ҚАЗАҚСТАН РЕСПУБЛИКАСЫНЫҢ ҒЫЛЫМ ЖӘНЕ ЖОҒАРЫ БІЛІМ МИНИСТРЛІГІ
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ КАЗАХСТАН
MINISTRY OF SCIENCE AND HIGHER EDUCATION OF THE REPUBLIC OF KAZAKHSTAN

ХАЛЫҚАРАЛЫҚ АҚПАРАТТЫҚ ЖӘНЕ КОММУНИКАЦИЯЛЫҚ ТЕХНОЛОГИЯЛАР ЖУРНАЛЫ

МЕЖДУНАРОДНЫЙ ЖУРНАЛ ИНФОРМАЦИОННЫХ И КОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ

INTERNATIONAL JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGIES

2024 (20) 4

қазан - желтоқсан

ISSN 2708–2032 (print)
ISSN 2708–2040 (online)

Халықаралық ақпараттық және коммуникациялық технологиялар журналы
ISSN 2708–2032 (print)
ISSN 2708–2040 (online)
Меншіктенуші: «Халықаралық ақпараттық технологиялар университеті» АҚ (Алматы қ.).
Қазақстан Республикасы Ақпарат және әлеуметтік даму министрлігінің Ақпарат комитетінде – **20.02.2020** жылы берілген.
№ KZ82VPY00020475 мерзімдік басылым тіркеуіне қойылу туралы куәлік.
Тақырыптық бағыты: ақпараттық технологиялар, әлеуметтік-экономикалық жүйелерді дамытудағы цифрлық технологиялар, ақпараттық қауіпсіздік және коммуникациялық технологияларға арналған.
Мерзімділігі: жылына 4 рет.
Тиражы: 100 дана
Редакцияның мекенжайы: 050040, Алматы қ-сы, Манас к-сі, 34/1, 709-кабинет, тел: +7 (727) 244-51-09.
E-mail: ijict@iitu.edu.kz Журнал сайты: https://journal.iitu.edu.kz

Международный журнал информационных и коммуникационных технологий
ISSN 2708–2032 (print)
ISSN 2708–2040 (online)
Собственник: АО «Международный университет информационных технологий» (г. Алматы).
Свидетельство о постановке на учет периодического печатного издания в Министерство информации и общественного развития Республики Казахстан **№ KZ82VPY00020475,** выданное от **20.02.2020 г.**
Тематическая направленность: информационные технологии, информационная безопасность и коммуникационные технологии, цифровые технологии в развитии социо-экономических систем**.**
Периодичность: 4 раза в год.
Тираж: 100 экземпляров.
Адрес редакции: 050040 г. Алматы, ул. Манаса 34/1, каб. 709, тел: +7 (727) 244-51-09.
E-mail: ijict@iitu.edu.kz Сайт журнала: https://journal.iitu.edu.kz

«International Journal of Information and Communication Technologies»
ISSN 2708–2032 (print)
ISSN 2708–2040 (online)
Owner: International Information Technology University JSC (Almaty).
The certificate of registration of a periodical printed publication in the Ministry of Information and Social Development of the Republic of Kazakhstan, Information Committee No. KZ82VPY00020475, issued on 20.02.2020.
Thematic focus: information technology, digital technologies in the development of socio-economic systems, information security and communication technologies
Periodicity: 4 times a year.
Circulation: 100 copies.
Editorial address: 050040. Manas st. 34/1, Almaty. +7 (727) 244-51-09. E-mail: ijict@iitu.edu.kz
Journal website: https://journal.iitu.edu.kz

МАЗМҰНЫ

ӘЛЕУМЕТТІК-ЭКОНОМИКАЛЫҚ ЖҮЙЕЛЕРДІ ДАМЫТУДАҒЫ ЦИФРЛЫҚ ТЕХНОЛОГИЯЛАР

АҚПАРАТТЫҚ ТЕХНОЛОГИЯЛАР

АҚПАРАТТЫҚ ҚАУІПСІЗДІК ЖӘНЕ КОММУНИКАЦИЯЛЫҚ ТЕХНОЛОГИЯЛАРҒА АРНАЛҒАН

СОДЕРЖАНИЕ

ЦИФРОВЫЕ ТЕХНОЛОГИИ В РАЗВИТИИ СОЦИО-ЭКОНОМИЧЕСКИХ СИСТЕМ

ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

ИНФОРМАЦИОННАЯ БЕЗОПАСНОСТЬ И КОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ

CONTENT

DIGITAL TECHNOLOGIES IN THE DEVELOPMENT OF SOCIO-ECONOMIC SYSTEMS

INFORMATION TECHNOLOGY

INFORMATION SECURITY AND COMMUNICATION TECHNOLOGIES

INTERNATIONAL JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGIES
ISSN 2708–2032 (print)
ISSN 2708–2040 (online)
Vol. 5. Is. 4. Number 20 (2024). Pp. 83–94
Journal homepage: https://journal.iitu.edu.kz
https://doi.org/10.54309/IJICT.2024.20.4.007
УДК 004.43

ANALYSIS OF THE APPLICATION OF LOW CODE AND NO-CODE TECHNOLOGIES IN SOFTWARE PRODUCT DEVELOPMENT

D.A. Rakhmetullina
International Informational Technology University, Almaty, Kazakhstan.
E-mail: darigarakh23@gmail.com

Rakhmetullina Dariga Assanovna — master's student in IT Project Management, International Information Technology University, Almaty, Kazakhstan
E-mail: darigarakh23@gmail.com, https://orcid.org/0009-0003-0179-2505.

