). IBM Cloud Education. https://www.ibm.com/cloud/blog/low-code-vs-no-code

Mackie, K. (2021). Microsoft Power Platform Gets a Boost with Clear Software Acquisition. https://rcpmag.com/articles/2021/10/25/microsoft-clear-software-acquisition.aspx?m=1

Mayring, P. (2000). Qualitative Content Analysis. Forum: Qualitative Social Research. https://doi.org/10.17169/fqs-1.2.1089

Newcomer, E. (2022). Low code applications are essential for cybersecurity development in applications. https://www.helpnetsecurity.com/2022/02/10/low-code-applications/

Overeem, M., Jansen, S. (2021). API Management Maturity of Low-Code Development Platforms. https://www.researchgate.net/publication/352475466_API_Management_Maturity_of_Low-Code_Development_Platforms

OWASP Low-Code/No-Code Top 10. (n.d.). OWASP. https://owasp.org/www-project-top-10-low-code-no-code-security-risks/

Paré, G., Kitsiou, S. (2017). Methods for Literature Reviews. https://www.ncbi.nlm.nih.gov/books/NBK481583/

Peruzzi, J. (2023). The Pros and Cons of Low-code in Cybersecurity Environments. https://www.carahsoft.com/community/tychon-pros-and-cons-of-low-code-in-cybersecurity-blog-2023

Pratt, M. (2021). Low-code and no-code development platforms. https://www.techtarget.com/searchsoftwarequality/definition/low-code-no-code-development-platform

PRISMA 2020 flow diagram for updated systematic reviews which included searches of databases and registers only. (2020). http://www.prisma-statement.org/documents/PRISMA_2020_flow_diagram_updated_SRs_v1.docx

Richardson, C., Rymer, J. (2014). New Development Platforms Emerge for Customer-Facing Applications. https://www.forrester.com/report/New-Development-Platforms-Emerge-For-Customer-Facing-Applications/RES113411

Rogers, Everett M. (1995, original theory published in 1962). Diffusion of Innovations. Simon and Schuster. ISBN 0-02-874074-2

Ruscio, D., Kolovos, D., de Lara, J., Pierantonio, A., Tisi, M., Wimmer, M. (2022). Low-code development and model-driven engineering: Two sides of the same coin? Springer. https://link.springer.com/article/10.1007/s10270-021-00970-2

Sanchis, R., García-Perales, Ó., Fraile, F., Poler, R. (2019). Low-Code as Enabler of Digital Transformation in Manufacturing Industry. https://doi.org/10.3390/app10010012

Schwartz, K. D. (2021). App Development: Staying Secure Using Low-Code Platforms. https://www.itprotoday.com/application-security/app-development-staying-secure-using-low-code-platforms

Spets, S. (2022). Application Security Verification Standard Compliance Analysis of a Low Code Development Platform. https://www.utupub.fi/bitstream/handle/10024/173528/Spets_Sami_DI.pdf

The State of Application Development - Is IT Ready for Disruption? V2. (2019). OutSystems. https://www.outsystems.com/1/state-app-development-banking/

The State of Application Development - Is IT Ready for Disruption? V3. (2020). OutSystems. https://www.outsystems.com/1/state-app-development-trends/

The State of Low-Code/No-Code. (2021). Creatio. https://www.creatio.com/page/sites/default/files/2021-05/Report-May.pdf

Sufi, F. (2023). Algorithms in Low-Code-No-Code for Research Applications: A Practical Review. https://www.mdpi.com/1999-4893/16/2/108

Tisi, M., Mottu, J., Kolovos, D. S., de Lara, J., Guerra, E., Di Ruscio, D., Pierantonio, A., Wimmer, M. (2019). Lowcomote: Training the Next Generation of Experts in Scalable Low-Code Engineering Platforms. https://hal.science/hal-02363416/document

Tissir, N., El Kafhali, S., Aboutabit, N. (2020). Cybersecurity management in cloud computing: semantic literature review and conceptual framework proposal. https://www.researchgate.net/profile/Said-El-Kafhali/publication/344845026_Cybersecurity_management_in_cloud_computing_semantic_literature_review_and_conceptual_framework_proposal/links/61b5f81e1d88475981e3fc3b/Cybersecurity-management-in-cloud-computing-semantic-literature-review-and-conceptual-framework-proposal.pdf

