CLOUD TRANSFORMATION CHALLENGES: DO THEY FAVOR THE EMERGENCE OF LOW-CODE AND NO-CODE PLATFORMS

by

Amit Puri, MS in Data Science

DISSERTATION

Presented to the Swiss School of Business and Management Geneva

In Partial Fulfillment

Of the Requirements

For the Degree

DOCTOR OF BUSINESS ADMINISTRATION

SWISS SCHOOL OF BUSINESS AND MANAGEMENT GENEVA

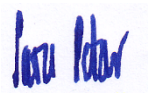
DECEMBER, 2024

CLOUD TRANSFORMATION CHALLENGES: DO THEY FAVOR THE EMERGENCE OF LOW-CODE AND NO-CODE PLATFORMS

by

Amit Puri, MS in Data Science

APPROVED BY



Prof.dr.sc. Saša Petar, Ph.D:, Dissertation chair

RECEIVED/APPROVED BY:

_____

Admissions Director

Dedication

This thesis is dedicated to my family, whose constant encouragement and support have guided me. I am also grateful to my parents for instilling the values of perseverance and hard work and to my spouse for her patience and belief in my abilities.

I also dedicate this work to my mentors, who have inspired me to explore, learn and grow. Without their guidance, this milestone would not have been possible.

Finally, thank you to all the citizen developers and AI industry workers who dare to dream and work tirelessly to make the software industry more accessible to everyone!

Acknowledgements

First and foremost, I am deeply grateful to my advisor, Dr. Monika Singh, whose insightful guidance, expertise, and encouragement have been invaluable throughout my research. I would also like to sincerely thank my thesis committee members.

I am profoundly thankful to my family and friends, whose support and understanding have helped me balance the challenges of research and life. To my parents, my wife Pallavi, my daughter Diksha, my son Ishan and my friends Harmit Singh, Rahul Tandon, and Umesh Radhakrishnan, your insights and belief in my potential have made this achievement possible.

Finally, I acknowledge SSBM and Upgrad, which provided the resources for this research and have been essential in bringing this work to realisation.

ABSTRACT

CLOUD TRANSFORMATION CHALLENGES: DO THEY FAVOR THE EMERGENCE OF LOW-CODE AND NO-CODE PLATFORMS

Amit Puri
2024

Dissertation Chair: Dr. Saša Petar
Co-Chair: Dr. Jaka Vadnjal

This research investigates the challenges associated with cloud transformation and explores whether these challenges create a conducive environment for the emergence of low-code and no-code (LCNC) platforms as viable solutions for digital innovation. The study focuses on cloud-native development strategies, cloud migration models, and the growing role of LCNC platforms in enabling faster application development and deployment.

Methodologically, this research applies both qualitative and quantitative approaches, comparing traditional custom development models with LCNC solutions. Key metrics evaluated include cost efficiency, scalability, deployment control, and business process automation. Additionally, the research delves into the integration of advanced technologies like artificial intelligence (AI) and machine learning (ML) within LCNC platforms, assessing their impact on accelerating digital transformation efforts.

The findings suggest that cloud transformation often faces significant hurdles, particularly in handling complex workloads and legacy systems. These challenges have fueled the rise

of LCNC platforms, which offer a more accessible and efficient alternative for both technical and non-technical users. LCNC platforms have proven to be instrumental in reducing development time and cost, facilitating rapid prototyping, and improving operational efficiency through automation. However, the research also highlights limitations, including customisation constraints and concerns about scalability in large enterprises.

In conclusion, while LCNC platforms present a compelling solution for overcoming cloud transformation challenges, their widespread adoption will require addressing issues related to platform scalability, vendor lock-in, and the integration of open standards. Nonetheless, these platforms are poised to play a pivotal role in future digital transformation initiatives, empowering businesses to innovate and scale rapidly in an increasingly competitive landscape.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER I:

INTRODUCTION

1.1 Introduction

Over the last two decades, significant investments in digital technologies and transformational trends have shaped the Information Technology industry. This journey, culminating in the recent global pandemic, has heightened discussions on digital transformation. Cloud transformation, in particular, has become a critical tool for senior leadership, promising enhanced efficiency and innovation.

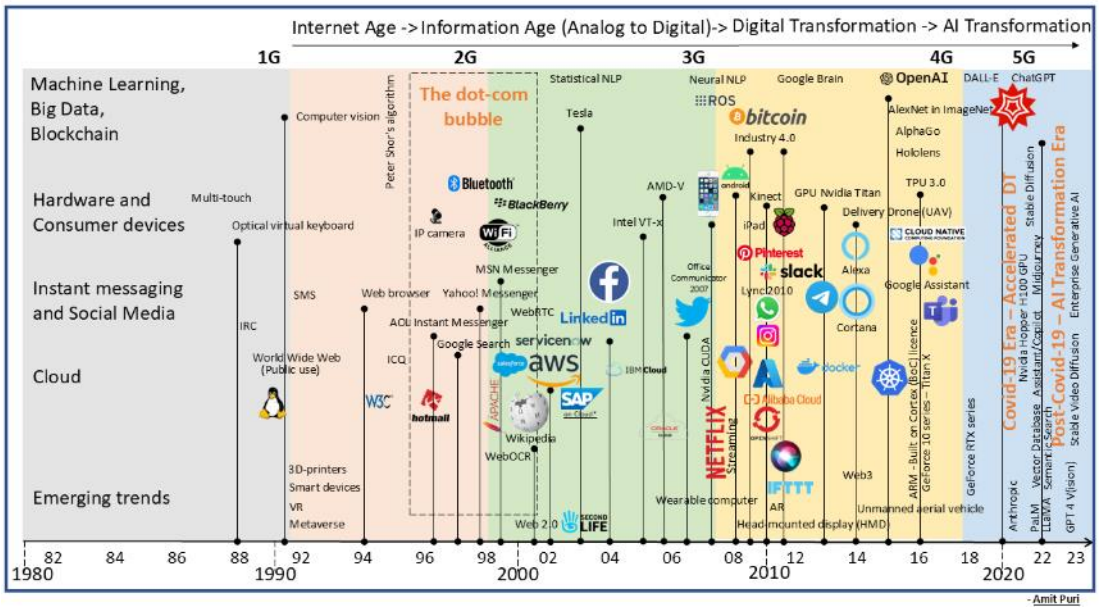


Figure 1.1 The evolution of digital technology

Unlike the Industrial Revolution, the Information Technology Revolution is still relatively young, gaining momentum after the Y2K efforts and the rise of the Internet. Mobile technology, powered by 2G and 3G networks (see Figure 1.1), and the widespread adoption of social media further accelerated this revolution. Businesses rapidly embraced

.com domains to establish an online presence, a trend that gained significant momentum in the early 21st century.

Since the late 2010s, integrating artificial intelligence, Big Data, and Cloud technologies (Mark Kryder, 2005) has accelerated innovation and transformation across industries. This trend continues to drive digitalisation through the 2020s.

By 2023, with 5.3 billion internet users, concerns over digital sovereignty have made data the "new oil" of the digital economy. Individuals now face the challenge of balancing privacy with the benefits of sharing data for personalised services. This dynamic fuels debates around consent, data protection, and ethical usage. As businesses leverage cloud technologies, Big Data, and AI to drive innovation, they add complexity to their software ecosystems. This increased complexity raises risks of insider threats, cybersecurity attacks, and challenges in integrating legacy systems with external platforms.

New business models rooted in the cloud and Big Data further complicate this landscape. They demand more robust security, governance, and regulatory frameworks. Managing these evolving risks requires strong data protection, transparent consent mechanisms, and robust cybersecurity measures to address external and internal threats. Additionally, integrating legacy systems remains a critical challenge as businesses modernise without disrupting operations.

Examining the evolution of digital technologies reveals the intricate nature of this ecosystem, where technical debt accumulates around evolving systems. Organisations must proactively address technical debt through system upgrades and future-proofing their architectures to ensure scalability and operational efficiency, maintaining competitiveness in an increasingly digital world.

As businesses adopt cloud, Big Data, and AI, implementing comprehensive governance frameworks becomes crucial to address security and privacy concerns.

Prioritising strong data protection and ethical usage will build trust and ensure compliance with evolving regulations. Simultaneously, businesses must invest in advanced cybersecurity measures to mitigate threats while modernising legacy systems to support seamless integration. Continuous evaluation of these frameworks will be critical as digital ecosystems grow more complex. Addressing technical debt early through proactive planning will help organisations avoid inefficiencies and ensure scalability as technology evolves.

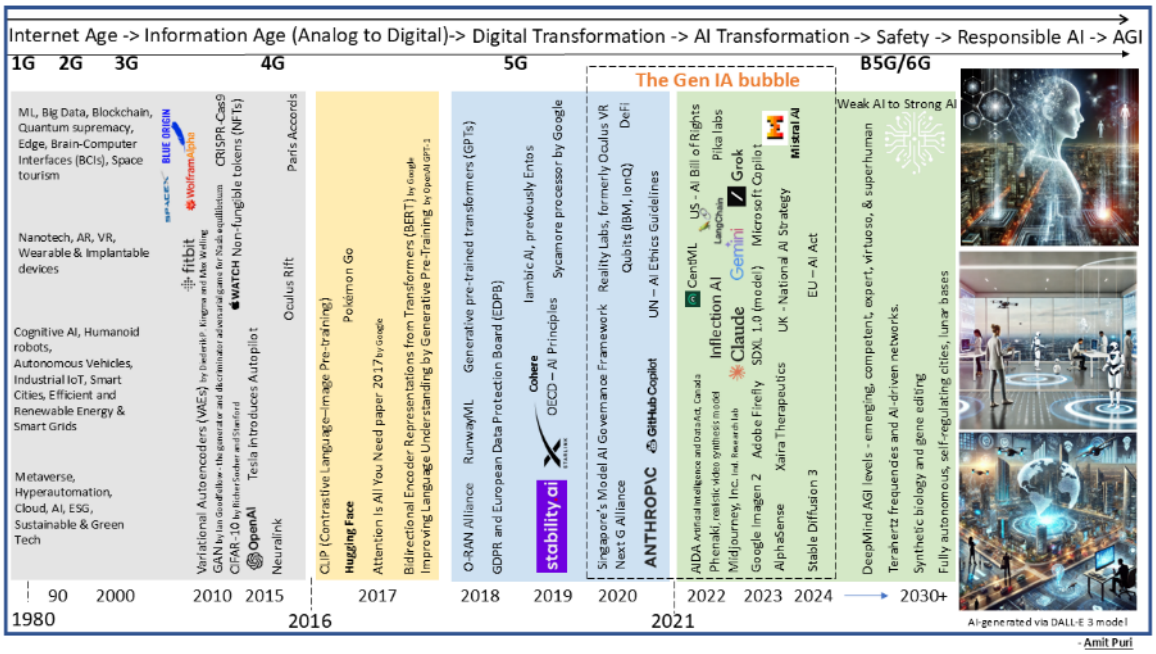


Figure 1.2 Recent trends in Digital technology

Recent trends in digital technology highlight the rapid advancement of Generative AI models, which are revolutionising industries by enabling personalised and intelligent interactions. Models like GPT-4 and Claude have transformed business operations by facilitating natural language processing, image generation, and large-scale automation. This surge in AI-driven solutions has led to the so-called Generative AI bubble, with increased demand across sectors. In parallel, blockchain, quantum computing, and brain-

computer interfaces (BCIs) are making strides, opening possibilities for enhanced human-machine interaction, decentralised systems, and faster, more secure data processing.

The adoption of 5G and the forthcoming 6G networks is unlocking new opportunities for real-time automation, edge computing, and immersive technologies like virtual reality (VR) and augmented reality (AR). These advancements enable faster data processing with lower latency, benefiting industries reliant on IoT devices and AI systems.

Meanwhile, the integration of cloud computing and Big Data continues to reshape enterprise strategies, allowing businesses to analyse vast datasets efficiently while managing the complexity of these systems. As these technologies evolve, businesses face growing concerns over security, privacy, and data sovereignty. Regulatory frameworks such as GDPR and the EU AI Act are becoming critical in ensuring responsible AI usage. Simultaneously, there is a growing emphasis on adopting sustainable technologies under the ESG (Environmental, Social, Governance) framework, encouraging organisations to innovate while staying ethically and environmentally conscious. The convergence of AI governance, responsible tech development, and regulatory compliance is essential for businesses looking to remain competitive in this rapidly evolving digital world.

The expanding AI startup ecosystem significantly transforms the digital landscape through Generative AI (GenAI). As shown in Figure 1.3, startups like Anthropic, OpenAI, and Midjourney are pushing the boundaries of AI capabilities in language models, image generation, and AI-powered decision-making tools. These startups lead the charge in various fields, including speech and text production, automation, personalised experiences, and AI-driven business processes.

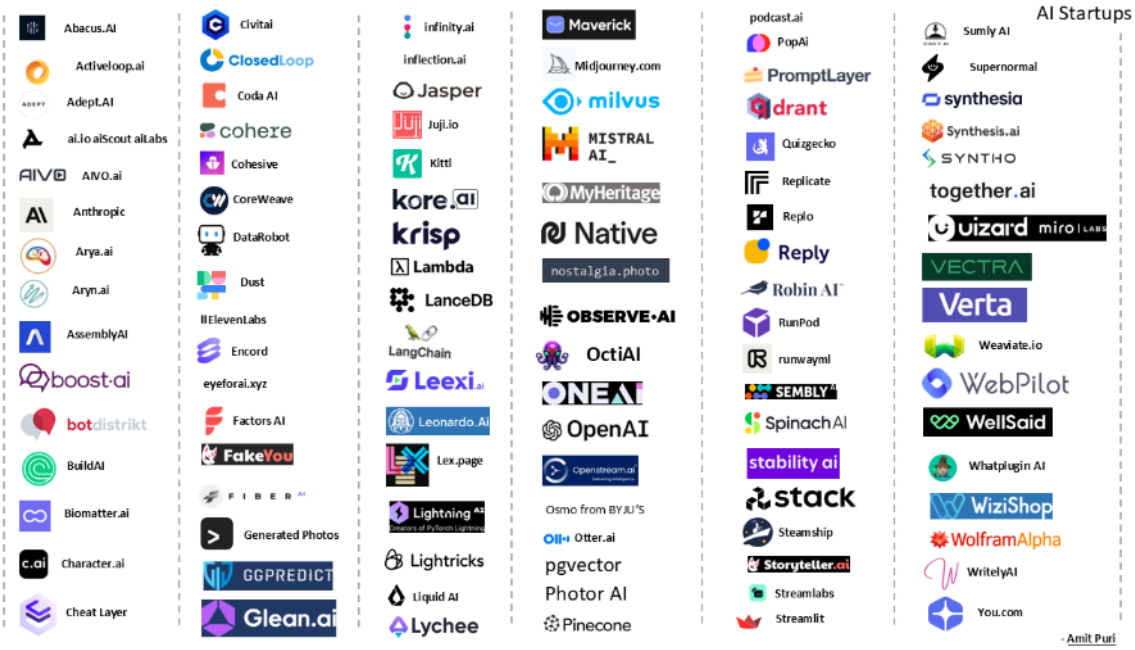


Figure 1.3 AI Startups

Many startups, such as Kore.ai, Sembly.AI, Abacus.ai, and Cohere, are transforming specific use cases with conversational AI. Others, like RunwayML and Leonardo.AI, are advancing creative AI, enabling new artistic expression and design forms. Platforms like Observe.AI and DataRobot are revolutionising business operations by offering AI-powered data analysis, insights, and automation. This ecosystem of startups plays a crucial role in shaping the future of AI, with innovations driving increased efficiency, automation, and new opportunities for human-machine collaboration. As AI evolves, these startups will be at the forefront of its application across industries like healthcare, education, entertainment, and more, creating new business models and disrupting traditional processes.

The rise of Low-code/No-code (LCNC) platforms complements the AI startup boom. Both trends are reshaping the digital landscape by democratising access to advanced technologies.

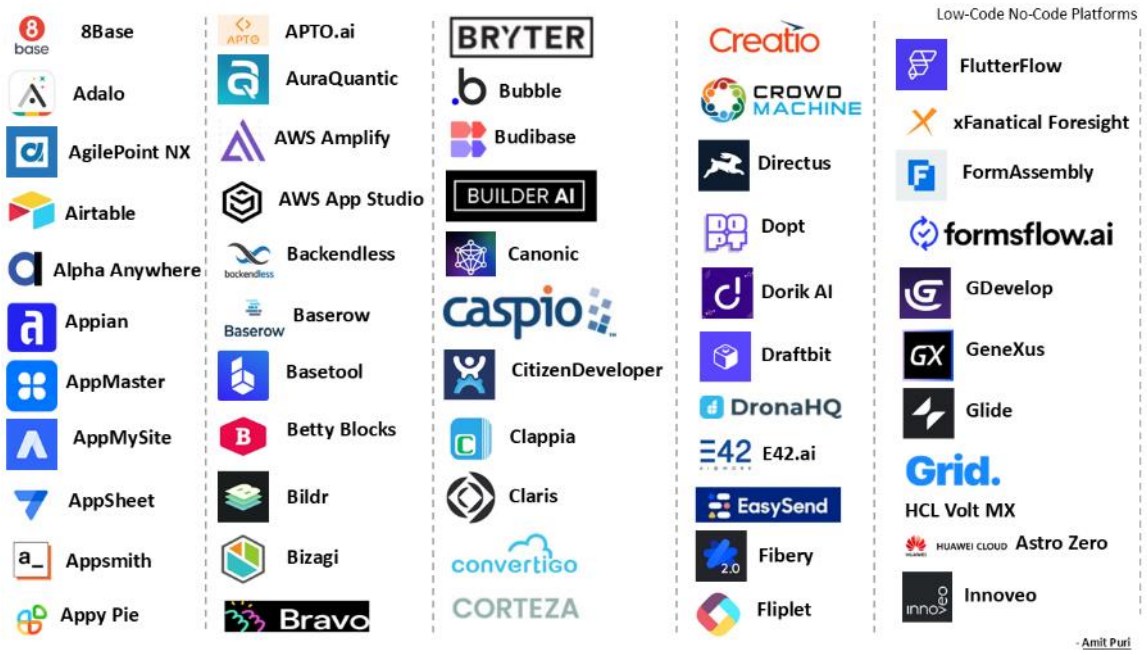


Figure 1.4 Low-code No-code Platforms

While AI startups like OpenAI, Anthropic, and Midjourney lead in AI development, LCNC platforms like OutSystems, Airtable, and Mendix make application development accessible to non-technical users. Together, they enable businesses to innovate rapidly, automate processes, create personalised experiences, and integrate AI-powered solutions with minimal development overhead.