Abstract. The proliferation of low-code and no-code (LC/NC) technologies marks a transformative era in software product development, offering innovative solutions that democratize the creation process and streamline development efforts. This article explores the implications and applications of LC/NC technologies, focusing on their potential to empower users without extensive coding expertise to participate in software creation. Through a com-prehensive analysis of ten prominent LC/NC platforms, including OutSystems, PowerApps, Mendix, Bubble, Webflow, Flutterflow, Zapier, Tilda, Creatio, and WordPress, the study elu-cidates the underlying models and languages of these platforms. It examines case studies of successful LC/NC implementations, analysing their benefits, such as reduced development time and increased agility, alongside challenges like limited customization and potential se-curity vulnerabilities. Moreover, the research explores the impact of LC/NC technologies on traditional roles within software development teams and the evolving skillsets required in the industry. While recognizing that LC/NC platforms will not entirely replace traditional cod-ing, this study highlights their significant role in fostering innovation, inclusivity, and agility in software development. This paper contributes to understanding how LC/NC technologies are reshaping the field of software development and anticipates future trends in the technol-ogy industry.

Keywords: low-code and no-code technologies, software development democratization, efficiency, customization, security, industry innovation and future trends

For citation: D.A. Rakhmetullina. ANALYSIS OF THE APPLICATION OF LOW CODE AND NO-CODE TECHNOLOGIES IN SOFTWARE PRODUCT DEVELOPMENT//INTERNATIONAL JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGIES. 2024. Vol. 5. No. 20. Pp. 83–94 (In Eng.). https://doi.org/10.54309/IJICT.2024.20.4.007.

БАҒДАРЛАМАЛЫҚ ӨНІМДЕРДІ ӘЗІРЛЕУДЕ LOW CODE ЖӘНЕ NO-CODE ТЕХНОЛОГИЯЛАРЫНЫҢ ҚОЛДАНЫЛУЫН ТАЛДАУ

Д.А. Рахметуллина
Халықаралық ақпараттық технологиялар университеті, Алматы, Қазақстан.
E-mail: darigarakh23@gmail.com

Рахметуллина Дарига Асановна — "IT жобаларын басқару" білім беру бағдарламасының магистранты, Халықаралық ақпараттық технологиялар университеті, Алматы, Қазақстан E-mail: darigarakh23@gmail.com, https://orcid.org/0009-0003-0179-2505.

Аннотация. Low-code және no-code (LC/NC) технологиялардың таралуы бағдарламалық өнімдерді әзірлеудегі трансформациялық дәуірді белгілейді, құру процесін демократияландыратын және әзірлеу әрекеттерін оңтайландыратын инновациялық шешімдерді ұсынады. Бұл мақалада LC/NC технологияларының салдары мен қолданылуын зерттейді, олардың кодтау бойынша кең тәжірибесі жоқ пайдаланушыларға бағдарламалық жасақтама жасауға қатысуға мүмкіндік беру әлеуетіне назар аударады. OutSystems, PowerApps, Mendix, Bubble, Webflow, Flutterflow, Zapier, Tilda, Creatio және WordPress сияқты ОН танымал LC/NC платформаларын жан-жақты талдау арқылы зерттеу осы платформалардың негізгі үлгілері мен тілдерін түсіндіреді. Ол LC/NC сәтті енгізулерінің жағдайлық зерттеулерін зерттейді, олардың даму уақытын қысқарту және икемділікті арттыру сияқты артықшылықтарын, сондай-ақ шектеулі теңшеу және ықтимал қауіпсіздік осалдықтары сияқты мәселелерді талдайды. Сонымен қатар, зерттеу LC/NC технологияларының бағдарламалық жасақтаманы әзірлеу топтарындағы дәстүрлі рөлдерге және салада талап етілетін дамып келе жатқан дағдыларға әсерін зерттейді. LC / NC платформалары дәстүрлі кодтауды толығымен алмастырмайтынын мойындай отырып, бұл зерттеу олардың бағдарламалық жасақтаманы әзірлеу саласындағы инновацияларды, инклюзивтілікті және ептілікті дамытудағы маңызды рөлін көрсетеді. Тұтастай алғанда, бұл жұмыс LC/NC технологияларының бағдарламалық жасақтаманы әзірлеу саласын қалай өзгертетінін түсінуге ықпал етеді және технологиялық индустриядағы болашақ тенденцияларды болжайды.