Turunen, O. (2022). The use of low-code application development in Finnish SMEs. https://www.theseus.fi/bitstream/handle/10024/754855/Turunen_Oona.pdf?sequence=2&isAllowed=y

Trends on the Gartner Hype Cycle for the Digital Workplace. (2020). Gartner. https://www.gartner.com/smarterwithgartner/6-trends-on-the-gartner-hype-cycle-for-the-digital-workplace-2020

Types of programming language. (n.d.). BBC. https://www.bbc.co.uk/bitesize/guides/z4cck2p/revision/1

By Design: How Default Permissions on Microsoft Power Apps Exposed Millions. (2021). UpGuard. https://www.upguard.com/breaches/power-apps

Virta, T. (2018). Relation of low-code development to standard software development: Case Biit Oy. https://lutpub.lut.fi/bitstream/handle/10024/158441/masters_thesis_virta_tatu.pdf?isAllowed=y&sequence=1

Vähemmän koodia markkinakartoitus. (2021). Arrow ECS Finland Oy. https://www.arrow.com/ecs-media/14448/2021-02-fi-infograph.pdf

Webropol. (n.d.). Webropol. https://webropol.com/

Wesley Fryer. (2007). Diffusion of Innovations. Flickr. https://www.flickr.com/photos/wfryer/1342355056, licensed under CC BY-SA 2.0.

What is Low-Code? (n.d.) IBM. https://www.ibm.com/topics/low-code

Wiley, J & Sons. (2022). Scrivener Publishing LLC. ISBN 978-1-119-79563-6

Appendices

Appendix 1. Interview guide

1. How long work experience do you have...

- in the information technology industry?
- in the area of cyber security?

2. How have you become familiar with LC/NC technologies?

- Are you currently utilizing these technologies in your work?
- What was the reasoning behind selecting these technologies?
- What kind of benefits was expected and have the expectations been met?

3. What is your perspective on the future developments of LC/NC technologies?

4. Have you considered the potential impact of LC/NC technologies on cyber security?

- What are your thoughts on the matter?
- Can you identify the potential security threats associated with the technology?
- How would you approach managing these security concerns?

5. Free speech on or around the topic

Appendix 2. Question 16: Please describe how does your company utilize low-code technologies.

The open-text question gathered the following 26 responses:

- Custom-designed application and software for clients
- Using Microsoft Power Platform and Webcon tools to offer Custom solutions to clients,
- Easy integrations e.g using zapier
- For some apps and automations.
- A part of our online business modernization effort was done using low-code technology. The low-code part of the effort was later rewritten with traditional software development practices and the low-code paradigm was abandoned altogether.
- Integrations within internal software, CRM and JIRA
- RPA to automate business processes or finance
- We create security automation pipelines
- Teaching and training these technologies
- My company utilizes low-code technologies in automation and generating static content from dynamic sources.
- We use some lowcode frameworks/programs, usually they lack customization needs that customer wants. If needs are simple, lowcode software can be good.
- Wp sites, visual builders like elementor or divi elegant. Hubspot cms
- We build business software, from e-commerce to PIM and ERP. Some of my colleagues use low-code SaaS for integrations and CRM.
- We have a content management system for customers where they can customize their websites and handle databases.
- Mostly in prototyping and process modeling/optimization
- Personally I am working with a low-code solution to create and manage integrations between different products and tools we utilize.
- Mobile application development and production line managenet software
- A personal project involving SAP Appgyver Composer Pro to create a mobile application.
- use cookies
- Use the app.
- use programming
- Graphics team utilizes low-code on a daily basis to achieve their results. Low-code has been quite paramount in bridging the gap between software developers and artists.
- My company is a software consultancy and in some of tour customer projects our employees develop business applications for our customers using MS PowerApps on top of cloud data stores and train the customer end-users to create their own applications. This usually extends our on-prem to cloud data integration offering in situations where the customers lack their own capabilities to utilize their data stores to produce business value.
- Automated script creating
- We use metamodeling tool which allows us freedom to create multiple domain-specific modeling languages for our specific needs. Features, such as code generation or model imports from different source systems, are in our control.
- Some manual workphases are made by using LC/NC tech

Appendix 3. Question 17: Please describe how could your company utilize low-code technologies.