LCNC platforms increasingly incorporate AI functionalities, allowing users to leverage tools like natural language processing, predictive analytics, and machine learning without in-depth technical expertise.

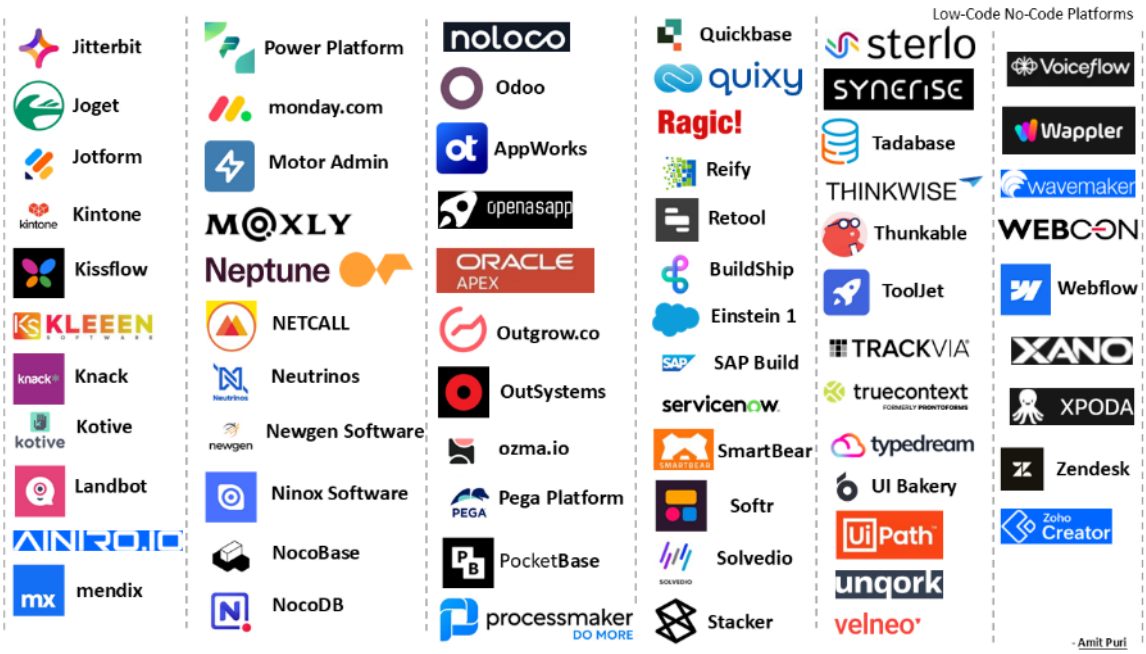


Figure 1.5 Low-code No-code Platforms (continued)

This LCNC and AI technologies convergence empowers businesses to build applications faster and integrate cutting-edge AI capabilities into their workflows, driving operational efficiency and enhancing customer experiences. As both ecosystems expand, AI-powered, easy-to-use development platforms are poised to play a pivotal role in transforming industries, enabling companies to stay agile and competitive in an increasingly digital world.

In this study, we will explore the challenges of cloud transformation and assess whether these challenges create a conducive environment for the rise of low-code/no-code platforms as viable solutions for accelerating digital innovation.

1.2 Research Problem

Recently, several firms have embraced cloud computing by employing a "lift and shift" approach, which involves shifting their on-premise workloads to the cloud with little modifications. Nevertheless, this method frequently results in notable inefficiencies and heightened expenses since it only partially exploits the capabilities inherent to cloud-native systems. These inefficiencies result in subpar performance, heightened operational difficulties, and elevated costs, which undermine the fundamental advantages of cloud computing (Juncal Alonso, 2013). To address these cost concerns, consulting firms suggest implementing the FinOps methodology proposed by Keith Conway in 2023. This approach facilitates the use of precise measures to audit, regulate, and oversee costs more efficiently. However, the question remains: Does this technique facilitate or impose additional intricacies and expenses? There were efforts to rewrite into PaaS or switch to SaaS offerings based on the make/buy decision. In the SaaS disposition strategy, Low-code and no-code platforms have become revolutionary technologies in the software development industry, providing a notable alternative to pre-made solutions and conventional bespoke development. These platforms facilitate expedited application creation by utilising visual programming, drag-and-drop interfaces, and pre-built templates, empowering technical and non-technical users to generate viable applications swiftly. Off-the-shelf solutions give pre-made, standardised software with limited modification and flexibility options. In contrast, low-code and no-code platforms offer a compromise between these pre-made solutions and fully customisable software. They provide higher customisation than pre-made solutions without the significant time and financial investment required for bespoke development. Nevertheless, there are reservations regarding these platforms, including their ability to handle increasing workloads, potential performance limitations, data protection issues, and the risk of being dependent on a single provider. Low-code platforms

present issues regarding integration difficulties and the constraints of customised code, whereas no-code platforms need to be more apprehensive about managing intricate business logic and safeguarding data privacy.

Notwithstanding these concerns, low-code and no-code systems' skill, ease of use, and effectiveness render them appealing options. They achieve a compromise between the convenience of ready-made solutions and a level of customisation typically associated with tailor-made applications.

1.3 Purpose of Research

The sole purpose of this research is to evaluate alternatives to cloud transformation strategies in the realm of SaaS-based solutions, as opposed to IaaS, off-the-shelf solutions, and undesirably complex implementations in PaaS in a few scenarios.

1.4 Significance of the Study

This research holds significant importance as it addresses the crucial role of cloud transformation in the broader context of digital transformation initiatives. By focusing on the efficiency and effectiveness of Low-code No-code (LCNC) platforms, the study contributes to a deeper understanding of how these platforms can reduce operational expenditures (OPEX) and improve decision-making processes. The findings are expected to aid organisations in navigating the complexities of digital transformation, particularly in optimising workloads and reimagining customer and employee experiences through advanced AI-integrated solutions.

Thus, this research highlights the pivotal role of LCNC platforms in streamlining digital transformation and serves as a valuable resource for organisations seeking to leverage these technologies for enhanced operational efficiency and strategic growth.

1.5 Research Purpose and Questions

This research study evaluates no-code and low-code platforms on various parameters such as ease of use, cost of development and deployment, recurring labour and non-labour costs, platform start-up, and development installation. Several parameters on non-functional requirements like data platform support, integration, security, scalability, performance, maintainability, observability, support, upgrade, etc., are compared with other dispositions such as lift-shift, cloud-native development in cloud transformation (Nane Kratzke, 2017) to help the industry use qualitative and quantitative analysis in assisting decision-making exercise on their journey to digital transformation.

As businesses increasingly seek cost-effective and efficient solutions for cloud computing, exploring innovative approaches that can optimise cloud workloads and rationalisation becomes imperative. In this context, low-code and no-code platforms have emerged as potential game-changers. This research investigates the viability, specific use cases, attributes, and compatibility of low-code and no-code platforms in optimising cloud services. The following research questions and hypotheses are proposed to guide this investigation:

RQ1: Are low-code and no-code platforms a viable alternative to rationalising cloud workloads for reducing costs on cloud services?

Hypothesis 1: Low-code and no-code platforms significantly reduce the cost of cloud services by simplifying and accelerating the development process, thus offering a viable alternative for rationalising cloud workloads.

RQ2: What are the possible scenarios, specific use cases, or application types in which organisations looking to optimise their cloud solutions might consider low-code and no-code platforms to achieve a quick turnaround?

Hypothesis 2: Low-code and no-code platforms are particularly effective in scenarios where rapid development, deployment, and scalability are critical, such as data analytics, customer relationship management, and automated workflows.

RQ3: What attributes of low-code and no-code platforms should be considered when assessing workloads against these platforms' features for investing in modernisation or rationalisation efforts?

Hypothesis 3: Key attributes like scalability, security, integration capabilities, and user-friendliness of low-code and no-code platforms are crucial in determining their suitability for various workloads in cloud rationalisation efforts.

RQ4: How are low-code and no-code platforms different from SaaS offerings, and how does integration with a SaaS platform affect the choice of low-code and no-code platforms?

Hypothesis 4: Low-code and no-code platforms offer greater customisation and flexibility than SaaS offerings, making them more suitable for businesses requiring bespoke solutions. Integration with SaaS platforms enhances these platforms' functionality and broadens their applicability.

RQ5: How does support for open standards and integration with other cloud models, such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Function as a Service (FaaS) — often associated with serverless computing — and Container as a Service (CaaS), affect the choice of low-code and no-code platforms?

Hypothesis 5: The effectiveness of low-code and no-code platforms is enhanced by their interoperability and integration with various cloud models, such as IaaS, PaaS, FaaS, and CaaS, providing a comprehensive solution for businesses seeking to optimise their cloud infrastructure.

In conclusion, a low-code or no-code platform can be significantly influenced by its support for open standards and its ability to integrate with various cloud models. This integration can affect everything from application flexibility and scalability to cost management, innovation potential, and the ability to meet security and compliance requirements.

CHAPTER II:

REVIEW OF LITERATURE

2.1 Background

Over the past decade, cloud transformation has become a crucial precursor to digital transformation for many organisations. The goal is to optimise their workload by effectively planning, implementing, and evaluating Digital Transformation initiatives to reduce CAPEX (Capital Expenditures) costs to OPEX (Operating Expenditures). While this shift provides many advantages, it presents unique challenges, mainly when dealing with specific workloads' complexity and legacy nature.

This comprehensive study explores cloud transformation approaches, challenges, and shortcomings in cloud transformation efforts. It investigates how Low-Code No-Code platforms (LCNCs), also referred to as Low-Code Development Platforms (LCDPs), No-Low Development Platforms (NCDPs), can enhance the efficiency and effectiveness of business processes and workflow transformation, achieving these improvements in a quicker turnaround. Integrating AI with LCNC platforms, using AI-augmented development techniques, enhances digital transformation by automating and optimising the application development process, promising a future of greater automation, increasingly sophisticated applications with actionable insights, and reimaging employees, customers, and partner experiences. Recent breakthroughs in pre-trained large language models (LLMs) and the Transformer model architecture are anticipated to fuel Low-Code and No-Code platforms with the capability of natural language processing tasks, extending their reach to a broader audience, and unlocking the expanded value of data, visualisation, and insights, which is the lifeblood of digital transformation, thereby enabling more efficient operations, improved decision-making, and more competitive strategy formulation. The next wave of AI-assisted digital transformation can substantially improve productivity,

establish new business models, and empower organisations, individuals, and AI agents to be proactive, intelligent, and learn from action/feedback mechanisms, thus promoting value or goal-driven and customer-centric business practices.

The literature review attempts to critically assess existing studies on cloud transformation, which is the strategic integration of digital technology into all aspects of a business, radically transforming how organisations operate and produce value, including a cultural and operational overhaul. Many organisations have successfully managed this shift despite hurdles such as opposition to change and cybersecurity concerns. This chapter also delves into the literature on LCNC platforms, which have emerged as game changers in this path, democratising software creation (Citizen development) by allowing developers and non-technical users to construct programs via graphical interfaces. While these platforms have limits, such as potential customisation, scalability concerns, and vendor lock-in, their widespread use across industries highlights their critical role in promoting digital transformation. Another major changer, artificial intelligence (AI), can automate commercial processes, deliver intelligent data analyses, and generate new services despite ethical and security concerns.

2.2 Cloud Transformation Challenges

As mentioned earlier, Cloud transformation or adoption is a fundamental component of numerous digital transformation initiatives. Cloud computing enables organisations to access computing resources on demand globally, with elasticity, without investing in costly hardware or software. This can reduce costs and improve serviceability, flexibility, and scalability. In addition, the cloud provides organisations access to numerous advanced technologies, such as machine learning, blockchain, analytics, and quantum computing, that can facilitate digital transformation. Like any transformation, it presents

hurdles and new opportunities for organisations aspiring to undergo it, as discussed in most research papers.

- Lack of digital skills and capabilities: Transformation requires organisations to develop new technical skills and capabilities, such as data analytics, artificial intelligence, and blockchain. However, many organisations lack the skills and expertise to implement Transformation effectively.
- Resistance to change: Transformation involves significant changes to organisational structures, processes, and cultures. However, many employees and stakeholders may resist these changes, leading to resistance and inertia.
- Uncertainty and complexity: Transformation is a complex and uncertain process that involves multiple stakeholders, technologies, and environments. This can make it challenging to effectively plan, implement, and evaluate Transformation initiatives.
- Integration and interoperability: Transformation requires organisations to integrate digital technologies into their existing systems and processes and to ensure interoperability between different systems and platforms. However, this can be challenging due to technical, organisational, and cultural barriers.
- Cybersecurity and privacy: Transformation involves the collection, storage, and processing of large amounts of data, which can pose significant cybersecurity and privacy risks. Organisations must ensure that their digital systems and processes are secure and compliant with relevant regulations and standards.
- Digital divide: Transformation can exacerbate existing social and economic inequalities, as some individuals and communities may lack access to digital technologies and skills. Organisations must ensure that their Transformation initiatives are inclusive and equitable and do not widen the digital divide.

The challenges and factors that are covered in this research paper (Xiaoteng Zhu, 2021):

- Digital business strategy: The challenge of digital business strategy is to align IT and business strategies to create and capture value in the digital era. This requires organisations to rethink their business models, products, and services considering digital technologies. The outcome of a digital business strategy is to develop new business models, products, and services that leverage digital technologies. For example, companies can use digital platforms to connect with customers and suppliers or use data analytics to personalise products and services.
- Strategic action field: The challenge of the strategic action field is to understand the social and institutional aspects of Digital Transformation, such as power relations, norms, and values. This requires organisations to navigate complex social and political environments and engage with stakeholders meaningfully. The outcome of the strategic action field is to develop new governance structures, policies, and regulations that support Digital Transformation. For example, governments can encourage companies to invest in digital technologies or establish data privacy and security standards.
- Digital technology: The challenge of digital technology is to leverage digital technologies, such as big data, artificial intelligence, and blockchain, to create value. This requires organisations to develop new technical skills and capabilities and to integrate digital technologies into their existing systems and processes. The outcome of digital technology is to create new applications, platforms, and ecosystems that enhance efficiency, innovation, and customer experience. For example, companies can use data analytics to optimise supply chain operations or blockchain to create secure and transparent transactions.

- Agile digital transformation: The challenge of agile digital transformation is to adopt an agile and iterative approach to Digital Transformation, which emphasises flexibility, speed, and customer-centricity. This requires organisations to embrace change and uncertainty and to involve customers and employees in the Digital Transformation process. The outcome of agile digital transformation is to reduce time-to-market, increase responsiveness, and improve customer satisfaction. For example, companies can use agile methodologies to develop and test new products and services quickly or use design thinking to co-create customer solutions.
- Digital enterprise architecture: The challenge of digital enterprise architecture is to design and implement digital architectures that support Digital Transformation. This requires organisations to develop a holistic view of their digital systems and processes and to align them with their business goals and strategies. The outcome of digital enterprise architecture is to enhance interoperability and scalability.

2.3 Cloud Transformation Approaches and Tools

Organisations can incorporate cloud technologies into their operations through various adoption strategies. A common approach is to use a hybrid cloud model, in which mission-critical workloads are executed on-premises, and cloud-ready workloads are executed in the cloud. Another approach is the multi-cloud model, where an organisation utilises multiple cloud service providers to meet its varied requirements. This can offer organisations greater flexibility and resilience, but it can also be more challenging to manage. In what is known as a cloud-native model, some organisations migrate all their workloads to the cloud. This strategy may provide the most significant benefits regarding scalability and flexibility, but its implementation may be the most challenging and time-consuming.