Түйін сөздер: Low-code және no-code технологиялары, бағдарламалық жасақтаманы демократияландыру, тиімділік, теңшеу, қауіпсіздік, саладағы инновациялар және болашақ тенденциялар

Дәйексөздер үшін: Д.А. Рахметуллина. БАҒДАРЛАМАЛЫҚ ӨНІМДЕРДІ ӘЗІРЛЕУДЕ LOW CODE ЖӘНЕ NO-CODE ТЕХНОЛОГИЯЛАРЫНЫҢ ҚОЛДАНЫЛУЫН ТАЛДАУ //ХАЛЫҚАРАЛЫҚ АҚПАРАТТЫҚ ЖӘНЕ КОММУНИКАЛЫҚ ТЕХНОЛОГИЯЛАР ЖУРНАЛЫ. 2024. Т. 5. No. 20. 83–94 бет. (ағылшын тілінде). https://doi.org/10.54309/IJICT.2024.20.4.007.

АНАЛИЗ ПРИМЕНЕНИЯ ТЕХНОЛОГИЙ LOW CODE И NO-CODE В РАЗРАБОТКЕ ПРОГРАММНЫХ ПРОДУКТОВ

Д.А. Рахметуллина
Международный университет информационных технологий, Алматы, Казахстан.
E-mail: darigarakh23@gmail.com

Рахметуллина Дарига Асановна — магистрант образовательной программы «Управление IT-проектами», Международный университет информационных технологий, Алматы, Казахстан. E-mail: darigarakh23@gmail.com, https://orcid.org/0009-0003-0179-2505.

Аннотация. Распространение технологий low-code и no-code (LC/NC) знаменует собой эпоху преобразований в разработке программных продуктов, предлагая ин-новационные решения, которые демократизируют процесс создания и оптимизируют усилия по разработке. В статье исследуются последствия и применение технологий LC/NC и уделяется особое внимание их потенциалу, позволяющему пользователям без обширных знаний в области кодирования участвовать в создании программного обеспечения. Благодаря всестороннему анализу десяти известных платформ LC/NC, включая OutSystems, PowerApps, Mendix, Bubble, Webflow, Flutterflow, Zapier, Tilda, Creatio и WordPress, исследование раскрывает базовые модели и языки этих платформ. В нем рассматриваются тематические исследования успешных реализаций LC/NC, анализируются их преимущества, такие как сокращение времени разработки и повы-шение гибкости, а также такие проблемы, как ограниченная настройка и потенциаль-ные уязвимости безопасности. Кроме того, исследование изучает влияние технологий LC/NC на традиционные роли в командах разработчиков программного обеспечения и развивающиеся наборы навыков, необходимые в отрасли. Признавая, что платформы LC/NC не заменят полностью традиционное программирование, это исследование подчеркивает их важную роль в содействии инновациям, инклюзивности и гибкости в сфере разработки программного обеспечения. В целом, эта статья способствует пони-манию того, как технологии LC/NC меняют область разработки программного обеспе-чения и предвосхищают будущие тенденции в технологической отрасли.

Ключевые слова: технологии low-code и no-code, демократизация разработки ПО, эффективность, кастомизация, безопасность, инновации в отрасли и будущие тенденции

Для цитирования: Д.А. Рахметуллина. АНАЛИЗ ПРИМЕНЕНИЯ ТЕХНОЛОГИЙ LOW CODE И NO-CODE В РАЗРАБОТКЕ ПРОГРАММНЫХ ПРОДУКТОВ//МЕЖДУНАРОДНЫЙ ЖУРНАЛ ИНФОРМАЦИОННЫХ И КОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ. 2024. Т. 5. No. 20. Стр. 83–94. (На англ.). https://doi.org/10.54309/IJICT.2024.20.4.007.

Introduction

The software development industry has been significantly transformed with the advent of Low-Code and No-Code (LCNC) platforms. Designed to lower or eliminate the need for complex coding skills, these platforms open the world of software creation to a broader audience. Traditionally, software development demanded specialized expertise, often limit-ing the ability to develop applications to those with formal training in programming (Frank et al., 2021: 1–6). LCNC platforms democratize this process, enabling even non-technical users

to contribute meaningfully to software development through intuitive tools and simplified interfaces. This transformation is particularly impactful for small and medium-sized enterprises (SMEs), which often face limitations in terms of IT budgets and skilled personnel. By leveraging LCNC technologies, SMEs can now participate in digital innovation, engage in rapid prototyping, and deploy applications without significant upfront investment, helping to level the playing field across industries (Gartner, 2021).

The relevance of studying LCNC platforms stems from the accelerating pace of technological advancement in software engineering. Businesses increasingly seek agile and responsive solutions that can quickly adapt to evolving market demands. LCNC technologies cater to these needs by offering faster development cycles and flexible solutions that align well with modern business requirements. However, questions remain regarding the sustainability, applicability, and scalability of LCNC technologies, especially in complex development scenarios. A thorough understanding of these aspects is crucial as LCNC platforms gain traction in the software development sector.

This study aims to explore the role of LCNC technologies in software product development by identifying their advantages, limitations, and future potential within the industry. The study will analyze the core principles that underpin LCNC platforms, review case studies from organizations that have implemented these technologies and assess the impact of LCNC adoption on development processes, product quality, and overall outcomes. In addition, this research seeks to provide actionable recommendations for businesses to adopt and implement LCNC platforms effectively, thus enhancing productivity and innovation.