The open-text question gathered the following 18 responses:

- Could speed up the development of "proof of concept" type application development that we do in some projects.
- Enabling higher level of platform customization done by customers and customer service
- Some simple mockups?
- Our university should have a proper plan for low-code/no-code technologies first based on its need. Then provided suitable training programs to train its employees and educated them to be familiar with low-code/no-code technologies. It will be a systematic way and will go smoothly.
- Our company could opt for a low code platform such as Mendix, OutSystems, Oracle. The big names of Mendix and OutSystems enable a range of services including mobile, web development and wearable apps.
- Sorry, it's inconvenient
- Our company could use Low-code technology for applications that require little or no integration from backend, and has less dynamic components
- Letting our customers customize some of our products via easy to use UI, that would automatically generate the customized version.
- Would allow application specialists that have no software development knowledge to make changes to the products themselves, allowing for a quicker workflow
- One possibility would be involving and empowering non-technical employees to prototype the kinds of internal tools that they need in their daily duties. Currently, developing these tools involves a lot of guesswork and reiteration, because non-technical employees don't feel confident in communicating their technical requirements and technical employees don't have a clear understanding of user scenarios.
- We could utilize them in many ways, especially when working with people who are not very skilled in coding or does not know how to code.
- Possibly some internal tools, simple information gathering
- Doubtfully would not use it.
- Integrating various systems.
- Some simple utility tools or add-ons could perhaps be made this way. I doubt any of our core software products could be done this way, maybe never.
- Test automation
- Simpleor repetitive apps could be developed quicker and devs could focus on more complex tasks
- To shortcut development of simple Apps

Appendix 4. Question 30: Do you have any comments concerning the list of risks from OWASP in previous page? Or do you suspect any other possible risks concerning low-code/no-code technologies?

The open-text question gathered the following 21 responses:

- Not enough familiar with low/no to answer, but generally a fresh concepts are usually insecure for a while, when not tested commonly/regularly.
- Complexity and opaqueness in troubleshooting situations. Hard time to get response when platform is outsourced to another company. If I have an incident with their code the chances are all their customers will which will result in a huge risk in case of disaster
- Risk of black box is always present. Company policies is used to mitigate user dependent risks, but those cannot be guaranteed. Selection of low-code/no-code product should be done thoroughly and not to invest on product that cannot fulfill basic security scrutinies.
- Lots of not (well) maintained parts depending on the solution used
- There is no or less visibility in NL code development. Similarly, there is no access to auditing or vendor systems
- I am not too concerned with code-level technical vulnerabilities since low-code/no-code is, to a degree, a black box, and ideally something as simple as a platform update could add things like input sanitization and prepared statements to all relevant components.
- I am more concerned with the potential for human error because these technologies could potentially obfuscate some key security choices from developers, leading to a situation where they are never fully aware whether they are creating secure or vulnerable software.
- I think the risk lies down in the process of making these low/nocode technologies and how you are going to restrict usage and application engagement with the end users or are you even going to.
- Depends a lot on the maturity of the solution, and the purpose of the tool implemented using a LC/NC.
- The platforms are not particularly cheap, so cost creep over time is possible.
- I would use such platforms for simple non-business critical needs, where also the cyber security risks are not high.
- account impersonation
- authorization abuse
- Data breaches and unintended consequences
- The technologies carry a big risk factor on the business side from my experience. Often investors or managers who are not familiar with the IT field might employ people to high risk tasks which should be carried out by more senior staff. While effective in the short-term business-sense and with junior staff hiring, later on the lack of design structure starts to cause an escalation of technical depth.
- From my perspective, low-code/no-code staff should always have a senior "hard code" staff member supervising their work in order to prevent technical mistakes and control the overall technical design. Low-code staff has been rarely qualified to analyze technical approaches in design, compared to junior programmers who might be more familiar with the hard technical concepts behind the technologies.
- This might solve itself in the future if the approach matures, but it might also cause a shift to more predatory development standards when the commitment level drops lower and mitigates the long term risks.
- If I made an app using low-code/no-code technologies, I'd also be worried about my further ability to fine tune its performance if needed.
- Many of the risks are fairly trivial to mitigate if you know what you are doing, but the entire point of no code is to let people who don't know what they are doing to do things. I don't know if that is

a solvable problem. Perhaps we can give such people inherently limited credentials for them to pass to the no code platform.

- On the other hand no code can be very declarative, which could enable it to know fairly precisely what rights it actually needs and why. This could allow very precisely crafted access rights; a good thing.
- Risks depends on used low-code/no-code tool, to some none from the list applies. Often risks have to be mitigated by testing tools to figure out how they work in certain situations, especially SaaS tools.