The paper (Mahdi Fahmideh, 2016) identifies several migration strategies that can be used to migrate applications to the cloud. Here are the strategies, along with a detailed explanation of each:

- Lift-and-shift involves moving an application from an on-premises environment to the cloud without significantly changing the application architecture. The application is typically moved to a virtual machine in the cloud, which provides the same operating system and hardware environment as the on-premises environment. This strategy is relatively simple and can be completed quickly, but it may not take full advantage of the benefits of the cloud.
- Re-architecture involves redesigning the application architecture to take advantage of cloud-specific features, such as auto-scaling, load balancing, and serverless computing. This strategy can result in significant cost savings and performance improvements, but it requires more time and effort than the lift-and-shift strategy.
- Hybrid combines on-premises and cloud resources to support an application. For example, the application may use on-premises resources for sensitive data and cloud resources for non-sensitive data. This strategy can provide the benefits of both on-premises and cloud environments, but it requires careful planning and management.
- Repurchase involves replacing an existing application with a cloud-based alternative. For example, an organisation may replace an on-premises email server with a cloud-based email service. This strategy can provide significant cost savings and performance improvements but may require significant organisational processes and workflow changes.
- Retire: This strategy involves decommissioning an application that is no longer needed. For example, an organisation may retire an application that a newer application has replaced. This strategy can reduce costs and simplify the organisation's IT

environment, but it requires careful planning and management to ensure that critical data is not lost.

The choice of migration strategy depends on the organisation's goals, resources, and constraints. The paper suggests that organisations should carefully evaluate each strategy and choose the one that best meets their needs. Building on this perspective, a paper on the Cloud migration process (Mahdi Fahmideh, 2016) proposed the evaluation framework with 28 criteria, classified into two dimensions:

- Generic Dimension

Process clarity, Procedure and supportive techniques, ability to tailor solutions, Development roles, Modelling language, Traceability, work products, Formality, Scalability, Tool support, and Domain applicability.

- Cloud-Specific Dimension

Elasticity, Multi-tenancy, Cloud architecture model definition, Cloud service model definition, Cloud deployment model definition, Cloud vendor lock-in, Cloud interoperability, Cloud security and privacy, Cloud performance, Cloud monitoring and management, Cloud data management, Cloud disaster recovery, Cloud compliance, Cloud cost, Cloud migration strategy, Cloud migration tool, and service.

The criteria were derived through an extensive literature review and validated through a web-based questionnaire survey of 104 experts from academia and practitioners in the cloud computing field. This paper focuses on the strategic task of migrating mission-oriented enterprise applications to cloud environments. It reviews and evaluates existing cloud migration methods from the perspective of process models. An evaluation framework is introduced to analyze these methods, emphasizing their characteristics, similarities, and differences. The study provides a comprehensive overview of current cloud migration research, compiling essential activities, suggestions, techniques, and

concerns related to the migration process. This consolidated view aids both researchers and practitioners in understanding the cloud migration process holistically. The paper also points out unaddressed challenges in current methods, suggesting areas for future research.

The research paper (Chee, Zhou, Meng, Bagheri, & Zhong, 2011) describes a pattern-based approach to cloud transformation, which leverages a knowledge base of enablement patterns and cloud platform information captured in a structured form. The paper presents a mathematical model to select an optimal solution for a given cloud transformation problem based on feature information common to an application profile, a set of enablement patterns, and a set of cloud platforms. The paper also describes a Cloud Transformation Advisor tool that uses this approach to assist an architect in solving a transformation problem. The paper's key findings include the importance of a structured knowledge base of enablement patterns, the use of a mathematical model to select an optimal solution, and the potential for additional applications of this approach beyond cloud transformation. The Cloud Transformation Advisor (CTA) tool assists architects or consultants in determining the best way to transform an application to the cloud. It uses a structured knowledge base of enablement patterns and cloud platform information to facilitate pattern selection. The CTA takes an application profile, a set of enablement patterns, and a set of cloud platforms as input and then uses a mathematical model to select the optimal solution for a given transformation problem. The CTA leverages a taxonomy of enablement patterns and reusable solutions to common issues encountered during cloud transformation. The taxonomy is organised into categories such as data management, security, and performance, and each pattern is described in terms of its purpose, benefits, and implementation details. To use the CTA, an architect or consultant first inputs information about the application to be transformed, such as its architecture, data requirements, and performance characteristics. The CTA then uses this information to

identify relevant enablement patterns and cloud platforms and applies the mathematical model to select the best combination of patterns and platforms for the given transformation problem. The output of the CTA is a set of recommendations for transforming the application to the cloud, including which enablement patterns to use and which cloud platforms to target. Overall, the CTA provides a structured and systematic approach to cloud transformation, helping to ensure that the transformation process is efficient, effective, and tailored to the specific needs of the application being transformed.

In summary, organisations have multiple strategies when migrating their operations to the cloud—from hybrid models combining on-premises and cloud resources to full cloud-native approaches. The choice of strategy is influenced by an organisation's specific goals, resources, and constraints. Mahdi Fahmideh's 2016 paper offers an in-depth exploration of various migration strategies, presenting an evaluation framework with 28 criteria spanning generic and cloud-specific dimensions. This framework was validated through extensive literature reviews and expert surveys to provide a holistic understanding of the cloud migration process. Research by Chee, Zhou, Meng, Bagheri, & Zhong in 2011 introduces a pattern-based approach to cloud transformation, emphasising the importance of structured knowledge bases and mathematical models for optimal solution selection. Their Cloud Transformation Advisor tool exemplifies this approach, offering architects a systematic method to determine the best cloud migration strategy for specific applications. These studies underscore the complexity of cloud migration and the need for well-informed, strategic decision-making in the process. Like CTA, cloud vendors provide a cloud adoption framework, migration services for business case analysis, migration plan creation, security and compliance review, and pricing calculators to help in modernisation efforts.

2.4 Modernization and Cloud-native

This paper (Bamigbala & Safonova, 2022) focuses on Amazon Web Services (AWS) and how it has transformed the traditional software operational model. The paper examines Cloud-native and serverless computing and how they have supported the software engineering teams in delivering scalable and agile modern applications. The paper also discusses the benefits of the cloud model, such as scalability, cost savings, and business agility, as well as the challenges of security and vendor lock-in.

The paper highlights the impact of Amazon on business and society, as many famous companies are using the AWS platform, including Netflix, Unilever, Adobe, and Canon. The paper also discusses how Amazon has changed its corporate culture and values, operating and delivery model, organisational structure, power distribution and control system, roles, skills and career path, compensation and incentives, and leadership to implement and test modern technologies and create new projects.

Overall, the paper suggests that the future of cloud services remains strong as businesses will continue to embrace the cloud service model in the years to come to remain competitive. The paper advises needing a generally acceptable industry standard to remove interoperability bottlenecks. The research opines that the future of the AWS cloud platform is promising, and the acceptance level will continue to increase as enterprises continue to leverage the benefits; however, continuous innovation is needed to sustain and keep the vendor in the technology market.

Originally designed by Google, Kubernetes has emerged as the de facto standard for orchestrating containerised applications in the cloud-native landscape. Its robust capabilities support the deployment and scaling of microservices, making it an integral part of modern software architectures.

The research paper Kubernetes as an Availability Manager for Microservice Applications (Vayghan, Saied, Toeroe, & Khendek, 2019) delves into the evolving landscape of microservice architectures. These architectures, characterised by their modular and independent components, are becoming the backbone of cloud-native applications. Nevertheless, their availability remains a challenge, especially for carrier-grade service providers. Enter Kubernetes, an open-source platform adept at orchestrating these microservices. It simplifies the intricacies of deployment, scaling, and maintenance, ensuring the microservices remain available. The authors had previously studied Kubernetes in a confined private cloud environment, examining its default settings and availability implications. This paper, however, offers a more comprehensive evaluation. It presents suitable architectures for diverse cloud settings, assesses Kubernetes' self-healing capabilities, investigates redundancy's role in enhancing availability, and conducts rigorous experiments on Kubernetes configurations. A notable comparison is also made with the Availability Management Framework (AMF), renowned for its high availability management. A pivotal revelation from the study is that Kubernetes-managed applications might experience significant service outages under certain conditions. This enlightening paper can be found in the Journal of Network and Computer Applications, and those seeking a deeper grasp on the subject are encouraged to peruse the complete study and related literature.

Kubernetes, often called K8s, is an open-source container orchestration platform. Due to its popularity and utility, various cloud providers and organisations have developed and managed Kubernetes services and distributions. Here is a breakdown of some of the most notable Kubernetes variants and cloud services:

Managed Kubernetes Services:

- Google Kubernetes Engine (GKE): Offered by Google Cloud Platform, GKE provides a managed environment for deploying, managing, and scaling containerised applications using Google infrastructure.
- Amazon Elastic Kubernetes Service (EKS): Amazon's managed Kubernetes service on AWS allows users to run Kubernetes without managing the underlying infrastructure.
- Azure Kubernetes Service (AKS): Microsoft's managed Kubernetes service on Azure simplifies deploying, managing, and scaling containerised applications using Azure infrastructure.
- IBM Cloud Kubernetes Service: A managed Kubernetes service on IBM Cloud designed to simplify deploying and managing Kubernetes clusters.
- DigitalOcean Kubernetes (DOKS): DigitalOcean's managed Kubernetes service simplifies the deployment, management, and scaling of applications.
- Oracle Container Engine for Kubernetes (OKE): Oracle's managed Kubernetes service that offers automated upgrades, scaling, and reliability.

Kubernetes Distributions:

- OpenShift: Developed by Red Hat, OpenShift is an enterprise Kubernetes platform that provides developer and operational tools.
- Rancher: An open-source platform that provides additional features on top of Kubernetes, making it easier to deploy and manage Kubernetes at scale.
- Tanzu Kubernetes Grid: Offered by VMware, Tanzu is an enterprise-ready Kubernetes runtime that streamlines operations across multi-cloud infrastructure.
- K3s: Developed by Rancher Labs, K3s is a lightweight Kubernetes distribution designed for edge and IoT use cases.

- MicroK8s: Developed by Canonical, the company behind Ubuntu, MicroK8s is a lightweight, pure upstream Kubernetes installation designed for developers and edge environments.

Specialized Kubernetes Solutions:

- Anthos: Developed by Google Cloud, Anthos is a modern application management platform providing consistent development and operations experience for both on-premises and cloud environments.
- Azure Arc: Microsoft's solution allows users to manage Kubernetes clusters across on-premises, multi-cloud, and edge environments.

Other Notable Offerings:

- Alibaba Cloud Container Service for Kubernetes: Alibaba Cloud's managed Kubernetes service.
- Huawei Cloud Container Engine: Huawei's managed Kubernetes service.

The Kubernetes ecosystem is extensive and constantly changing, so this list does not cover everything. Numerous other niche providers and open-source initiatives provide services and tools related to Kubernetes. When considering using Kubernetes, it is crucial to assess each option according to your unique needs and infrastructure demands.

Challenges with Kubernetes

This comprehensive study (Shamim, Gibson, Morrison, & Rahman, 2022) delves deep into the world of Kubernetes, a pivotal open-source software that has revolutionised the automated deployment and orchestration of containers. Renowned IT giants such as IBM,

Pinterest, and Spotify have significantly leveraged Kubernetes to enhance their software release frequency. The primary objective of this paper is to provide a holistic view of the advantages and pitfalls of Kubernetes, aiming to guide both industry professionals and academic researchers. To achieve this, the authors embarked on an extensive multi-vocal literature review. They meticulously analyzed a vast array of 321 Kubernetes-centric online resources to capture the real-world benefits and challenges experienced by practitioners. Furthermore, a thorough examination of 105 scholarly articles was conducted to gauge the current research trends surrounding Kubernetes. The study unveiled eight notable benefits of Kubernetes, emphasising its self-healing containers and SLO-based scalability features. However, it highlighted 15 distinct challenges, underscoring the pressing need for enhanced diagnostic and security tools. Interestingly, the research community's focus on 14 pivotal topics related to Kubernetes was highlighted, with efficient resource utilisation standing out. The paper also underscored nine challenges that, despite their significance, remain underrepresented in academic literature. These encompass cultural adaptation, hardware compatibility, and the inherent learning curve associated with Kubernetes.

The research paper identifies the following 15 challenges related to Kubernetes:

- Unavailability of diagnostic tools.
- Lack of security tools.
- Attack surface reduction.
- Cultural change.
- Hardware compatibility.
- Learning curve.
- Maintenance.
- Testing.
- Service level objective (SLO) management.

- Scalability concerns.
- Managing self-healing containers.
- Integration with existing systems.
- Resource optimization.
- Networking and service discovery.
- Data persistence and management for stateful applications.

(Note: The last six challenges are inferred based on common Kubernetes challenges)

Migrating apps to Kubernetes-based platforms presents several challenges:

- Complexity of Configuration: Kubernetes has a steep learning curve, and setting up application configurations can be intricate.
- Stateful Applications: Managing and migrating stateful applications, like databases, requires careful planning to ensure data persistence and integrity.
- Networking Concerns: Setting up networking, load balancing, and service discovery in Kubernetes can differ from traditional setups, necessitating a re-evaluation of existing network configurations.
- Security Implications: Ensuring security in a Kubernetes environment involves configuring role-based access controls, network policies, and secrets management, which can differ from previous security models.
- Dependency Management: Applications with specific dependencies might face compatibility issues or require additional configurations to run smoothly in a containerized environment.
- Resource Optimization: Ensuring efficient resource utilization without over-provisioning or under-provisioning can be challenging, especially when migrating legacy applications not designed for containerization.

- Monitoring and Logging: Traditional monitoring and logging tools might not be directly compatible with Kubernetes, requiring adopting new tools or adapting existing ones.
- Integration with Existing Systems: Integrating Kubernetes with existing CI/CD pipelines, storage solutions, and other tools might require significant modifications.
- Downtime Concerns: While migrating, there might be concerns about application downtime, which requires strategies for blue-green deployments or canary releases.
- Skill Gap: Organizations might face a skill gap, as Kubernetes requires expertise that might not be present in teams familiar with traditional deployment methods.

It is imperative to mention the Cloud Native Computing Foundation (CNCF), accessible at https://www.cncf.io. This influential entity plays a pivotal role in the modern software landscape. Established under the umbrella of the Linux Foundation, CNCF is dedicated to fostering and advancing the adoption of cloud-native technologies. Cloud-native approaches prioritise scalable, resilient, and flexible applications designed to run in dynamic, containerised environments like Kubernetes. The CNCF is a neutral home for key projects, including Kubernetes, Prometheus, and Envoy, ensuring their development and growth are collaborative and community-driven. By providing many resources, tools, and community events, CNCF supports the ecosystem's technical evolution and cultivates a vibrant community of developers, end-users, and stakeholders. Through its efforts, the foundation aims to shape the future of software development, making cloud-native approaches the new norm for application deployment and management.

Modernisation and cloud-native approaches represent the forefront of technological evolution, offering unparalleled scalability, resilience, and flexibility in software deployment. As businesses strive to stay competitive in a rapidly changing digital landscape, embracing these paradigms is no longer optional but essential. By adopting

cloud-native methodologies, organizations position themselves for sustainable growth, ensuring agility and adaptability in the face of future challenges. Cloud-native adoption can be hindered by steep learning curves, legacy system entanglement, and initial resource investments.

McKinsey's article "Cloud Adoption to Accelerate IT Modernization" (Bommadevara, Migilo, & Jansen, 2018) discusses the increasing adoption of cloud computing and the challenges companies face in fully leveraging the benefits of cloud technology. The authors argue that simply moving IT systems to the cloud, a “lift-and-shift” process, does not automatically yield the benefits cloud infrastructure and systems can provide. This strategy may sometimes result in more complicated, time-consuming, and expensive IT infrastructures than before. According to McKinsey's article, the whole advantage of cloud technology comes from incorporating it within a complete strategy to achieve digital transformation. This strategy is enabled by the standardization and automation of the IT environment through an OpenAPI architecture, adopting a modern security posture, deploying an automated agile operating model, and exploiting new capabilities to create innovative business solutions. According to the authors, the whole advantage of cloud technology comes from incorporating it within a complete strategy to achieve digital transformation. While the cloud is not required for any of these characteristics, it does operate as a force multiplier.

McKinsey's article also emphasises conventional organisations’ difficulties while migrating to the cloud. Existing business applications built with the old IT paradigm are often monolithic and designed for fixed/static capacity in a few data centers. Simply shifting them to the cloud will not provide them with all their dynamic characteristics. Furthermore, the average corporate IT staff is well-educated in designing business applications in the old IT framework, and most of them will need to be reskilled or

upskilled for the cloud environment. The article suggests that to embrace cloud technology fully, companies need to make substantial upfront investments in what is often a multiyear journey. Long-term commitment and a clear mandate from the CEO and board are required. Companies should focus on four essential themes for effective cloud adoption at scale: sourcing, building a public-cloud operating model, legacy-application remediation, and nurturing the proper capabilities. Finally, McKinsey's article argues that by using cloud computing as a foundation for IT automation, businesses may achieve the following benefits: scalability, agility, flexibility, efficiency, and cost savings. However, doing so necessitates the development of both automation and cloud capabilities.

Application modernization is a crucial component of many cloud adoption initiatives. Organisations transitioning to the cloud frequently must modernize their existing applications to make them more scalable, resilient, and agile. This may involve refactoring an application's code, migrating it to a cloud-native platform, and incorporating new technologies, such as machine learning and analytics.

Organisations should carefully evaluate their needs, priorities, and the costs over benefits of various cloud adoption options to determine the best action. This may involve comprehensively analysing the organisation's current applications and workloads and identifying candidates for cloud migration. It may also involve comparing various cloud service providers' capabilities and determining which best suits the organisation's requirements. Additionally, organisations consider their long-term objectives and strategies and how cloud adoption fits into those plans. By considering all these factors, organisations arrive at a suitable cloud adoption strategy tailored to their specific needs and objectives.