Materials and Methods

This research is built on a structured analysis of LCNC technologies through both theoretical and empirical methodologies. The foundation of the study includes a comprehensive review of relevant literature on Low-Code and No-Code platforms, complemented by case studies and real-world applications within the software development industry. The case studies are selected from a range of industries, providing diverse perspectives on the deployment, effectiveness, and challenges associated with LCNC technologies. Empirical data collection involves both qualitative and quantitative methods. Interviews with industry professionals, software developers, and business stakeholders provide insights into the practical applications and limitations of LCNC platforms, while surveys gather quantitative data on the usage, impact, and outcomes of LCNC adoption. The study also conducts a comparative analysis of development projects that utilize LCNC platforms versus those employing traditional development methods. Statistical data from these analyses highlight trends, measure effectiveness, and contribute to an understanding of LCNC technologies' role within software development.

The study's methodology further includes an evaluation of specific LCNC platforms — alongside an analysis of key metrics, including development time, resource allocation, and product quality. This approach allows for a comprehensive examination of LCNC technologies in real-world contexts, balancing both theoretical frameworks and empirical evidence.

Results and Discussion

The findings from this research reveal significant insights into the advantages, limitations, and future applications of Low-Code and No-Code technologies in software development. One of the key advantages identified is the reduction in development time, as LCNC platforms streamline application building through pre-configured templates and reusable modules. This efficiency has been shown to improve productivity, particularly in ag-

ile development environments, allowing for faster responses to business needs and market changes. Moreover, the flexibility offered by LCNC platforms aligns well with modern business demands, enabling rapid iterations and adaptations based on user feedback and shifting requirements.

However, the research also highlights notable limitations of LCNC technologies, especially in complex or highly customized projects. While LCNC platforms excel in providing accessible solutions for straightforward applications, they often struggle with scalability and may lack the depth of customization required in more intricate development scenarios. Security remains another concern, as the abstraction layers inherent in LCNC platforms can obscure potential vulnerabilities. This limitation suggests that while LCNC platforms are highly beneficial for prototyping and less complex projects, traditional coding may still be necessary for mission-critical or highly tailored applications.

The study's results indicate that LCNC technologies are likely to coexist with traditional software development methods rather than replace them. They offer a complementary approach that fosters inclusivity and agility, particularly valuable for SMEs and teams focused on rapid development. Recommendations from this research outline strategies for integrating LCNC technologies within organizations, such as selecting projects best suited for LCNC, training teams in platform-specific best practices, and establishing security protocols tailored to LCNC environments.

The landscape of software development has continually evolved in response to advancing technologies, changing market demands, and the imperative for enhanced efficiency. Within this transformative landscape, Low Code and No-Code (LCNC) platforms have emerged as game changers, providing tools that democratize software development and enhance the speed of application deployment. This review delves deeper into LCNC's rise, applications, challenges, and future implications (Brooke, 2013: 29–40).

In the vast realm of technology, there has always been a tussle between advancing at the speed of light and ensuring that the masses can keep up with such progress. The field of software development, once exclusive to those trained in the intricate nuances of coding languages, has experienced a radical shift in this balance in recent years. Emerging from the shadows of traditional, code-intensive development processes, Low Code and No-Code (LCNC) platforms have begun to challenge the established norms, promising a democratized landscape where software creation is no longer the domain of the few but the playground of many (Hakimi, 2019).

This literature review aims to dissect the evolution, significance, challenges, and potential trajectories of LCNC technologies, examining their transformative impact on software product development. The dawn of LCNC platforms has been nothing short of revolutionary, offering a unique blend of accessibility and efficiency. At its core, this paradigm shift aims to simplify the complex, making software development more intuitive and less dependent on deep technical prowess. But like all seismic shifts in technology, the rise of LCNC platforms brings with it a host of questions, potential pitfalls, and uncharted territories that merit thorough examination (Shah, 2020: 15).

To appreciate the current state of Low Code and No-Code (LCNC) platforms, it's imperative to traverse back in time and understand the trajectory that brought us here. Historically, software development was an exclusive realm, a citadel for those equipped with specialized training. The 1960s and 70s, often referred to as the 'dark ages' of computing, were dominated by punch-card systems and mainframes. These cumbersome processes necessitat-

ed proficiency in arcane programming languages and a deep understanding of complex systems. As we transitioned into the 1980s, the personal computer revolution began (Frank et al., 2021: 1–6). Tools like Microsoft's MS-DOS and later Windows made computers accessible to the masses. But while hardware became increasingly user-friendly, software development retained its high entry barrier. The 90s introduced visual programming languages, such as Visual Basic, which attempted to simplify the coding process, but still required a significant amount of technical know-how.

The turn of the millennium saw a surge in internet usage, driving demand for software solutions. Traditional development cycles lagged, leading to the emergence of LCNC platforms, simplifying web development with tools like WYSIWYG editors. LCNC platforms democratized software development, enabling a wider audience to participate, and levelling the playing field for SMEs. Studies by Frank, Maier, and Bock (2021) and others highlight the traction and productivity benefits of low-code platforms, marketed for their efficiency and reduction in hand-coding. Visual elements and citizen development further empower non-experts to build applications quickly and affordably, as seen in success stories like T-Mobile US Inc.'s rapid app development (Schneid et al., 2021).

Furthermore, market research firms have made bold forecasts regarding the market size for low-code platforms (LCPs), catching the attention of investors. Forrester predicts a 15 % growth in the LCP market, reaching USD 7.7 billion in 2021, while Brandessence estimates the market was already worth nearly USD 13 billion in 2020, projecting it to reach USD 65 billion by 2027.

This growing interest in low-code platforms is not confined to the industry sector; it has also garnered significant attention in academic circles, leading to dedicated workshops and discussions within the scholarly community.

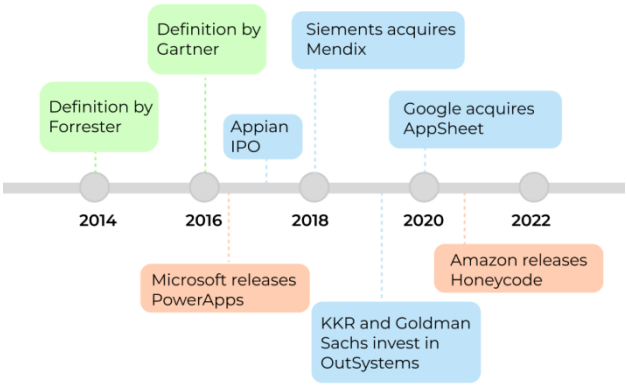

Fig.1. Major events in low-code history

As highlighted by Di Ruscio, Kolovos, de Lara, et al. (2022), the year 2021 saw a notable trend among major cloud service providers incorporating low-code development platforms (LCDPs) into their cloud solutions. This movement was initiated by Microsoft with the launch of its Power Apps LCDP in November 2016. Google also entered this space by acquiring AppSheet in January 2020, subsequently making it their primary low code offering. Additionally, Amazon introduced Honeycode in June 2020, an LCDP designed for web and mobile application development (Gurcan et al., 2021: 67–72).

The concept of no-code development platforms (NCDPs) is closely related to LC-

DPs. NCDPs are characterized by their ability to remove the need for traditional programming using visual languages, graphical user interfaces, and configuration tools. Despite its widespread use in marketing, the term NCDP is not universally accepted by market analysis firms as a distinct market segment. In the context of their research, Di Ruscio and colleagues treat NCDPs and LCDPs as interchangeable terms, focusing solely on LCDPs for their analysis.

According to Frank, Maier, and Bock (2021), Low Code Platforms (LCPs) are increasingly recognized for their critical role in the ongoing digitization of various sectors, including business, personal, and social domains. The widespread integration of software into different facets of life underscores its transformative impact. Consequently, the ability to swiftly develop, adapt, and deploy software has become a key competitive factor for many companies. This relevance extends to the academic sphere, where despite longstanding research into efficient software design and alignment, numerous challenges persist, presenting opportunities for innovative research, particularly in the realm of LCPs. The study also highlights a notable lack of clarity and consistency in the conceptualization of LCPs (Di Ruscio et al., 2022: 437–446). The market offers a broad spectrum of products under the LCP label, with diverse features and descriptions often obscured by marketing rhetoric. Gartner's definition, while comprehensive, lacks specificity, especially in terms of high-level abstractions and metadata-based programming languages, concepts that are prevalent in various software development systems (Gartner, 2021).

According to research by Waszkowski (2019), Low-Code Development Platforms (LCDPs) represent a significant shift in the approach to software development, emphasizing graphical user interfaces over traditional hard-coded programming techniques. LCDPs focus on developing databases, business processes, and user interfaces, particularly for web-based applications. This methodology stems from the ideology of fourth-generation programming (4GL) and incorporates principles of Rapid Application Development (RAD), automatic code generation, and visual programming. Introduced in 2011, low-code programming allows developers to focus more on the aesthetics and functionality of an application rather than on the syntax of the code, thereby reducing troubleshooting and implementation time (Brooke, 2013: 29–40).

The study also sheds light on the growing demand for IT professionals in Europe, with an estimated 275,000 unfilled vacancies in the ICT sector and projections suggesting this shortfall could reach one million by 2020. Gartner research indicates that by 2021, the demand for information systems will outpace the supply capabilities of IT departments by a factor of five. This gap underscores the timeliness and relevance of low-code platforms, which enable faster creation and modification of applications without extensive coding (Martinez et al., 2023: 1–7).

Key global players in the low-code platform market include Salesforce, Microsoft PowerApps, Mendix, Google App Maker, Tilda, Flutterflow, Wordpress, TrackVia, and Appian. In Poland, while there are fewer providers, the available low-code solutions often build upon European or American technologies, enhancing them with customized elements and modules (Frank et al., 2021: 1–6).

The implementation of low-code platforms in automating business processes, especially in manufacturing, poses a complex challenge, necessitating a range of research and technological solutions, including basic, applied, diagnostic, verification, and heuristic research.