In application modernisation, organisations adopt many distinct approaches. One approach is to refactor the code of an existing application without modifying its underlying architecture. This can make the application more efficient and scalable and enhance its performance. An alternative strategy is to migrate the application to a cloud-native platform, such as a container-based platform like Kubernetes. This can make the application more portable and easier to manage, allowing it to use cloud features like automatic scaling and failover. Lastly, some organisations reconstruct their applications from scratch utilising modern technologies and architecture. This can provide the most significant benefits in flexibility and scalability, but it can also be the most time-consuming and costly strategy.

While the most prevalent method is "lift and shift," in which the existing application is migrated to the cloud without significant changes to its architecture or source code. This can be a quick and straightforward way to get an application running in the cloud, but it may only provide some of the benefits of cloud computing.

The container-based virtualisation technology is bringing about significant changes in how organisations manage their IT infrastructure. Here is how:

Scalability and Flexibility: Container-based virtualization allows for easy scalability. Organisations can quickly scale up or down based on their needs without extensive infrastructure changes. This is particularly useful for adapting to rapidly changing IT demands.

Cost-Efficiency: Traditional IT infrastructure often involves significant overhead costs in hardware and maintenance. Containers, being lightweight, can run multiple applications on the same hardware without needing full-fledged operating systems for each, leading to cost savings.

Rapid Deployment: Containers package applications with all their dependencies, ensuring consistent environments. This means applications can be quickly deployed, moved, or replicated across different stages of development or even different machines, reducing the time-to-market.

Isolation and Security: Containers provide isolation between applications. This means that if one application fails or has a security issue, it does not necessarily affect others. This isolation enhances application security and reliability.

Efficient Resource Utilization: Containers efficiently use system resources. They share the same OS kernel and isolate the application processes from each other. This leads to faster start-up times and better utilisation of underlying resources than traditional virtual machines.

Portability: Since containers encapsulate all dependencies, they can be moved seamlessly across a developer's local machine, test environments, and cloud providers. This portability ensures consistency and reduces the "it works on my machine" problem.

Integration with Modern Development Practices: Containers fit well with modern development practices like continuous integration and deployment (CI/CD). They can be easily integrated into DevOps workflows, leading to faster and more reliable software releases.

Enhanced Monitoring and Management: As the article mentions, with tools like Datadog, companies can monitor the performance of their containerised applications in real-time, detect anomalies, and ensure optimal performance.

In summary, container-based virtualisation technology revolutionises organisations' IT infrastructure by offering flexibility, cost-efficiency, and enhanced performance. These advantages make it a compelling choice over traditional methods, especially in a dynamic and competitive environment like manufacturing.

The paper (Deng, et al., 2023) decouples the life cycle of cloud-native applications into four states: building, orchestration, operate, and maintenance.

Building: This stage involves developing cloud-native applications, including selecting appropriate programming languages, frameworks, and tools. Critical problems in this state include containerisation, microservices architecture, and DevOps practices. Key performance metrics in this state include infrastructure, platform, and software metrics.

Orchestration: This state involves deploying and managing cloud-native applications, including container orchestration platforms like Kubernetes. Critical problems in this state include service discovery, load balancing, and auto-scaling. Key performance metrics in this state include availability, scalability, and fault tolerance.

Operate: This state involves operating and monitoring cloud-native applications, including logging, monitoring, and tracing tools. Critical problems in this state include observability, security, and compliance. Key performance metrics in this state include response time, throughput, and error rate.

Maintenance: This state involves constantly evolving cloud-native applications, including continuous integration and continuous delivery (CI/CD) pipelines. Critical problems in this state include versioning, testing, and release management. Key performance metrics in this state include maintainability, testability, and deployability.

The paper highlights the critical implications and limitations of existing works in each state. For example, in the building state, the paper notes that containerisation and microservices architecture can improve scalability and agility and increase complexity and management overhead. In the orchestration state, the paper notes that while Kubernetes is a popular container orchestration platform, it can be challenging to configure and manage. In the operating state, the paper notes that while logging, monitoring, and tracing tools can improve observability, they can also introduce performance overhead. The paper notes that

CI/CD pipelines can improve release velocity, introduce risk, and require careful management in the maintenance state.

The paper discusses the challenges, future directions, and research opportunities in cloud-native computing. For example, the paper notes that future research could focus on improving the security and compliance of cloud-native applications, developing new tools and techniques for observability and monitoring, and exploring new approaches to testing and release management. The paper also notes that future research could explore emerging technologies such as serverless computing and edge computing in the context of cloud-native applications.

2.5 Low-Code No-code Revolution

The paper (Yan, 2021) states that Low/No-code development is a software development method that provides users with a platform for visually creating applications with little or no coding. The paper explores the benefits and limitations of Low/No-Code development and modern platforms in the industry. It also analyses how it can be improved and prospects the impacts of Low/No-Code development on society and related industries in the future.

The paper's key findings are that Low/No-Code development is a promising trend that can significantly impact future software development and digital transformation. The paper notes that Low/No-Code development gives non-IT professionals a convenient tool for rapidly building simple business applications they need without or with little coding. This can lead to increased productivity and efficiency in organisations. The paper also notes that Low/No-Code development can help bridge the gap between IT and business departments, allowing business professionals to create applications without relying on IT departments.

The paper also discusses some limitations of Low/No-Code development, such as the potential for security vulnerabilities and the need for more research and assessments to solve the limitations and issues with current Low/No-Code technology. The paper suggests that organisations can cooperate with Low/No-Code platform providers to improve their platforms. Low/No-Code platform vendors should continuously research and address limitations by studying competitors' work and seeking to combine the latest technologies with the platforms.

In terms of impacts, the paper suggests that Low/No-Code development will play a crucial role in digital transformation and cause a turnaround in the software development industry. It notes that Low/No-Code development can help organisations respond to dynamic and complex market environments and requirements. The paper also suggests that Low/No-Code development can democratise software development and make it more accessible to non-IT professionals, leading to increased innovation and creativity in the industry.

2.6 Digital Transformation and the Covid-19 Era

Digital transformation refers to integrating digital technologies into all business areas, fundamentally changing how businesses operate and deliver value to their customers. It is a cultural shift that requires organisations to continuously challenge the status quo, experiment, and embrace new working methods. Importance of Digital Transformation:

- Competitive Advantage: Digital transformation allows businesses to leverage technology to create better customer experiences, streamline operations, and quickly adapt to changing market conditions.
- Operational Efficiency: Automation and other digital tools can significantly reduce manual processes, leading to cost savings and faster decision-making.

- Innovation: Embracing a digital-first approach often leads to developing new business models, products, and services.
- Data Utilization: Digital transformation enables businesses to collect, analyze, and act on data in real time, leading to better-informed decisions.
- Customer Experience: Modern customers expect seamless digital experiences. Businesses that can deliver these experiences can foster loyalty and drive growth.

Given the paradigm shift from a physical workplace to a digital one during the pandemic, understanding digital transformation is crucial across the pre-COVID, COVID-19, and post-COVID periods. For instance, the rise of remote working tools like Zoom, Microsoft Teams, and Slack exemplifies how businesses have adapted to this new digital-centric environment. Similarly, the education sector has embraced platforms like Google Classroom, Microsoft Teams for online learning, and Coursera. By spearheading and supporting this change, these tech companies highlight the growing significance of digital solutions in addressing contemporary challenges across diverse fields. In a related vein, digital payments in India experienced an unprecedented surge during the pandemic, becoming an essential tool in the country's response. Platforms such as UPI-based apps like Google Pay, PhonePe, and Paytm facilitated contactless transactions, ensuring commerce's continuity. This shift bolstered economic resilience during challenging times and hastened India's progression towards a more digitally inclusive economy.

Pre-Covid Era:

- Gradual Adoption: Many businesses were slowly integrating digital solutions, with some industries lagging due to a perceived lack of urgency or the costs associated with transformation.

- Physical Over Digital: While online platforms grew, many businesses still prioritised physical interactions and traditional business models.
- Initial Resistance: Some sectors expressed scepticism and resistance, viewing digital transformation as a buzzword or a trend that might fade.

Covid-19 Era:

- Rapid Acceleration: The pandemic forced businesses to pivot quickly. Remote work, e-commerce, online education, and telehealth became the norm, necessitating rapid digital adoption.
- Survival Mechanism: For many, digital transformation was no longer a choice but a necessity for survival as lockdowns and restrictions limited physical interactions.
- New Business Models: Businesses had to innovate, leading to the rise of contactless deliveries, virtual events, and online consultations.

Post-Covid Era:

- Hybrid Models: As the world adjusts to a "new normal," a blend of physical and digital interactions is expected. Workplaces might adopt hybrid models, combining remote and in-office work.
- Digital Maturity: Businesses will likely continue to invest in digital infrastructure, aiming for more advanced stages of digital transformation.
- Continued Innovation: The lessons learned during the pandemic will likely spur continued innovation, with businesses seeking to be better prepared for future disruptions.
- Emphasis on Resilience: Post-Covid digital strategies will likely prioritize flexibility and resilience, ensuring businesses can adapt to changing conditions quickly.

In conclusion, while digital transformation was already underway before the pandemic, Covid-19 acted as a catalyst, accelerating its adoption across sectors. The post-Covid era likely sees a world where digital integration is a foundational aspect of business strategy, emphasising flexibility, innovation, and customer experience.

As its name suggests, digital transformation transforms industries, society, and individuals as never before. According to Microsoft CEO Satya Nadella, two months have witnessed the digital transformation of two years during the pandemic, which was accurately stated. Covid-19 had a more significant impact on the acceleration of digital transformation than CEOs, CTOs, or CIOs, source: https://www.forbes.com/sites/peterhigh/2020/05/26/who-led-your-digital-transformation-your-cio-or-covid-19.

Along similar lines to the global pandemic, this book titled "How to Alleviate Digital Transformation Debt: post-COVID-19" (Khoshafian, 2020) Dr. Setrag Khoshafian emphasises the concept of "debt" in the context of digital transformation, drawing parallels with "technical debt." The author highlights that organisations that opt for quick fixes over comprehensive solutions accumulate this "debt," which can have severe repercussions. The book provides insights into the meaning of Digital Transformation, its opportunities, core digital technologies, best practices, and practical recommendations to alleviate the Digital Transformation Debt. The chapters cover various aspects, from the importance of organisational culture and the rise of Low-Code/No-Code platforms to the significance of data-centric enterprises, the changing customer behaviours, and the increasing interconnectedness through IoT. The book offers a holistic approach to addressing Digital Transformation Debt and suggests ways to mitigate it.

It also delves into the rising prominence of low-code and no-code platforms. These platforms promise to simplify software development, likening the process to using applications like Word or PowerPoint, enabling average business users to progress projects without requiring an engineering team. While no-code platforms are designed for those without coding skills, low-code platforms accelerate software development by allowing developers to utilize pre-written code components. Despite its advantages, the low-code/no-code landscape is intricate, with many solutions catering to different business sizes and needs. The article also touches upon the challenges associated with these platforms, such as the cultural shift required in organisations, the learning curve, the potential need for multiple platforms, limited community support, and confusing pricing structures. Dr. Setrag Khoshafian, the author, concludes that while low-code/no-code is not a cure-all, its benefits in speeding up development and enhancing productivity are undeniable.

- Culture: The article discusses the profound impact of the Covid-19 lockdown on businesses and governments, emphasising the cultural shifts and trends it has instigated. Even before the pandemic, digitisation was transforming various aspects of our lives, both technologically and culturally. However, Covid-19 has accelerated this digital transformation. The author refers to the concept of "Digital Transformation Debt (DTD)," suggesting that organisations that fail to address their "debts" or challenges will face severe consequences. The article highlights the importance of organisational culture in the post-Covid-19 era, emphasising the need for a shift towards a more virtual, flattened, and empowered organisational structure. The piece also touches on the changing dynamics of the workplace, focusing on trust, empowerment, and the balance between home and office work. The author advocates for "Servant Leadership" to drive organisational cultural change.

- Operational Excellence & VSaaS: Digital Transformation Debts series delves into the challenges organisations face in achieving operational excellence amidst the pandemic. The article underscores the importance of Value Stream digitisation and automation, particularly in customer service and inter-enterprise collaboration. A joint McKinsey and Harvard Business School study highlighted that agile organisations fared better during the pandemic. The article emphasises the need for businesses to become "agile enterprises" and discusses the resurgence of business process re-engineering due to the economic downturn. The piece also touches on the problem of silos, which were exposed during the Covid-19 lockdown, and the opportunities for innovation in operational excellence. The article introduces the concept of Value-Stream-As-A-Service (VSaaS) as a new model for inter-enterprise collaboration, emphasising its importance in the post-Covid-19 era. The author concludes with recommendations for enterprises, including the importance of measuring and improving processes, re-engineering existing processes, and leveraging VSaaS for dynamic partnerships.
- Automation: The third pillar in the series addresses the implications of the Covid-19 pandemic on automation and its impact on the workforce. Before the pandemic, automation was already transforming the labour landscape, but the crisis intensified this shift. The article cites studies predicting significant job losses due to automation, with the pandemic further exacerbating these numbers. The spectrum of work affected by automation ranges from repetitive tasks increasingly handled by robots and software to AI-assisted work, where intelligent systems aid workers. The article also touches on the rise of cognitive work, where human expertise remains irreplaceable. The piece delves into the concept of Industry 4.0, highlighting the increasing importance of automation in manufacturing and supply chain management. The author introduces the idea of hyper-automation, which refers to applying advanced technologies to enhance processes and

augment human capabilities. The article concludes with recommendations for businesses, emphasising the need for reskilling, upskilling, prioritising automation opportunities, and adopting an entrepreneurial mindset to navigate the challenges of the post-Covid-19 era.

- No Code Citizen Developers: The fourth pillar in the series explores the rise of "Citizen Developers" and the transformative potential of LCNC platforms in the post-Covid-19 era. The article emphasises that Low Code/No Code represents the next evolutionary step in programming, allowing for rapid application development without the traditional coding process. This paradigm shift enables business stakeholders, subject-matter experts, and even conventional programmers to develop applications quickly. The article highlights the distinction between LCNC platforms, with the former requiring some coding and the latter being drag-and-drop. The potential productivity gains from using these platforms are substantial, with some benchmarks showing improvements of up to 5X to 7X compared to traditional development methods. The author also discusses the various LCNC platforms, from web and mobile application platforms to business process management platforms. The article concludes with recommendations for organisations, emphasising the importance of platform selection, reskilling/upskilling for Low Code/No Code, fostering a Citizen Developer culture, and leveraging design sprints for efficient application development.

The post-Covid-19 landscape has undeniably accelerated the digital transformation journey for organisations worldwide. These articles collectively shed light on the multifaceted challenges and opportunities ahead. The pandemic has reshaped business paradigms from the cultural shifts emphasising trust, empowerment, and a balance between remote and office work to the pressing need for operational excellence through Value Stream digitisation. Automation's intensified role is evident, with a spectrum ranging from repetitive tasks to AI-assisted work, emphasising the need for businesses to adapt and

upskill their workforce. Perhaps the most transformative is the rise of Citizen Developers and the Low Code/No Code movement, which promises to democratise application development and obliterate traditional silos. As organisations navigate this new terrain, the overarching theme is clear: agility, continuous learning, and innovation are paramount. Embracing these changes will help businesses settle their "Digital Transformation Debts" and thrive in a world forever changed by a global pandemic.

2.7 Generative AI

Digital transformation in the AI era, paired with the use of Large Language Models (LLMs), could bring about substantial changes in several areas. LLMs can understand and generate human-like text, which allows them to automate digital content creation, provide superior customer support, and offer personalised recommendations. LLMs can process and analyse vast amounts of data more quickly and precisely than humans, leading to more insightful decision-making and strategy formulation. Also, LLMs can alleviate scalability problems in business, enhancing efficiency and productivity. Hence, integrating LLMs into digital transformation will massively augment the capabilities of AI, bringing profound impacts in many sectors, including business, healthcare, education, and more.

The significance of Codex and similar LLMs in this context is immense. OpenAI's Codex, for instance, is a powerful LLM that can generate code, making it a game-changer in software development. Codex can understand a wide range of programming languages and can generate code based on natural language instructions. This ability to generate code from natural language instructions can significantly reduce the time and effort required to write code, making software development more accessible to a broader range of people. This is particularly relevant in the growing trend towards low-code/no-code platforms, which aim to make software development more accessible to non-programmers.

Here are several papers that merit discussion on the topic of code generation:

- Evaluating Large Language Models Trained on Code (Chen, et al., 2021) paper delves into evaluating large language models specifically trained on code, spotlighting the Codex model. Codex, a derivative of the GPT language model, has been optimized using code from GitHub, with a primary focus on enhancing its Python coding abilities. Interestingly, this model serves as the backbone for GitHub Copilot. The paper introduces "HumanEval," a unique evaluation method designed to gauge the accuracy of code generated from docstrings. In this assessment, Codex emerged as a frontrunner, solving 28.8% of the tasks, while GPT-3 could not address any, and GPT-J managed 11.4%. A pivotal discovery was the efficacy of the "repeated sampling" strategy. By drawing multiple samples for each problem, the researchers boosted the model's problem-solving rate to 70.2%. However, Codex is not without its challenges. It grapples with intricate docstrings, especially those detailing extended operations or requiring meticulous variable bindings. Beyond the technicalities, the paper also broaches the broader ramifications of such cutting-edge code-generating models, touching upon their safety, security, and potential economic impacts. This research offers a holistic view of the potential and challenges associated with code-focused language models, emphasising Codex.
- A Systematic Evaluation of Large Language Models of Code (Xu, Alon, Neubig, & Hellendoorn, 2022) Large language models (LMs) designed for code like Codex have demonstrated impressive understanding and generating code capabilities. However, a significant challenge in the field is that many top-tier code LMs, including Codex, are not open to the public. This lack of accessibility raises concerns about the methodologies and data choices behind these models. Researchers thoroughly assessed

the most prominent available models across various programming languages, like Codex, GPT-J, and others to address this. Interestingly, even though some of these models were primarily built for natural language tasks, they performed comparably to Codex in specific programming languages. To fill the gap of a dedicated, open-source code LM, the researchers introduced PolyCoder. This model, built on the GPT-2 framework and trained on a vast multi-lingual code dataset, outperformed all other models in the C programming language. A significant advantage is that PolyCoder, among other models from the study, is open-source, promoting further research and development in the code LM arena. This research highlights the potential of large code LMs and presents a promising new contender, PolyCoder, in code comprehension and creation.