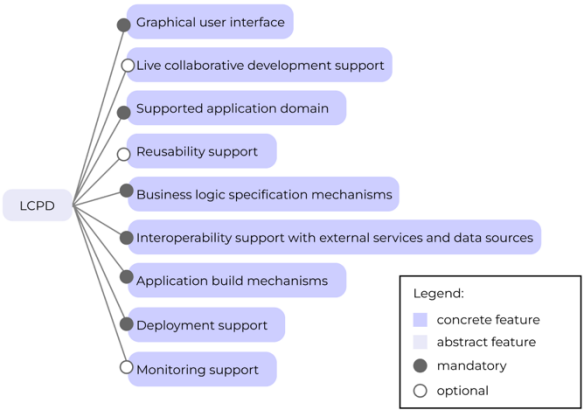

Fig.2. Top-level features of LCDPs

As detailed by Di Ruscio, Kolovos, de Lara, et al. (2022), Low-Code Development Platforms (LCDPs) facilitate the creation of versatile applications, suitable for web-only environments or as native applications for specific deployment environments. These platforms are adept at supporting both desktop and mobile devices and seamlessly integrating with prevalent Software-as-a-Service (SaaS) applications such as Zapier, Amazon AppFlow, and Trello. Among the notable LCDPs, Appian stands out for its longevity, while Amazon Honeycode and Google AppSheet represent more recent developments in this domain (Di Ruscio, et al., 2022: 437–446).

A key feature distinguishing LCDPs is their user experience, particularly the advanced graphical user interfaces they offer. These interfaces include tools and widgets that empower citizen developers to design applications effectively (Waszkowski, 2019: 376–381). Functionalities like drag-and-drop interfaces, sophisticated reporting features, decision engines for complex logic modeling, and form builders are integral to the front-end experience of LCDPs. Furthermore, LCDPs often provide live collaborative development support, which is especially beneficial for developers working remotely or in geographically dispersed teams. This feature enhances the collaborative aspect of developing applications (Kalpeyeva et al., 2020: 1–6).

The analysis delves into survey data gathered from 100 respondents, aiming to elucidate their perspectives and engagement with Low-Code/No-Code (LCNC) platforms in software development. Key quantitative findings highlight widespread awareness and adoption of LCNC technologies, with 85 % of respondents indicating familiarity and 71 % integrating these platforms into their project workflows. Tilda emerges as the most utilized platform (40 %), followed closely by PowerApps, Flutterflow, and WordPress. LCNC platforms are predominantly employed for web application development (52 %), with notable usage also observed in business process automation, prototyping, and mobile application development. Respondents overwhelmingly attest to the acceleration of project development facilitated by LCNC platforms (72 %), with visual programming emerging as the most satisfying aspect (47 %). A significant majority (73 %) express intentions to either maintain or increase their usage levels, underlining perceived benefits and potential for innovation within the LCNC landscape. These observations offer a holistic perspective on LCNC platforms within software development, detailing their benefits and potential directions for future development.

Navigating the diverse landscape of Low-Code/No-Code (LCNC) platforms in software development unveils platforms like Flutterflow, Tilda, and Zapier, which prioritize user-friendly interfaces. For instance, Flutterflow offers a visually intuitive platform for building mobile apps, ideal for entrepreneurs venturing into app development without extensive coding knowledge. Tilda, known for its drag-and-drop website builder, caters to small businesses and individuals seeking to establish an online presence swiftly. Zapier, specializing in automation, serves professionals seeking to streamline repetitive tasks across various applications, like automatically saving email attachments to cloud storage (Wong et al., 2019).

Conversely, WordPress, while renowned for its versatility, requires a certain technical aptitude, particularly for advanced customization. For instance, a freelance web developer might leverage WordPress's extensive plugin ecosystem to create a bespoke e-commerce platform for a client, but integrating custom features may necessitate familiarity with HTML, CSS, or PHP (Pantelimon et al., 2019: 566–571).

In contrast, enterprise-grade platforms like Creatio offer robust customization options tailored to complex business processes. For example, a multinational corporation might utilize Creatio's workflow orchestration tools to streamline procurement processes across multiple departments, ensuring regulatory compliance and efficiency.

While WordPress boasts a vibrant community of developers and users contributing plugins and support, emerging platforms like OutSystems and PowerApps are steadily building their support networks. These platforms target businesses seeking scalable solutions for app development, with OutSystems catering to large enterprises requiring intricate integrations and PowerApps offering a more accessible option for small to midsize businesses looking to digitize workflows.

Understanding these nuances is crucial for organizations seeking the right LCNC platform to meet their specific development needs and objectives. Whether it's empowering non-technical users to create web content, streamlining business processes, or building sophisticated enterprise applications, selecting the appropriate platform ensures efficient and effective software development (Sanchis et al., 2020: 1–7).

Low Code and No-Code (LC/NC) technologies democratize software development, letting users create applications and websites without deep coding knowledge. They use models and languages to abstract complexity, speeding up development. OutSystems uses a model-driven approach with visual tools and reusable components, allowing users to define logic visually, boosting collaboration. PowerApps (Microsoft) also employs a model-driven approach with visual tools and data connectors, letting users create custom applications without coding. Mendix focuses on model-driven development with visual tools and reusable components, simplifying app development and deployment. Bubble offers a visual development model, allowing users to design web apps through drag-and-drop, with a proprietary language for logic and workflows. Webflow enables visual website creation with design tools and templates, simplifying website development and customization. Flutterflow, utilizing Flutter framework, streamlines cross-platform app development with visual UI design and Dart language for behavior definition (Valsamakis et al., 2020: 1–4). Zapier automates workflows through a web interface, simplifying task integration and process automation with rule-based logic. Tilda uses a block-based model for web page creation via a visual editor, facilitating rapid website customization without coding. Creatio specializes in CRM and BPM solutions, employing a model-driven approach for process automation, enhancing business efficiency visually. WordPress, acting as a CMS and website builder, offers a plugin architec-

ture and template system for customization, with a visual dashboard for content management, reducing the need for PHP coding. Understanding LC/NC models and languages is crucial for leveraging their potential in driving innovation and digital transformation, making software development more inclusive and agile (Bock et al., 2021: 733–740).