- CodeT5+: Open Code Large Language Models for Code Understanding and Generation (Wang, et al., 2023) Large language models (LLMs) trained on extensive code databases have made remarkable strides in understanding and generating code. However, they often suffer from two main drawbacks: a rigid architecture that might not be ideal for all tasks and a narrow focus during their training phase, which might not cater to all downstream applications. Researchers developed "CodeT5+", a more adaptable LLM for code-related tasks to overcome these challenges. This unique model blends various training objectives, ensuring it is well-prepared for diverse tasks. Instead of building it from the ground up, they enhanced existing LLMs, making the development process more efficient. They also fine-tuned it with natural language instructions to align it more with human-like tasks. When tested, CodeT5+ outperformed its peers in numerous code-related benchmarks, setting new records in tasks like code generation and code completion. One of its versions even set a new

standard in the HumanEval code generation challenge, showcasing its superior capabilities compared to other available code LLMs.

- GPT-4 Technical Report (OpenAI, 2023) delves into the intricacies and potential of the GPT-4 model. One of its standout features is its multimodal capability, allowing it to process both text and image inputs. This suggests that GPT-4 could transform visual elements like diagrams or flowcharts into corresponding code, seamlessly integrating visual design with coding. Its performance, comparable to human levels in various benchmarks, underscores its advanced reasoning and understanding. This could translate to generating deeply contextual code that meets specific requirements. Built on the Transformer architecture, GPT-4 is naturally equipped for tasks like code completion. Its post-training alignment ensures the generated code is accurate and aligns with best practices. Another highlight is its scalable infrastructure, making it adept at handling extensive codebases or intricate software projects. Moreover, GPT-4's performance can be predicted based on its smaller counterparts, indicating its adaptability for specific code tasks without heavy computational demands. In essence, GPT-4, with its advanced features, emerges as a potential powerhouse for tasks related to code generation, offering a bridge between conceptual design and tangible code.

The research on large language models trained on code, such as Codex, GPT-J, PolyCoder, CodeT5+, and GPT-4, holds immense potential for these platforms. Firstly, models like Codex, which powers GitHub Copilot, can assist users in generating accurate code snippets based on simple descriptions, making the development process more intuitive. Introducing evaluation methods like "HumanEval" ensures the generated code is high quality and meets the desired functionality. Secondly, as highlighted in the research, the systematic evaluation of these models ensures that the best practices and methodologies are adopted, leading to more reliable and efficient code generation. The open-source nature

of models like PolyCoder fosters community-driven enhancements, ensuring the generated code is up-to-date with the latest programming paradigms. Furthermore, with their advanced capabilities, models like CodeT5+ and GPT-4 can transform visual inputs into code, bridging the gap between visual design tools commonly used in Low-code platforms and the underlying code. These research advancements can supercharge Low-code and No-code platforms, making them more powerful, efficient, and user-friendly, further democratising software development.

Low-code/no-code platforms are transforming the software development landscape by enabling people with little or no coding experience to create applications. These platforms provide a visual interface for designing applications, and they automate the underlying code generation. The rise of these platforms is part of a broader trend toward democratizing software development, making it more accessible to a more comprehensive range of people. However, while these platforms are powerful, they have limitations. They often need more flexibility and power than traditional coding, and they can struggle with more complex tasks.

This is where LLMs like Codex can come in. By generating code based on natural language instructions, Codex can bridge the gap between the simplicity of low-code/no-code platforms and the power of traditional coding. For example, a user could describe the functionality they want in natural language, and Codex could generate the corresponding code. Automating some of the more mundane aspects of coding could make it easier for non-programmers to create more complex applications and speed up the development process for experienced programmers.

Moreover, using LLMs like Codex in digital transformation goes beyond code generation. These models can also automate digital content creation, provide customer support, and offer personalised recommendations. For example, they could automatically

generate product descriptions, answer customer queries, or recommend products based on a user's browsing history. This could lead to more efficient and personalised customer experiences, increasing customer satisfaction and loyalty.

Furthermore, LLMs can process and analyse vast amounts of data more quickly and precisely than humans, leading to more insightful decision-making and strategy formulation. For example, they could analyse customer behaviour data to identify trends and patterns, which could then be used to inform business strategy. This could lead to more effective and targeted marketing campaigns, improved product development, and better business decisions.

In conclusion, the integration of LLMs like Codex in digital transformation has the potential to bring about profound changes in many sectors. By automating tasks like code generation and data analysis, these models can enhance efficiency and productivity, make software development more accessible, and drive more insightful decision-making. As such, they represent a powerful tool for businesses and organisations looking to leverage the power of AI in their digital transformation efforts.

McKinsey's special report "The economic potential of Generative AI: The next productivity frontier" (Chui, et al., 2023) suggests that generative AI could revolutionise roles across various sectors, potentially unlocking trillions of dollars in value. Specifically, generative AI could contribute between $2.6 trillion to $4.4 trillion annually across 63 use cases analysed. This impact is especially pronounced in areas like customer operations, marketing and sales, software engineering, and R&D. Technology also has the potential to reshape the nature of work, automating tasks that currently take up a significant portion of employees' time. As the era of generative AI is just beginning, businesses and society face the challenge of harnessing its full potential while addressing associated risks.

The article “Digital transformation: Rewiring for digital and AI” (McKinsey Digital, 2023) from McKinsey & Company emphasises the importance of leadership in driving successful digital transformations, particularly in digital and AI advancements. The insights provided highlight the critical role of the C-suite in spearheading initiatives that ensure organisations remain competitive in the digital and AI age. The content showcases various case studies, including how certain enterprises have effectively integrated digital strategies to achieve enterprise-wide gains. Additionally, the importance of harnessing the potential of 'superpower' technologies like AI is underscored, emphasising the need for skilled developers to realise this potential fully. The overarching theme is organisations' need to rewire their operations and strategies to harness the benefits of digital and AI transformations.

This "Digital Transformation Playbook" in the book “Digital Hesitation: Why B2B Companies Aren't Reaching Their Full Digital Transformation Potential” (Lah & Wood, 2022) offers a comprehensive guide for technology-centric B2B companies aiming to navigate the next five years of their development. It outlines the essential tactics for establishing a successful X-as-a-Service business model, emphasising the importance of digital customer experience (DCX), data-driven sales, and outcome-aligned pricing. The playbook underscores the significance of understanding and implementing a full-scale digital transformation, highlighting its potential to drive efficient growth and maintain competitiveness in the ever-evolving technology market. It sheds light on the pivotal role of customer interaction in determining the success or failure of businesses, as evidenced by the B2C wars of the past two decades. The book further explores the challenges B2B companies face in innovating their operational models and the urgent need to address these challenges to realise the full potential of their digital transformation endeavours. Drawing from the insights of industry experts, the book provides a holistic view of the digital

transformation journey, from securing board commitment to redefining sales and service delivery models.

"Digital Transformation: Survive and Thrive in an Era of Mass Extinction" (Siebel, 2019) by Thomas M. Siebel offers a profound examination of the disruptive technologies shaping the modern business and governmental landscape. Siebel discusses the convergence of four pivotal technologies: elastic cloud computing, big data, artificial intelligence, and the Internet of Things. He presents a compelling narrative on how these technologies revolutionise businesses and governments' functioning in the 21st century. Through insightful discussions, Siebel provides a roadmap for organisations to harness these technologies as strategic opportunities, showcasing how leading enterprises leverage AI and IoT to achieve remarkable results. This book is an essential guide for business and government leaders aiming to navigate the challenges and opportunities of the digital age.

Conclude with a talk emphasising that adopting technology and undergoing cloud transformation is not merely about hopping on the bandwagon, which applies to digital disruption. In a light humour talk (Hariri, The Silver Bullet Syndrome, 2022), Hadi Hariri, VP at JetBrains, delves into the tech industry's perpetual quest for the next big solution, often viewed as a panacea for all past challenges. Hariri reflects on developers' journey, observing that pursuing simplicity usually results in unnecessary complexity, even after trying various solutions. He questions whether this complexity is merely a facade for job security and contemplates the industry's future direction. Are we heading towards an even more convoluted tech landscape, or is there hope for simplicity? In another talk (Hariri, 2015), Hariri criticises the tech community's infatuation with the "next big thing," whether it is a new language, framework, or platform. He emphasises the need to critically evaluate the actual benefits of these new tools against their costs, significantly when the financial burden does not directly impact the adopters. Hariri challenges the audience to introspect:

Are we genuinely seeking a silver bullet, or are we just riding the technology bandwagon without considering the implications? It is also prevalent in tech companies, particularly concerning the hype surrounding new technologies. CIOs often find themselves in a predicament where upper management is eager to adopt the latest technological trends and needs to thoroughly understand the problems these technologies aim to solve. This eagerness is frequently driven by the desire to appear innovative and cutting-edge. Such demands can arise suddenly, with vague justifications for the need for a specific technology and ambiguous expected outcomes. A classic scenario occurs when executives, after being exposed to a new tech trend—perhaps through a magazine or seminar—rush to their teams, insisting that the company must urgently adopt this new technology. The disproportionate attention and inflated market valuations that new technologies often attract exacerbate the situation. For instance, many companies hastily label themselves as "data-driven" or even a "technology company" when their primary business might be far removed from such designations. While technology can unlock significant value and provide competitive advantages, it does not necessarily alter a company's core identity. Addressing this syndrome requires a multifaceted approach:

- Work Backwards Together: Initiatives should start with understanding customer value and then working backwards to determine the technology needed.
- Increase Data Literacy: Companies should invest in educating their teams about data and its practical applications.
- Buy Options: Instead of committing to one tool or technology, companies should explore multiple options, especially in the early stages.
- Operationalize Insights: Data and technology should lead to actionable insights that can be implemented on the ground.

While technology can be a powerful enabler, companies must differentiate between genuine needs and mere tech hype. Adopting technology without a clear understanding of its purpose and benefits can result in wasted resources and missed opportunities. For instance, the adoption of Generative AI, which has been disruptive in 2023, is just hype or a genuine need for an organisation's business growth that requires a thoughtful approach. Human creativity should guide this decision rather than simply mimicking patterns, as machine learning might do when influenced by hype.

Navigating the evolving landscape of project management and financial operations, it is essential to understand two pivotal frameworks: the PMI Citizen Development framework and the principles set forth by the FinOps Foundation.

2.8 PMI Citizen Development Framework

PMI Citizen Development Framework (PMI, 2022) includes the PMI Citizen Developer Foundation, offers a foundational understanding of citizen development, introducing the Citizen Development Canvas and emphasising best practices in the methodology. Building on this, the PMI Citizen Developer Practitioner focuses on the essential methodologies that enable the creation of efficient, effective, and scalable applications through low-code and no-code platforms. At a more advanced level, the PMI Citizen Developer Business Architect equips leaders with vital insights to streamline the governance process, foster stakeholder collaboration, and establish organisational frameworks that bolster the success of citizen development initiatives.

The PMI Citizen Development Framework is a structure or guideline developed by the Project Management Institute (PMI). This framework, created by industry experts,

provides a structured approach to adopting Low-code and no-code platforms. It allows organisations to integrate low-code and no-code solutions into their operations more efficiently and in a more organized way.

2.9 FinOps Foundation

FinOps, short for Financial Operations by FinOps Foundation, is a strategic approach to cloud financial management (Conway, Saleme, Srivathsan, & Tyrman, 2023). As organisations transition to cloud-based infrastructures, they often face challenges in managing and optimising cloud costs. This is where FinOps comes into play. It emphasises balancing speed, cost, and quality, ensuring businesses extract the maximum value from their cloud investments. The FinOps way fosters collaboration between IT, finance, and business teams, enabling them to make informed decisions about cloud usage and expenditures. By adopting FinOps practices, organizations can avoid common pitfalls like unforeseen expenses and misaligned business goals. The result is a more cost-effective, agile, and efficient cloud strategy that drives savings and innovation.

By integrating the principles and methodologies covered in the Citizen development framework and FinOps, organisations can achieve enhanced agility, transparency, and efficiency, ensuring they remain competitive and responsive in an ever-evolving business landscape.

2.10 Summary

After reading nearly 150 research papers, five books, several articles, and podcasts on transformation work, around 25-30 papers with a few books are narrowed down for the literature review. These papers specifically discussed the challenges, approaches, tools, and

trends in cloud transformation. Additionally, they delved into the impact of the Low-code No-code revolution on digital transformation efforts during the Covid-19 era and the recent advancements in Generative AI that fuel Low-code No-code platforms.

Cloud transformation is an essential step towards digital transformation, enabling businesses to navigate the technological battles to stay relevant in the digital arena. Cloud transformation paves the way for new opportunities in machine learning, AI, IoT, Analytics, Business intelligence, and Blockchain on the planet's scale. It facilitates the shift from CAPEX to OPEX costs. This transformation is a significant journey for any organisation of any size, presenting challenges in strategy definition, implementation, and operationalisation. Numerous research studies delve into its benefits, adoption approaches, and challenges. Cloud transformation has matured significantly and is extensively covered in several systematic literature reviews.

2.11 Observations

After delving into the multifaceted realms of cloud transformation challenges, the nuances of modernisation and cloud-native paradigms, the intricacies of Kubernetes and serverless architectures, and the diverse offerings from leading cloud vendors like AWS, Azure, GCP, and IBM, coupled with the rise of Low-code and No-code platforms, it becomes evident that the technological landscape is rife with complexity. However, this complexity is not insurmountable. By embracing continuous learning, fostering collaboration, and leveraging the right tools and frameworks, organisations can navigate this intricate ecosystem, turning potential challenges into opportunities for innovation and growth. In essence, dealing with complexity requires a blend of adaptability, strategic foresight, and a commitment to harnessing the full potential of emerging technologies.

The Gartner Peer Insights (Enterprise Low-code Application Platform Reviews and ratings, n.d.) provides comprehensive reviews and ratings for products in the Enterprise Low-Code Application Platforms market. The platform offers insights into various low-code application platforms, highlighting their strengths, weaknesses, and user experiences. Some notable platforms mentioned include OutSystems, Appian Low-Code Platform, Power Apps by Microsoft, Salesforce Platform, Quickbase, Mendix Platform by Siemens, and many others. Each product is rated based on user feedback, with ratings ranging from one to five stars, and includes reviews detailing users' experiences with the platform.

However, while Gartner provides valuable insights into these platforms, there is a noticeable gap in research regarding the broader Low-code No-code industry. This includes aspects like the adoption rate of these platforms, their capabilities in different business scenarios, and the maturity model of the technology. Comprehensive research in these areas would provide a more holistic view of the industry's current and potential trajectory. Gartner's article, "Gartner Forecasts Worldwide Low-Code Development Technologies Market to Grow 20% in 2023," highlights the burgeoning growth and adoption of low-code development technologies. According to Gartner's latest forecast, the worldwide market for these technologies is set to reach $26.9 billion in 2023, marking a 19.6% increase from the previous year. This surge is attributed to the rise of business technologists and the increasing number of hyperautomation and composable business initiatives. Varsha Mehta, a Senior Market Research Specialist at Gartner, emphasises that organisations lean more toward low-code development technologies to meet the growing demands for swift application delivery and tailored automation workflows.

The article further reveals that Low-code application platforms (LCAPs) are anticipated to be the dominant component of the low-code development technology market, with a projected growth of 25%, amounting to nearly $10 billion in 2023.

	STAMFORD, Conn., February 16, 2021			STAMFORD, Conn., December 13, 2022			
	Actual	Actual	Forecast	Actual	Actual	Forecast	Forecast
	2019	2020	2021	2021	2022	2023	2024
Low-Code Application Platforms (LCAP)	$ 3,473.5	$ 4,448.2	$ 5,751.6	$ 6,324.0	$ 7,968.0	$ 9,960.0	$ 12,351.0
Intelligent Business Process Management Suites	$ 2,509.7	$ 2,694.9	$ 2,891.6	$ 2,416.0	$ 2,585.0	$ 2,761.0	$ 2,940.0
Multiexperience Development Platforms (MDXP)	$ 1,583.5	$ 1,931.0	$ 2,326.9	$ 2,081.0	$ 2,508.0	$ 2,999.0	$ 3,563.0
Robotic Process Automation (RPA)	$ 1,184.5	$ 1,686.0	$ 2,187.4	$ 2,350.0	$ 2,892.0	$ 3,401.0	$ 3,879.0
Citizen Automation and Development Platform (CADP)	$ 341.8	$ 438.7	$ 579.5	$ 554.0	$ 732.0	$ 953.0	$ 1,232.0
Other Low-Code Development (LCD) Technologies*	$ 59.6	$ 73.4	$ 87.3	$ 92.0	$ 109.0	$ 126.0	$ 146.0
Integration Platform as a Service (iPaaS)	Tech metric - Not introduced in 2021			$ 4,680.0	$ 5,668.0	$ 6,668.0	$ 7,838.0
Overall	$ 9,152.6	$ 11,272.2	$ 13,824.3	$ 18,497.0	$ 22,462.0	$ 26,868.0	$ 31,949.0
Source : Gartner forecast Worldwide Low-code Development Technologies Market							

Figure 2.1 Gartner 2021 and 2022 Actuals and 2023 Forecasts Worldwide Low-Code Development Technologies Market

Additionally, while LCAPs dominate the market, the Citizen Automation Development Platform (CADP) is expected to witness the fastest growth rate, with a forecasted increase of 30.2% in 2023. The article underscores the pivotal role of low-code development tools in enhancing productivity, efficiency, and agility, especially in the context of fusion teams. By 2026, Gartner predicts that developers outside formal IT departments will constitute at least 80% of the user base for low-code development tools, a significant rise from 60% in 2021. The article concludes by emphasising the importance of low-code development technologies in supporting the composable enterprise, enabling the creation of agile and resilient software solutions that can adapt to changing business needs.

The current literature on the low-code and no-code landscape (in such a vast marketplace, as revealed by Gartner) has yet to provide a comprehensive understanding, especially regarding a detailed evaluation of no-code and low-code platforms across a spectrum of parameters. These criteria encompass ease of use, costs associated with development and deployment, ongoing labour and non-labour expenses, the initiation of the platform, and the intricacies of development installation. Additionally, there is a focus on non-functional aspects like support for data platforms, integration capabilities, security measures, scalability, performance metrics, maintainability, observability, user support, and the nuances of upgrades. This research also aims to compare these platforms against

other prevalent strategies in the expansive realm of cloud transformation, including methodologies like lift-and-shift, serverless and cloud-native development. The overarching objective is to arm the industry with in-depth qualitative insights and rigorous quantitative data, facilitating informed decision-making as businesses navigate their digital transformation trajectories.

CHAPTER III:

METHODOLOGY

3.1 Overview of the Research Problem

Recently, some organisations have adopted cloud computing using a 'lift and shift' strategy, transferring their existing workloads to the cloud with minimal alterations. However, this approach often leads to significant inefficiencies and increased costs, as it fails to utilise cloud-native features and services entirely. These inefficiencies manifest as poor performance, increased operational challenges, and higher expenses, undermining the core benefits of cloud computing (Juncal Alonso, 2013). To mitigate these cost issues, consulting firms recommend the FinOps approach (Keith Conway, 2023). This method involves introducing specific roles to audit, control, and manage costs more effectively. Nevertheless, the question remains: Does this approach help or introduce further complexities and expenses?

To tackle these problems, it is suggested that a low-code no-code (LCNC) platform be considered. This platform facilitates a more efficient and successful integration of cloud technology by simplifying the procedures involved in rationalising and optimising workloads for the cloud environment. By utilising LCNC platforms, companies can optimise the alignment of their workloads with the capabilities offered by cloud computing. This strategic approach leads to improved performance, cost reduction, and maximising return on investment in cloud technologies. The implementation of this methodology holds the potential to revolutionise cloud migration tactics, resulting in enhanced efficiency, cost-effectiveness, and overall success of cloud implementations (Woo, 2020).

Nevertheless, the proliferation of low-code, no-code (LCNC) platforms has inherent difficulties. Organisations have challenges evaluating their current workloads and choosing the most appropriate ones for migration and optimisation using these platforms.

The intricate nature of this complexity introduces an additional dimension to the decision-making process, necessitating a more refined methodology for assessing and choosing the suitable LCNC platform. Every platform provides distinct features and capabilities, and it is crucial to determine the most viable and advantageous solution for a particular workload to achieve the required efficiency and cost-effectiveness in cloud deployments. Therefore, it is imperative to develop a complete approach that incorporates a meticulous assessment of current workloads and the capacities of different LCNC platforms to achieve a smooth and effective migration to cloud computing.

3.2 Research Questions/Hypothesis

As businesses increasingly seek cost-effective and efficient solutions for cloud computing, exploring innovative approaches that can optimise cloud workloads and rationalisation becomes imperative. In this context, low-code and no-code platforms have emerged as potential game-changers. This research investigates the viability, specific use cases, attributes, and compatibility of low-code and no-code platforms in optimising cloud services. The following research questions and hypotheses are proposed to guide this investigation:

RQ1: Are low-code and no-code platforms a viable alternative to rationalising cloud workloads for reducing costs on cloud services?

Hypothesis 1: Low-code and no-code platforms significantly reduce the cost of cloud services by simplifying and accelerating the development process, thus offering a viable alternative for rationalising cloud workloads.

- For Simplifying Development: Low-code and no-code platforms can significantly reduce the time and resources needed for developing and deploying applications. This

efficiency can save costs, especially for businesses that do not require highly complex custom software solutions.

- For Non-Technical Users: These platforms empower non-technical users to create and manage applications without deep programming knowledge. This democratisation can reduce the need for specialised and often more expensive development resources.
- For Speed and Agility: Low-code and no-code solutions can accelerate development, allowing businesses to respond more quickly to market changes or internal needs. This agility can translate into cost savings by reducing time-to-market and improving business responsiveness.
- For Complex Customizations: Low-code and no-code platforms might not offer flexibility and customisation for businesses with precise, complex requirements. In such cases, traditional development approaches might be more cost-effective in the long run.
- For Long-Term Scalability: While low-code and no-code platforms are excellent for rapid development, they pose challenges in scaling, especially for large enterprises with complex systems. Over time, the cost of maintaining and scaling these solutions might outweigh the initial savings.
- Depends on the Use Case: The viability of these platforms as a cost-saving measure largely depends on the specific use case. They can be highly cost-effective for small to medium-sized businesses with standard application needs. However, traditional development might suit large enterprises or applications requiring extensive customisation.
- Integration Challenges: Integrating low-code/no-code solutions with existing legacy systems and databases can be challenging and might incur additional costs, reducing the overall cost-effectiveness.

- Vendor Lock-in Risks: Relying on a specific low-code/no-code platform can lead to vendor lock-in, where a business becomes dependent on a vendor's tools and services. This can limit flexibility and potentially lead to higher costs in the future if the company needs to switch platforms.
- Security and Compliance Concerns: Low-code and no-code platforms might only sometimes meet the necessary standards for industries with stringent security and compliance requirements, potentially leading to higher costs in ensuring compliance.
- Cost-Benefit Analysis: Whether low-code and no-code platforms are cost-effective for rationalising cloud workloads depends on a thorough cost-benefit analysis, considering factors like development time, resource requirements, scalability, security, and long-term maintenance costs.

RQ2: What are the possible scenarios, specific use cases, or application types in which organisations looking to optimise their cloud solutions might consider low-code and no-code platforms to achieve a quick turnaround?

Hypothesis 2: Low-code and no-code platforms are particularly effective in scenarios where rapid development, deployment, and scalability are critical, such as data analytics, customer relationship management, and automated workflows.

- Rapid Prototyping and MVP Development: Organisations can use low-code/no-code platforms for rapid prototyping and developing Minimum Viable Products (MVPs). This approach allows businesses to quickly test ideas in the market and gather user feedback without investing heavily in full-scale development.
- Business Process Automation: These platforms can provide quick and efficient solutions without complex coding to automate routine business processes, such as HR processes, inventory management, or customer service workflows.

- Internal Tools and Dashboards: Developing internal tools, such as administrative dashboards, reporting tools, or data visualisation applications, can be efficiently handled using low-code/no-code platforms, saving time and resources.
- Customer Relationship Management (CRM) Applications: These platforms allow for more rapid customisation or extension of CRM systems to suit specific organisational needs, enhancing customer engagement and management.
- E-commerce and Retail Applications: For small to medium-sized e-commerce businesses, low-code/no-code platforms can quickly set up online stores, manage inventory, and integrate with payment and shipping services.
- Event Management and Booking Systems: Organisations can develop event management and booking systems for conferences, workshops, or other events, enabling efficient handling of registrations, payments, and scheduling.
- Educational and Training Tools: Educational institutions or corporate training departments can use these platforms to create interactive learning modules, quizzes, and training programs.
- Marketing Campaign Management: Low-code/no-code solutions can offer quick and flexible tools for launching and managing marketing campaigns, including email marketing, social media management, and lead tracking.
- Mobile App Development for Non-Critical Functions: These platforms can expedite the development of mobile applications for non-critical business functions, such as internal communication or employee engagement.
- Integration of Cloud Services: Low-code/no-code platforms can integrate various cloud services and APIs, creating a unified system that enhances data flow and process efficiency.

- Customized Customer Portals: Building customised portals for customer interaction, support, and feedback can be quickly achieved, improving customer experience and engagement.
- Data Collection and Survey Tools: These platforms offer a quick way to create and deploy data collection tools and surveys and to gather and analyse data for market research or customer feedback.
- Project Management and Collaboration Tools: Developing tools for project management and team collaboration, including task tracking, scheduling, and document sharing, can be efficiently handled with low-code/no-code solutions.
- Compliance and Regulatory Reporting: These platforms can be used to develop customised reporting tools that align with legal standards for industries with specific compliance and regulatory reporting requirements.

RQ3: What attributes of low-code and no-code platforms should be considered when assessing workloads against these platforms' features for investing in modernisation or rationalisation efforts?

Hypothesis 3: Key attributes like scalability, security, integration capabilities, and user-friendliness of low-code and no-code platforms are crucial in determining their suitability for various workloads in cloud rationalisation efforts.

- Ease of Use and User Experience: The platform should be user-friendly, enabling non-technical staff to build and manage applications. A good user interface and intuitive design are crucial for maximising productivity and reducing the learning curve.
- Customization and Flexibility: It is essential to customize applications to meet specific business needs. The platform should offer enough flexibility to accommodate unique requirements without extensive coding.

- Integration Capabilities: Consider how well the platform integrates with existing systems and data sources. Seamless integration with other tools and services is vital for ensuring smooth operations and data consistency.
- Scalability: The platform should scale with the business. This includes handling increased user loads, data volume, and application complexity as the industry grows.
- Security and Compliance: Assess the platform's security features and ability to comply with industry regulations. This is particularly important for industries that handle sensitive data or are subject to strict regulatory requirements.
- Performance and Reliability: The platform should be robust and reliable, with minimal downtime. Performance considerations are crucial, especially for applications that require real-time processing.
- Development Speed: One of the primary benefits of low-code/no-code platforms is the development speed. Evaluate how quickly applications can be built, tested, and deployed.
- Cost-Effectiveness: Consider the total cost of ownership, including licensing fees, training costs, and maintenance expenses. The platform should offer a good return on investment.
- Support and Community: Good vendor support and an active user community can be invaluable for troubleshooting, sharing best practices, and staying updated on new features.
- Mobile Responsiveness and Cross-Platform Compatibility: The platform should support the development of responsive and compatible applications across various devices and platforms.

- Version Control and Update Management: Assess how the platform handles version control and application updates. It should allow for smooth transitions and minimal disruption during updates.
- Reporting and Analytics: Generating reports and analytics is essential for monitoring application performance and making data-driven decisions.
- Vendor Stability and Roadmap: Consider the platform's provider's stability and future development roadmap. A stable vendor with a clear vision is crucial for long-term success.
- Collaboration Features: The platform should facilitate collaboration among team members, including features for sharing, commenting, and tracking changes.

RQ4: How are low-code and no-code platforms different from SaaS offerings, and how does integration with a SaaS platform affect the choice of low-code and no-code platforms?
Hypothesis 4: Low-code and no-code platforms offer greater customisation and flexibility than SaaS offerings, making them more suitable for businesses requiring bespoke solutions. Integration with SaaS platforms enhances these platforms' functionality and broadens their applicability.

- Fundamental Differences:
  - Low-Code/No-Code Platforms: These platforms enable users to create applications with minimal coding, often through a graphical user interface. They are designed to simplify the software development process, allowing users to build custom applications quickly.
  - SaaS Offerings: SaaS refers to software hosted in the cloud and accessed via the internet. It is typically a finished product that end-users subscribe to and use without the need to develop or maintain the software themselves.

- Customization and Flexibility: Low-code and no-code platforms offer more customisation and flexibility than typical SaaS products. Users can build bespoke applications tailored to their specific business processes, whereas SaaS products are generally more rigid in their functionality.
- Integration Capabilities: The choice of a low-code/no-code platform can be significantly influenced by its ability to integrate with existing SaaS solutions. A platform that seamlessly integrates with a business's existing SaaS ecosystem can extend the functionality of those services and create more cohesive workflows.
- Control and Ownership: With low-code/no-code platforms, businesses have more control over their applications and own the custom solutions they create. In contrast, SaaS products are owned by the service provider, and users have limited control over the software's functionality.
- Cost Considerations: The cost structures are different. SaaS typically involves subscription fees, while low-code/no-code platforms might have licensing fees based on usage, development, and deployment scales.
- Development and Maintenance: Low-code/no-code platforms require some level of development and maintenance from the user's side, albeit much less than traditional software development. On the other hand, the provider maintains SaaS solutions, with updates and support typically included in the subscription.
- Purpose and Use Cases: SaaS solutions are often designed for specific applications (like CRM, ERP, and email marketing), while low-code/no-code platforms are more about providing a general environment to create a wide range of applications.
- Data Handling and Security: When integrating with SaaS platforms, it is essential to consider how data will be handled and secured across different systems. The chosen

low-code/no-code platform should comply with the organisation's data security and privacy standards, especially when integrating with external SaaS applications.

- Scalability: The scalability of low-code/no-code solutions, especially when integrated with SaaS platforms, is a critical consideration. The platform should be able to scale as the business grows and as the demands for integrated SaaS solutions change.
- Vendor Lock-In: Vendor lock-in is risky with both SaaS and low-code/no-code platforms. However, with low-code/no-code platforms, exporting and migrating applications to other environments might be more flexible than SaaS products.

RQ5: How does support for open standards and integration with other cloud models, such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Function as a Service (FaaS) — often associated with serverless computing — and Container as a Service (CaaS), affect the choice of low-code and no-code platforms?

Hypothesis 5: The effectiveness of low-code and no-code platforms is enhanced by their interoperability and integration with various cloud models, such as IaaS, PaaS, FaaS, and CaaS, providing a comprehensive solution for businesses seeking to optimise their cloud infrastructure.

- Enhanced Flexibility and Interoperability: Support for open standards in low-code/no-code platforms ensures better interoperability with various cloud services and models. This flexibility is crucial for organisations that use a mix of cloud services and want to ensure seamless integration across their IT infrastructure.
- Broader Range of Applications: When a low-code/no-code platform integrates well with IaaS, PaaS, FaaS, and CaaS, it broadens the range of applications that can be developed. For instance, it allows businesses to leverage the scalability of IaaS, the

development environment of PaaS, the event-driven architecture of FaaS, and the portability of CaaS.

- Cost-Effective Scaling and Resource Management: Integration with IaaS and PaaS can make scaling and managing resources more cost-effective. For example, a low-code platform that integrates with a PaaS can automatically scale the underlying infrastructure, reducing the need for manual intervention and resource allocation.
- Leveraging Serverless Architectures: Integration with FaaS or serverless computing models allows businesses to build highly scalable and efficient applications in resource usage. This is particularly beneficial for applications with variable workloads or needing to scale automatically based on demand.
- Containerisation Benefits: Compatibility with CaaS platforms means that applications developed on low-code/no-code platforms can be containerised. This offers portability, consistency across different environments, and more efficient use of underlying resources.
- Simplifying Complex Architectures: For organisations that utilise complex cloud architectures, a low-code/no-code platform that integrates well with these services can simplify developing and managing applications that interact with various cloud components.
- Speed of Development and Deployment: Integrating with various cloud models can speed up the development and deployment. For instance, using a PaaS integrated with a low-code platform can streamline the deployment process, while integration with IaaS can expedite the provisioning of necessary infrastructure.
- Compliance and Security: When low-code/no-code platforms support open standards and integrate with established cloud models, it can be easier to ensure compliance with

industry standards and security protocols, as these models often come with built-in compliance and security features.

- Vendor Lock-in Considerations: A platform that supports open standards and integrates with various cloud services can reduce the risk of vendor lock-in, providing more flexibility for businesses to switch providers or cloud models if needed.
- Innovation and Future-Proofing: Choosing a low-code/no-code platform that integrates with advanced cloud models like FaaS and CaaS can position an organisation to take advantage of emerging technologies and trends, future-proofing their technology stack.

In conclusion, a low-code or no-code platform can be significantly influenced by its support for open standards and its ability to integrate with various cloud models. This integration can affect everything from application flexibility and scalability to cost management, innovation potential, and the ability to meet security and compliance requirements.

3.3 Overall Objective

This research study evaluates no-code and low-code platforms on various parameters such as ease of use, cost of development and deployment, recurring labour and non-labour costs, platform start-up, and development installation. Several parameters on non-functional requirements like data platform support, integration, security, scalability, performance, maintainability, observability, support, upgrade, etc., are compared with other dispositions such as lift-shift, cloud-native development in cloud transformation (Nane Kratzke, 2017) to help the industry use qualitative and quantitative analysis in assisting decision-making exercise on their journey to digital transformation.