The advent of Low-Code and No-Code (LC/NC) technologies has significantly reshaped the software development landscape, offering faster development times and accessibility to non-technical users. However, along with their benefits, these platforms come with their own unique challenges and limitations. LC/NC platforms often prioritize ease of use over granular control, limiting customization options for complex applications (Sahay et al., 2020: 171–178). Additionally, the abstraction layers inherent in these platforms can lead to performance bottlenecks as applications scale up and integrating them seamlessly with existing IT ecosystems remains a challenge, especially in complex enterprise environments. While LC/NC platforms facilitate easy development and deployment, they may inadvertently increase data security risks due to novice users overlooking vulnerabilities (Cabot, 2020: 535–538). Moreover, ensuring compliance with regulatory frameworks demands meticulous adherence during application development and deployment. Proficient utilization of LC/NC platforms requires comprehensive training in platform functionalities and software engineering principles. Additionally, organizations must be cautious of over-reliance on vendors for updates and enhancements, which can lead to dependency risks. Integrating LC/NC methodologies requires comprehensive change management strategies to facilitate organizational transition (Daniel et al., 2020: 15332–15346). Moreover, the democratization of application development may lead to unauthorized deployment of unsanctioned applications, posing governance challenges. Decentralized development across LC/NC platforms necessitates robust governance mechanisms to ensure application quality. Orchestrating projects across diverse platforms requires streamlined project governance frameworks. Staying agile in assimilating emerging trends and technological advancements is essential for organizations leveraging LC/NC technologies (Chang et al., 2017: 50–55). Additionally, strategic selection of platforms requires vigilant evaluation of vendor lock-in risks. Assessing the long-term viability of LC/NC platforms is crucial to mitigate operational risks. Robust community engagement and vendor support are essential for platform adoption and sustainability. Understanding these challenges is imperative for organizations and developers aiming to leverage LC/NC technologies effectively in their software development endeavors (Hecht, 2019).

Conclusion

In conclusion, this thesis has investigated the transformative impact of Low Code and No-Code (LC/NC) platforms within the domain of software development. Drawing from the research of Frank, Maier, and Bock (2021), the discussion has centered on the inherent productivity enhancements offered by conceptual models and the specialized modeling languages designed to facilitate their construction and implementation. These platforms have been shown to democratize the development process, enabling individuals with varying levels of technical expertise to create, adapt, and deploy software efficiently.

Reference models have been identified as a valuable means of streamlining the development process, providing a template that encapsulates commonalities across a class of systems while striving for an improved state of design. However, the development of such models is not without its challenges, including higher costs and the uncertainty of returns on investment, as well as the need for adaptability to meet specific requirements.

The synthesis of domain-specific modeling languages (DSMLs) with reference mod-

els has emerged as a particularly potent approach, affording developers a significant reduction in effort and enhanced model comprehensibility. This synergy is conducive to the reuse and adaptation of models, thereby expediting the development process and reducing the propensity for errors.

Model-Driven Software Development (MDSE) and the Object Management Group's (OMG) Model-Driven Architecture (MDA) have been explored as methodologies that bridge the gap between conceptual abstraction and practical implementation. These approaches underscore the importance of models as primary artifacts in the development process, advocating for automated transformations and a unified representation that caters to both model execution and system modification.

The comparative analysis of LC/NC platforms—including Tilda, Flutterflow, Creatio, WordPress, and Zapier—has provided a comprehensive view of the diverse functionalities, use cases, and adaptability of these platforms. From website construction to complex business process automation, these platforms offer a broad spectrum of capabilities, thus enabling a wide demographic to engage in software development.

Ultimately, the examination of LC/NC platforms within this thesis underscores a pivotal shift in software development paradigms. By reducing the technical barrier to entry, these platforms are facilitating a new wave of innovation and inclusivity in technology creation. This shift is poised to have profound implications for the future of software development, fostering a more collaborative, efficient, and user-centric approach to creating digital solutions. The convergence of these platforms with traditional software development practices heralds a new era of digital transformation, one where the complexities of coding are abstracted, and the focus is shifted to conceptual design and rapid deployment.

REFERENCES

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

Brooke J. (2013). SUS: A retrospective. — *Journal of Usability Studies. — 8*. — 29–40.