3.4 Specific Aims

This research includes qualitative and quantitative analyses of the capabilities of code no-code platforms to help organisations adopt a combined strategy for custom development versus no-code development platforms. Low-code and no-code platforms are relatively recent market entrants with insufficient evidence. There is a need for a comprehensive analysis of the various capabilities of these platforms to determine the viability of considering them alternatively or employing them selectively in certain circumstances to transform the customer experience. This is one of the proposed reasons for conducting additional research on this topic and pursuing these areas of study:

- This study investigates the challenges of cloud transformation and their effects on digital transformation efforts, the emergence of low-code/no-code platforms as a tool for accelerating digital transformation, and their limitations and trends.
- The research examines the literature on cloud transformation, determine the specific methods that led to subpar outcomes, and analyse the most popular platforms to identify areas where low-code development platforms can be the best alternative.

- This exhaustive analysis evaluates the capabilities of No-code and Low-code development platforms, their cost implications, and the industries and applications for which they are best suited.

Low-code/no-code platforms were compared to custom development on the Cloud with cloud-native (Josef Spillner, 2018) and microservices investments in terms of machine learning capabilities, business process mining, robotic process automation, automation engine, data mining, transformation and visualisation, integration, workflow management, and applicability in terms of compliance, organisation size, industry, and region.

3.5 Background and Significance

Over the past decade, cloud transformation has become a crucial precursor to digital transformation for many organisations. The goal is to optimise their workload by effectively planning, implementing, and evaluating Digital Transformation initiatives to reduce CAPEX (Capital Expenditures) costs to OPEX (Operating Expenditures). While this shift provides many advantages, it presents unique challenges, mainly when dealing with specific workloads' complexity and legacy nature.

This comprehensive study explores cloud transformation approaches, challenges, and shortcomings in cloud transformation efforts. It investigates how Low-Code No-Code platforms (LCNCs), also referred to as Low-Code Development Platforms (LCDPs) No-Low Development Platforms (NCDPs), can enhance the efficiency and effectiveness of business processes and workflow transformation, achieving these improvements in a quicker turnaround. Integrating AI with LCNC platforms, using AI-augmented development techniques, enhances digital transformation by automating and optimising the application development process, promising a future of greater automation, increasingly

sophisticated applications with actionable insights, and reimaging employees, customers, and partner experiences. Recent breakthroughs in pre-trained large language models (LLMs) and the Transformer model architecture are anticipated to fuel Low-Code and No-Code platforms with the capability of natural language processing tasks, extending their reach to a broader audience, and unlocking the expanded value of data, visualisation, and insights, which is the lifeblood of digital transformation, thereby enabling more efficient operations, improved decision-making, and more competitive strategy formulation. The next wave of AI-assisted digital transformation can substantially improve productivity, establish new business models, and empower organisations, individuals, and AI agents to be proactive, intelligent, and learn from action/feedback mechanisms, thus promoting value or goal-driven and customer-centric business practices.

The literature review attempts to critically assess existing studies on cloud transformation, which is the strategic integration of digital technology into all aspects of a business, radically transforming how organisations operate and produce value, including a cultural and operational overhaul. Many organisations have successfully managed this shift despite hurdles such as opposition to change and cybersecurity concerns. This literature study explores the ever-changing realm of digital transformation, explicitly examining Low-Code No-Code (LCNC) platforms and techniques for transforming to the cloud. LCNC platforms have brought about a revolutionary change in software creation by making the process accessible to a broader audience. These platforms empower developers and non-technical people to construct programs using graphical interfaces. Although there are limitations in personalisation and scalability, the fact that they are widely adopted highlights their significance in driving digital progress (Yajing Luo, 2021). Furthermore, cloud transformation, a fundamental aspect of several digital endeavours, provides firms worldwide access to computing resources that can be instantly accessed. Nevertheless, this

process is complete with difficulties, encompassing deficiencies in capabilities, opposition to alteration, apprehensions regarding cybersecurity, and the expanding gap in digital access. The examined research papers offer valuable insights into several cloud migration strategies, including hybrid, multi-cloud, and cloud-native models. They also introduce tools such as the Cloud Transformation Advisor, which aids in making strategic decisions (Chee, Zhou, Meng, Bagheri, & Zhong, 2011). These studies emphasise the intricacies and the imperative for educated decision-making in cloud migration and digital transformation endeavours.

Ultimately, incorporating Large Language Models (LLMs) such as Codex into the digital transformation process will result in significant changes across multiple industries (Frank F. Xu, 2022). LLMs improve efficiency and productivity in software development by automating operations such as code creation and data analysis (Yue Wang, 2023). This automation also makes software development more accessible and facilitates informed decision-making. Therefore, they serve as an essential instrument for enterprises and organisations aiming to utilise the capabilities of AI in their efforts to undergo digital transformation. The literature review emphasises the significant impact of cloud services, artificial intelligence (AI), and low-code/no-code (LCNC) platforms in modernising business operations and strategy (Khoshafian, 2020). In the aftermath of the Covid-19 pandemic, enterprises must adapt to the changing digital world using AI and cloud technologies. This is crucial for success in a rapidly changing digital environment. Various research papers offer deep insights into evolving technologies and strategies in digital transformation and cloud computing. These studies highlight significant trends, challenges, and innovations in the field, as summarised in the following key points:

LCNC Platforms:

- These platforms democratise software creation by enabling developers and non-technical users to construct programs via graphical interfaces (Raquel Sanchis, 2020).
- They have limitations like potential customisation and scalability concerns but play a critical role in digital transformation. (Yan, 2021)

Cloud Transformation:

- Cloud computing enables organisations to access computing resources globally with elasticity, reducing costs and improving flexibility and scalability. (Chee, Zhou, Meng, Bagheri, & Zhong, 2011)
- It presents hurdles like lack of digital skills, resistance to change, and cybersecurity concerns.

Digital Business Strategy:

- Challenges include aligning IT and business strategies in the digital era, navigating social and institutional aspects, and leveraging digital technologies like big data and AI (Digital transformation: Rewiring for digital and AI, 2023).

Cloud Transformation Approaches and Tools:

- Various adoption strategies include hybrid, multi-cloud, and cloud-native models. (Bommadevara, Miglio, & Jansen, 2018)
- Different migration strategies like lift-and-shift, re-architecture, hybrid, repurchase, and retire, each with specific pros and cons. (Mahdi Fahmideh Gholami, 2016)

Modernisation and Cloud-native:

- Focus on Amazon Web Services (AWS) and its impact on traditional software operational models, including cloud-native and serverless computing. (Shuiguang Deng, 2023)

- Discuss AWS's impact on business and society, highlighting its use by major companies like Netflix and Adobe. (SafonovaIrina & Bamigbala, 2022)

Kubernetes Study:

- A comprehensive study on Kubernetes emphasises its benefits, such as self-healing containers and challenges, such as the lack of diagnostic and security tools (Leila Abdollahi Vayghan, 2019).
- Identifies 15 distinct challenges related to Kubernetes. (Shazibul Islam Shamim, 2022)

Post-Covid-19 Digital Transformation:

- The pandemic has accelerated digital transformation, emphasising the need for operational excellence and automation.
- Highlights the rise of Citizen Developers and the Low Code/No Code movement. (Khoshafian, 2020)

Digital Transformation in the AI Era:

- The use of LLMs in digital transformation, such as automating digital content creation and providing customer support.
- Highlights the significant investment and adoption of generative AI technologies in various enterprises. (Michael Chui, 2023)

In summary, this literature review encapsulates a diverse range of studies that collectively shed light on the complexities and advancements in digital transformation and cloud computing, underscoring the critical importance of strategic decision-making and innovative solutions in navigating the ever-evolving landscape of technology. Thus, this research highlights the crucial role of LCNC platforms in facilitating efficient digital transformation, offering a valuable resource for organisations aiming to utilise these technologies for improved operational efficiency and strategic development.

3.6 Research Design and Methods

This study methodically outlines a comprehensive approach, combining quantitative and qualitative methodologies to investigate the impact of LCNC platforms on digital transformation. Quantitatively, it involves evaluating multiple characteristics of LCNC platforms, while qualitatively, the study establishes indicators for assessing the viability of adopting various platforms. This multifaceted approach aims to guide executives in making strategic decisions, particularly in weighing the trade-offs between conventional approaches and AI-assisted transformations facilitated by LCNC platforms, providing a detailed blueprint for achieving the research objectives.

3.7 Population and Study Sample

The study includes diverse sources representing cloud transformation's complex nature and Low-code/No-code platforms. The sources encompass the following:

- Industry Reports and Public Data: Esteemed industry reports from Gartner (Oleksandr Matvitskyy, 2023) and Forrester offer valuable insights into market trends, challenges, and forecasts about cloud transformation and Low-code/No-code platforms.
- Professional Consultancy Studies provide research and conclusions from multiple cloud and digital transformation consulting firms. These studies give professional expertise and real-world case studies that demonstrate the problems and solutions involved in cloud transformation.
- Community-generated content refers to data from conversations and contributions made on technical forums such as Stack Overflow and Reddit (Yajing Luo, 2021). These platforms serve as a repository of user-created content, firsthand experiences, and perspectives on Low-code/No-code platforms, offering an authentic and grassroots viewpoint.

- Technical Analysis: A comprehensive assessment of the characteristics and functionalities of leading Low-code and No-code platforms to determine their appropriateness and effectiveness in tackling cloud transformation obstacles.

The study sample is selected from several sources to ensure a thorough comprehension of the subject matter. The selection procedure encompasses the following:

- Selecting Appropriate Reports: Curating Gartner and Forrester's latest and pertinent reports primarily focus on cloud transformation and Low-code/No-code technologies.
- Interacting with Consulting businesses: Establishing partnerships with consulting firms to obtain exclusive research or carry out collaborative research, emphasising practical knowledge and real-life examples.
- Online forum data mining involves extracting and analysing information from online technical communities. This entails discerning pertinent threads and postings on Stack Overflow, Reddit, and comparable platforms to assess the community's perspectives and encounters.
- Technical Assessment: Perform an in-depth evaluation of specific low-code and no-code platforms, with a particular emphasis on their characteristics, functionalities, and appropriateness for different scenarios in cloud transformation.

3.8 Sample Size and Selection of Sample

This study aims to choose a sample that offers comprehensive insights into the correlation between obstacles in cloud transformation and the utilisation of Low-Code and No-Code Platforms (LCNC). The sample size was set by considering both qualitative richness and quantitative representativeness to fully cover essential factors such as high performance, application availability and scalability, disaster recovery, and security.

Selection Criteria

The selection criteria prioritises businesses and persons actively implementing or administrating Low-Code Application Platforms (LCAPs) and No-Code solutions in contexts where cloud transformation plays a significant role. The following items are included:

- Performance and Scalability: Organizations that have utilised LCNC to tackle issues related to performance limitations and the capacity to handle increasing workloads in cloud-based applications.
- Disaster Recovery and Security: Organizations implementing LCAPs to bolster their disaster recovery plans and strengthen their security measures in cloud environments.
- API Integration: Examining instances where LCNC platforms have effectively enabled the seamless data exchange between enterprise and third-party cloud services, emphasising the platforms' ability to integrate various systems.
- Implement LCNC solutions to enhance usage monitoring and ensure compliance with rigorous service-level agreements in cloud environments.
- Support and Training: Analysing the impact of LCNC on improving the accessibility of technical assistance and training for cloud-based applications and reducing the intricacy associated with conventional development environments.

The sample also demonstrates the varying degrees of acceptance and implementation of Low-code, Zero-code, and No-code methodologies, encompassing a broad range of practical applications across diverse sectors and organisations of various scales. This diversity facilitates comprehension of the multiple causes that propel enterprises towards different platforms amidst hurdles in cloud transformation.

Size of the sample

Data from the specified sources from the past 2-4 years is utilised. This study aims to incorporate various entities, including major corporations, small and medium-sized firms, and insights from industry experts and technical architects.

The chosen sample provides a thorough outlook on how obstacles related to cloud transformation impact the transition towards low-code and no-code platforms. It offers significant insights into this developing field's current trends, difficulties, and future directions.

Sources of Data

Data was collected using a comprehensive and versatile approach. This methodology guarantees a thorough comprehension of the subject matter by incorporating several sources, each providing distinct insights and viewpoints.

Choosing Suitable Reports

- Industry Reports: Prominent agencies like Gartner and Forrester were valuable secondary data sources in industry reports. The primary emphasizes on the latest publications discussing the difficulties encountered during cloud migration and the significance of Low-code/No-code solutions. These studies are crucial for comprehending industry trends, technical breakthroughs, and expert forecasts in the sector.

Engaging with Consulting Enterprises

Partnerships and Exclusive Research: An essential element of our data collection process entails establishing collaborations with consulting organisations specialising in cloud transformation and digital innovation. By accessing their exclusive research or participating in collaborative research, we acquire practical insights, case studies, and real-world implementations of Low-code and No-Code platforms. This cooperation aims to

provide insight into how firms effectively address cloud-related difficulties using these developing technologies.

Data mining of online forums

- Community Perspectives: We employed data mining techniques on platforms such as Stack Overflow and Reddit to access the valuable knowledge and expertise exchanged in online technical communities. This entails identifying and analysing debates, queries, and feedback about Low-code and No-code platforms. By analysing these community interactions, we can gain insights into user experiences, obstacles encountered, and the practical functionality of these platforms in real-life situations.

Technical Assessment: - Platform Evaluation: An essential aspect of our primary data collection involved thoroughly evaluating different Low-code and No-code platforms. This study scrutinises these platforms' characteristics, capabilities, and flexibility in various cloud transformation scenarios. Through this approach, we can assess their efficacy, productivity, and possible limitations, offering a thorough perspective on their suitability in multiple situations.

Data integration refers to combining and consolidating data from several sources into a unified and coherent format.

- Consolidating Diverse Data: The data gathered from these many sources were combined to create a comprehensive overview of the present condition and future potential of Low-code and No-code platforms in cloud transformation. This integration facilitates the identification of patterns, discrepancies, and developing trends in the sector.

3.8 Data Collection Procedures

Data Analysis

This research on the impact of Low-code and No-code platforms in cloud transformation extensively utilises social media platforms as a primary source for data collection. These

platforms provide a rich, real-time source of user opinions, experiences, and trends related to Low-code and No-code technologies.

Social Media Platforms

- LinkedIn: As a professional networking site, LinkedIn offers valuable insights from industry experts, tech professionals, and organisations. We was reviewed posts, articles, and discussions tagged with hashtags like #LowCode, #NoCode, #CitizenDevelopment, and #LowCodeRevolution. This helped in understanding the professional and business perspectives on these technologies.

- Reddit: Known for its community-driven content, Reddit hosts numerous technology forums where professionals, enthusiasts, and users share their experiences and views. Subreddits dedicated to programming, software development, and technology trends were monitored. Threads and posts under tags like #LowCodeDevelopment and #CitizenDeveloper were analysed to gain insights into the user community's perspective.

- Twitter: Twitter's fast-paced and concise content provides up-to-the-minute information on industry trends and public opinions. By tracking tweets and threads associated with hashtags such as #LowCode, #NoCode, and #CitizenDeveloper, we helped capture the immediate reactions and short-form opinions of a wide range of users, from industry leaders to casual users.

Data Collection Methodology

- Content Analysis: We employed content analysis to systematically evaluate and interpret the data gathered from these social media platforms. This involved categorising content and identifying prevalent themes, trends, opinions, and sentiments expressed by the users.

- Data Aggregation and Filtering: Using social media analytics tools, we aggregated large volumes of data and filtered them based on relevance, date, and engagement levels. This helped in focusing on the most pertinent and influential content.

- Ethical Considerations: This data collection respects user privacy and platform guidelines that align with ethical research practices. Publicly available data was used, and personal identifiers were anonymised in the research analysis.

Expected Outcomes

- Understanding Public Sentiment: This approach comprehensively views public sentiment and prevailing attitudes towards Low-code and No-code platforms.

Identifying Key Trends and Patterns: Analyzing social media content revealed critical trends, challenges, and patterns in adopting and perceiving these technologies in cloud transformation.

- Diverse Perspectives: Collecting data from multiple platforms ensures a diverse range of perspectives, from technical experts to end-users, offering a holistic understanding of the subject.

Exposure Assessment

This research is centred on the technological dynamics of cloud computing and its influence on adopting low-code and no-code platforms. It primarily addresses the intersection of cloud technology and software development without direct physical, chemical, or biological implications. However, should any repercussions related to Generative AI arise during the research, they were also duly considered and added, even outside the initial research proposal.

Data Management

Data Collection Methods:

- Mixed Approach: Utilizing both quantitative and qualitative data collection techniques.

- Reports Analysis: Curating relevant reports from industry leaders like Gartner and Forrester, focusing on cloud transformation and low-code/no-code technologies.

- Deep dive into various low-code and no-code platforms reviewed by research agencies, including Gartner, Forrester, TrustRadius, G2, Capterra, Everest Group, and Product Hunt. Online Forum Data Mining involves extracting and analysing information from technical communities online, particularly from platforms like Stack Overflow and Reddit.

- Platform Evaluation: Conduct in-depth evaluations of specific low-code and no-code platforms, focusing on their features and suitability for various cloud transformation scenarios.

- Social Media Analysis: Analysing LinkedIn posts, articles, and discussions, as well as tweets and threads, tagged with relevant hashtags to gain insights into community perspectives and immediate reactions.