Cabot J. (2020). Positioning of the low-code movement within the field of model-driven engineering. In *Proceedings of the 23rd ACM/IEEE International Conference on Model Driven Engineering Languages and Systems: Companion Proceedings*. —Pp. 535–538. IEEE. https://doi.org/10.1145/3417990.3420210

Chang Y.-H. & Ko C.-B. (2017). A study on the design of low-code and no-code platform for mobile application development. *International Journal of Advanced Smart Convergence. — 6*(4). — *Pp.* 50–55. https://doi.org/10.7236/IJASC.2017.6.4.7

Daniel G., Cabot J., Deruelle L. & Derras M. (2020). Xatkit: A multimodal low-code chatbot development framework. *IEEE Access. — 8*. — 15332–15346. https://doi.org/10.1109/ACCESS.2020.2966919

Di Ruscio D., Kolovos D., de Lara J., et al. (2022). Low-code development and model-driven engineering: Two sides of the same coin? *Software and Systems Modeling. — 21*. — *Pp.* 437–446. https://doi.org/10.1007/s10270-021-00970-2

Frank U., Maier P. & Bock A. (2021). Low code platforms: Promises, concepts, and prospects. A comparative study of ten systems. *Universität Duisburg-Essen, Institut für Informatik und Wirtschaftsinformatik (ICB)*. https://doi.org/10.17185/duepublico/75244

Gartner. (2021, February 15). Gartner forecasts worldwide low-code development technologies market to grow 23% in 2021. Retrieved from https://www.gartner.com/en/newsroom/press-releases/2021-02-15-gartner-forecasts-worldwide-low-code-development-technologies-market-to-grow-23-percent-in-2021

Gurcan F. & Taentzer G. (2021). Using Microsoft PowerApps, Mendix and OutSystems in two development scenarios: An experience report. In *2021 ACM/IEEE International Conference on Model Driven Engineering Languages and Systems Companion (MODELS-C)*. — Pp. 67–72. IEEE.

Hakimi E. (2019). Low-code development platform. Retrieved from https://www.slideshare.net/Ehsan-Hakimi/low-code-development-platform

Hecht L.E. (2019). Low-code platform adoption gets a boost from digital transformation. Retrieved from https://thenewstack.io/low-code-platform-adoption-gets-a-boost-from-digital-transformation/

Kalpeyeva Z., Kassymova A., Umarov T., Mustafina A. & Mukazhanov N. (2020). The structure and composition of the business process model. In *Proceedings of the 6th International Conference on Engineering & MIS 2020*. — Article 48. — Pp. 1–6. Association for Computing Machinery. https://doi.org/10.1145/3410352.3410783

Martinez E. & Pfister L. (2023, May 11). Benefits and limitations of using low-code development to support digitalization in the construction industry. — *Automation in Construction*. https://doi.org/10.1016/j.autcon.2023.104073

Pantelimon S., Rogojanu T., Braileanu A., Stanciu V. & Dobre C. (2019). Towards a seamless integration of IoT devices with IoT platforms using a low-code approach. In *2019 IEEE 5th World Forum on Internet of Things*. — Pp. 566–571.

Sahay A., Indamutsa A., Di Ruscio D. & Pierantonio A. (2020). Supporting the understanding and comparison of low-code development platforms. In *Proceedings – 46th Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2020*. — Pp. 171–178. Institute of Electrical and Electronics Engineers Inc.

Sanchis R., García-Perales Ó., Fraile F. & Poler R. (2020). Low-code as an enabler of digital transformation in the manufacturing industry. *Applied Sciences (Switzerland). — 10*(1). — 1–7. https://doi.org/10.3390/app10010012

Schneid K., Stapper L., Thone S. & Kuchen H. (2021). Automated regression tests: A no-code approach for BPMN-based process-driven applications. IEEE.

Shah A. (2020, May 15). Emptying offices prompt adoption of low-code to build work apps. — *The Wall Street Journal*.

Valsamakis Y. & Savidis A. (2020). Smart automations for everybody: When IoT meets visual programming. In P. Davidsson (Ed.), *ACM Digital Library, 10th International Conference on the Internet of Things Companion*. — Pp. 1–4. Association for Computing Machinery. https://doi.org/10.1145/3423423.3423470

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

Wong J., Driver M. & Vincent P. (2019). Low-code development technologies evaluation guide. Retrieved from https://www.gartner.com/en/documents/3902331

ХАЛЫҚАРАЛЫҚ АҚПАРАТТЫҚ ЖӘНЕ КОММУНИКАЦИЯЛЫҚ ТЕХНОЛОГИЯЛАР ЖУРНАЛЫ

МЕЖДУНАРОДНЫЙ ЖУРНАЛ ИНФОРМАЦИОННЫХ И КОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ

INTERNATIONAL JOURNAL OF INFORMATION AND COMMUNICATION TECHNOLOGIES

Правила оформления статьи для публикации в журнале на сайте:
https://journal.iitu.edu.kz
ISSN 2708–2032 (print)
ISSN 2708–2040 (online)
Собственник: АО «Международный университет информационных технологий» (Казахстан, Алматы)

ОТВЕТСТВЕННЫЙ РЕДАКТОР
Мрзабаева Раушан Жалиқызы

КОМПЬЮТЕРНАЯ ВЕРСТКА
Асанова Жадыра

Подписано в печать 15.12.2024.
Формат 60x881/8. Бумага офсетная. Печать - ризограф. 9,0 п.л. Тираж 100
050040 г. Алматы, ул. Манаса 34/1, каб. 709, тел: +7 (727) 244-51-09).

Издание Международного университета информационных технологий
Издательский центр КБТУ, Алматы, ул. Толе би, 59