Data Organization and Storage:

- Storage Format: Storing data in CSV file format.

- Storage Location: Amazon S3 is used for secure and efficient data storage.

- Tools: Employing Python and various EDA (Exploratory Data Analysis) related libraries for data handling and processing.

Data Security and Confidentiality:

- As the data sources are publicly available, significant concerns regarding security and confidentiality are not prevalent. However, this aspect will be revisited and revised as necessary throughout the research process.

Data Processing and Analysis:

- Exploratory Data Analysis: Engaging in content analysis, data aggregation, and filtering.

- Trend Identification: Identifying key trends, patterns, and public sentiment to provide diverse perspectives from technical experts to end-users.

- Comparative and Regression Analysis: Examining the impact of various factors on the subject matter.

- Quantitative and Qualitative Insights: Expecting statistically significant findings in technological skills and gaining insights into perceptions of digital literacy and its impact on employability.

Data Sharing and Accessibility:

- The public nature of the data sources minimises concerns regarding data sharing and accessibility.

Ethical Considerations:

- Using publicly available data minimises ethical concerns, but adherence to ethical standards was maintained.

Long-term Data Management:

- Post-research, the plan is to responsibly discard the CSV files stored on Amazon S3.

Compliance with Legal and Institutional Policies:

- The public availability of data sources simplifies compliance with legal and institutional policies, with a commitment to address any unforeseen issues as they arise.

3.9 Data Analysis Strategies

The research's Data Analysis Strategies encompass a comprehensive and methodical approach, integrating various techniques and tools to derive meaningful insights from the collected data.

The initial analysis phase involves fundamental Exploratory Data Analysis (EDA) using Python and leveraging modules such as Pandas NumPy. The focus was examining the datasets to condense and outline their primary attributes, often using visual techniques. This critical stage includes data cleansing, pattern recognition, anomaly detection,

hypothesis testing, and assumption verification to ensure a solid foundation for further analysis.

A qualitative content analysis was conducted for the textual material extracted from internet forums, social media platforms, and reports. This involves utilising coding methodologies to classify and analyse the data to recognise themes, patterns, and trends.

Trend Identification and Sentiment Analysis also play crucial roles. Sentiment analysis algorithms were applied to social media postings and forum threads to assess public sentiment towards low-code/no-code technologies. This process was automated using natural language processing (NLP) tools using Python NLTK, providing a nuanced view of public opinion.

Integrating quantitative and qualitative data was critical to obtaining a thorough understanding of the research topics. This process involved triangulating the findings from both types of analysis, analysing, and combining findings to derive significant conclusions regarding the influence and ramifications of low-code/no-code technologies in cloud transformation.

Ethics and Human Subjects Issues

All personally identifiable information and inappropriate language found within the curated social media data were rigorously anonymised to respect privacy and maintain the integrity of the research.

Search process

This research is outlined as a five-step process.

- Curate Data: This phase involves collecting data from various sources, such as social media hashtags and forums like Stack Overflow and Reddit, and utilising reviews and ratings (Gartner, 2023) from enterprise platforms to narrow down the top 50 platforms. Findings from professional consultancy are used to clean the data and focus on critical areas.

- Analyse Data: In this step, exploratory data analysis is conducted on the curated data to identify trends, patterns, and public sentiment.
- Identify Key Metrics: Comparative and regression analyses are performed to define quantitative and qualitative insights. This helps determine critical parameters and metrics documented in specific report chapters.
- Visualize Data: This stage focuses on studying critical metrics and using visualisations like charts and graphs to present findings. It also includes refining earlier chapters of the report based on these visualisations.
- Findings and Conclusions: The final phase involves refining the documentation of findings, discussing implications, and formulating recommendations and conclusions, which are included in the last chapters of the report. To finalise the report, a thorough review of all chapters is conducted.

This comprehensive approach ensures a robust and systematic progression from data collection to analysis, visualisation, and the presentation of findings, culminating in a well-documented and conclusive research report.

3.10 Platform Assessment criteria

Evaluating low-code and no-code platforms is based on 31 key metrics, each rated on a scale from 0 to 5. These are categorised into three metric types:

- Core Metrics: 15 essential attributes that define baseline platform capabilities.
- Extended Metrics: 7 additional attributes providing advanced functionality and flexibility.
- AI-based Metrics: 9 metrics reflecting the integration of AI capabilities into the platform.

4.11 Conclusion

Strengths: This study's primary strength lies in its comprehensive exploration of cloud transformation approaches, challenges, and the role of Low-Code No-Code (LCNC)

platforms in enhancing business process efficiency and integrating AI with LCNC platforms, particularly with the advancements in large language models (Mark Chen, 2021) and the Transformer model architecture, positions the study at the forefront of understanding digital transformation's future.

Weaknesses: One potential weakness could be the rapid pace of technological advancement in AI and cloud computing, leading to some findings needing to be updated. Additionally, the study may face challenges in quantifying the precise impact of LCNC platforms across diverse industries due to the varying nature of digital transformation initiatives.

This research holds significant importance as it addresses the crucial role of cloud transformation in the broader context of digital transformation initiatives. By focusing on the efficiency and effectiveness of Low-code No-code (LCNC) platforms, the study contributes to a deeper understanding of how these platforms can reduce operational expenditures (OPEX) and improve decision-making processes. The findings are expected to aid organisations in navigating the complexities of digital transformation, particularly in optimising workloads and reimagining customer and employee experiences through advanced AI-integrated solutions.

Thus, this research highlights the pivotal role of LCNC platforms in streamlining digital transformation and serves as a valuable resource for organisations seeking to leverage these technologies for enhanced operational efficiency and strategic growth.

The motivation behind this research stems from the need to understand the evolving landscape of digital transformation, where cloud transformation has become an essential precursor. The study evaluates how LCNC platforms and AI technologies can revolutionise application development and digital transformation. By investigating the potential of LCNC platforms to streamline operations and foster more agile, intelligent, and customer-

centric business practices, the research aims to provide actionable insights and recommendations for organisations embarking on their digital transformation journeys.

CHAPTER IV:

RESULTS

Overview

This extensive study evaluates over 300 platforms for citizen developers, including low-code, no-code, and related platforms, such as workflow automation, integration, analytics, CMS, e-commerce, website builders, identity solutions, AI-code generation, No-Code AI, and over 100 AI startups.

Platforms	#
Low-code No-code	123
Related Platforms	#
Integration and Workflow Automation	51
Work and Collaboration	25
No-Code AI	21
UI Design	17
Website Builder	12
Identity	11
CMS	10
Analytics	9
eCommerce	9
Domain-specific	5
Search and Recommendation	4
AI-Code Generation	4

Unified Business Platforms	4
CRM	2
Services and Training	1
DevSecOps	1
AI Startups	106
Total	415

Table 4.1 Platforms reviewed in this study

Note: All the platform listing and related data are available at https://research.amitpuri.com.

The following insights are derived from a review of over 2,000 posts and reviews across various social media platforms, market research bodies, and consulting firms, as well as 800+ case studies, extensive documentation, tutorials, YouTube channels, forums, and blog posts:

- By utilising visual user interfaces with drag-and-drop support for application creation, these platforms assert that they can accelerate development by up to 20-30 times or reduce time to market by 70-80% compared to traditional methods.
- Applications can be developed responsively and deployed across any web browser or device.
- The number of applications developed with the same budget on these platforms nearly doubles.
- To facilitate rapid deployment, these platforms provide approximately 2K pre-built templates, connections with over 5-7K integrations in various platforms, in-app purchases, significant payment gateways, and a component marketplace for extensibility.

- Workflow automation and integration are available across assessed platforms, ensuring that business processes are efficiently automated.
- Most platforms are designed for small businesses and venture incubation, while a small number are specifically designed for large and mid-sized companies.
- These low-code, no-code platforms offer pricing information for various plans and their coverage. A few of these platforms offer ROI or TCO calculators, which are extremely basic for calculating savings development efforts.
- These platforms facilitate data exposure by integrating RESTful APIs, webhooks for real-time data synchronisation, social media, and analytics to improve data coverage and analysis.
- Native CI/CD support and real-time collaboration and communication within the development cycle facilitate governance at scale.
- Most social media posts focus on customer testimonies, promotional content, seminars and events, new features or releases, and the adoption of modern trends such as Gen AI.

Case studies review

The case studies available on these platforms emphasised trends like citizen development, cost savings, productivity improvements, and legacy system modernisation. The review of 800+ case studies highlights key trends and the impact of various strategies on development, productivity, and cost-saving initiatives. The findings emphasise a strong focus on citizen development, modernisation efforts, and significant reductions in operational inefficiencies. Here are the core insights from the analysis:

- Citizen Development:
    - Mentioned 183 times across 58 case studies.

- Organizations improve efficiency by empowering non-technical users to create solutions.

- Development and Deployment Efficiency:
  - Cited 171 times across 43 case studies.
  - Streamlined processes reduce time-to-market, improving speed in development and deployment.

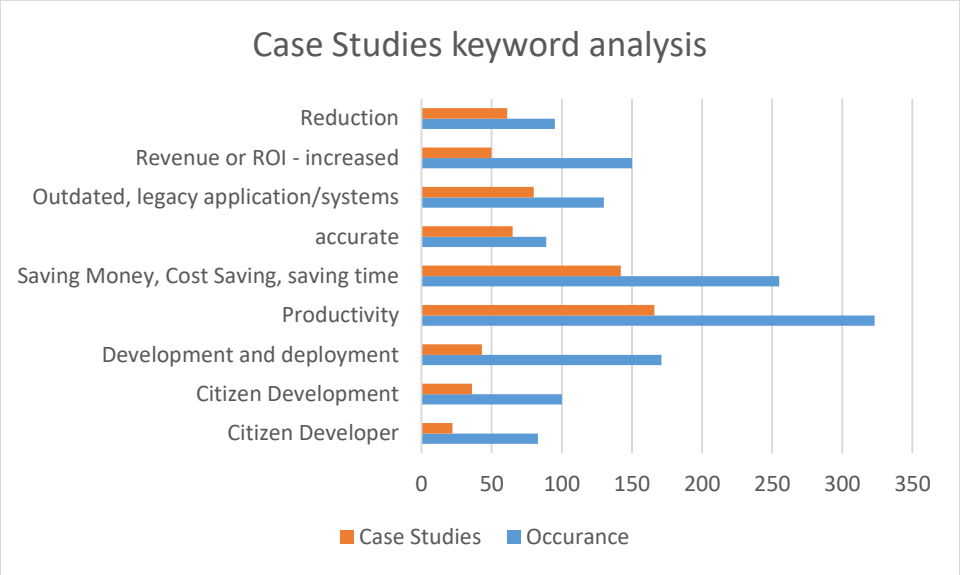


Figure 4.1 Keyword analysis of the 800+ case studies of Low-code, No-code platforms

- Productivity:
  - Referenced 323 times across 166 case studies.
  - Enhancements include streamlined workflows, automation, and better resource allocation.
- Cost Savings and Time Efficiency:
  - Mentioned 255 times across 142 case studies.

    - Organisations experience both immediate and long-term financial benefits from technology deployment.
- Accuracy and Legacy System Modernization:
    - Accuracy: Mentioned 89 times.
    - Legacy Modernization: Cited 130 times across 80 case studies, allowing companies to meet current demands more effectively.
- Reductions in Time, Cost, and Errors:
    - 95 occurrences across 61 case studies.
    - Reduced development time, cost, and errors, with significant savings and improvements reported.
- Revenue Growth and ROI:
    - Cited 150 times across 50 case studies.
    - Organizations report increased revenue and ROI from implemented strategies.

These case studies, accessible on the websites of these platforms, illustrate how organisations use automation, modernised systems, and citizen development to achieve remarkable efficiency improvements, productivity gains, and cost savings. Businesses are reducing expenses and improving their capacity to innovate and scale rapidly by reducing development time, operational costs, and error rates.

These case studies delineate the value propositions of agility, speed, and reduced development costs. Nevertheless, they do not explicitly link these results to the rationalisation of cloud workloads. This is likely since low-code and no-code platforms, which are PaaS or SaaS solutions, provide numerous advantages. These platforms necessitate additional analysis and are discussed in the subsequent sections on cloud resource utilisation and cost efficiency.

The adoption of Generative AI is another trend identified through the review of these platforms. The GenAI bubble and the success of OpenAI have made generative AI ubiquitous on most platforms in recent years. Currently, these platforms comprise the following experiences:

Get started experience: Conversational assistants produce visual user interfaces by refining visuals, establishing connections with integration, refining data retrieval, and defining actions to accomplish app objectives in response to user input (the prompt).

Help and Docs experience: These chat agents eliminate the need to consult documentation when developing applications, minimise coding errors, and collaborate as co-pilots to construct applications by utilising low-code platforms, a form of peer programming that improves productivity.

Developer's experience: Citizen developers are provided with a unique user experience through chat interactions, which enables them to develop the solution.

End-user experience: Citizen developers can also construct this interface to enhance the efficiency of the end user in completing their tasks. End users can query business data and perform complex business workflows using chat-based interfaces.

These experiences—spanning getting started, help and docs, and both developer and end-user interfaces—are unlocking new possibilities in the digital landscape. By focusing on the expectations of new generations and delivering a cutting-edge user experience, these platforms are poised to revolutionise how developers and end users interact with technology. Adopting Generative AI is not just a trend but a game changer that can fundamentally reshape productivity, collaboration, and user engagement in enterprise solutions. Platforms embracing this evolution are set to provide innovative, chat-based interfaces that meet the demands of Gen Z and Gen Alpha, transforming how business and development are conducted.

Based on available data, these platforms have generated 3-4 million applications, serving 1.5K to 1 billion monthly users. Adopting such technologies across industries—banking, finance, insurance, healthcare, life sciences, and the public sector—hinges on risk, safety, regulatory compliance, and governance surrounding Generative AI. While these innovations have significantly impacted creativity, scalability, and efficiency, a long journey remains. Continued advancements in GenAI will further push the boundaries of what is possible, offering exciting opportunities for businesses and developers.

4.1 Research Question One

RQ1 - Are low-code and no-code platforms a viable alternative to rationalising cloud workloads to reduce costs on cloud services?

We will approach this question by focusing on cost efficiency, scalability, development speed, complexity management, and resource utilisation. Cloud service costs, encompassing direct and indirect expenses, are generally categorised into three primary areas: compute and storage, data transfer and networking, and monitoring and management, along with labour costs. For compute costs, right-sizing resources in Infrastructure-as-a-Service (IaaS) environments is crucial to avoid overspending. This practice is directly tied to the performance efficiency pillar of the Well-Architected Framework. Leveraging reserved instances or savings plans can enhance long-term financial efficiency by locking in lower rates. However, moving from IaaS to Platform-as-a-Service (PaaS) or Software-as-a-Service (SaaS) can provide better cost efficiency. These models take advantage of consumption-based pricing or feature/user-based licensing, enabling businesses to scale operations cost-effectively.

In terms of storage, understanding data usage patterns is essential for cost optimisation. Organisations can align their storage needs with different pricing levels by classifying data into hot, warm, or cold storage tiers, thus reducing costs and, additionally, selecting the appropriate database model—whether a Relational Database Management System (RDBMS), NoSQL, or Blob storage—based on the type and frequency of data access can further streamline expenses. Data transfer costs, which can be one-time during migration or ongoing in hybrid cloud environments, should also be carefully planned. Implementing strategies such as using dedicated network paths for large migrations and optimising data flows in distributed architectures can help minimise these costs.

Networking expenses typically arise from establishing and maintaining internal and external networks, alongside implementing necessary security architectures. To control these costs, businesses should be mindful of inter-region or inter-zone traffic charges, particularly in hybrid or multi-cloud setups. Monitoring and management costs apply across all cloud models—whether IaaS, PaaS, or SaaS—and depend on the selected pricing tier. While some levels of monitoring, diagnostics, and security are included, advanced services for compliance and governance often come at an additional cost. Reviewing native cloud monitoring tools, such as AWS CloudWatch or Azure Monitor, against third-party solutions can ensure cost efficiency without compromising oversight.

If significant cloud spending remains after implementing these optimisation strategies, exploring low-code and no-code platforms may offer additional cost savings. Low-code and no-code platforms empower non-technical users to build applications quickly, reducing the need for specialised development resources. By shifting development workloads to these platforms, businesses can streamline processes, improve agility, and potentially reduce the financial burden associated with traditional software development and maintenance. Low-code and no-code solutions provide a flexible, scalable option for

businesses seeking further cost reductions in cloud expenditures. While low-code and no-code solutions have shown promise for reducing upfront costs, their long-term cost-effectiveness remains critical. Can embedding these platforms into long-term strategic investments prevent cost overruns, or do they risk escalating costs over time due to scalability limitations, hidden fees, or dependency on proprietary ecosystems?

Estimating, managing, and controlling costs is one of the most challenging aspects of cloud transformation projects, yet it is crucial for ensuring the cost-effectiveness of cloud adoption. Several key factors influence the estimation and calculation of cloud spending, and addressing these is essential before exploring cost-efficiency strategies. This study examines small, medium, large, and enterprise organisations to analyse workload patterns, review budgets, account for labour and non-labour costs, and conduct a cost-benefit analysis. To facilitate this, we developed a Total Cost of Ownership (TCO) calculator and carried out the following steps:

- Selected the organisation size and defined the application portfolio based on complexity and usage patterns, categorising them into low, medium, or high complexity, including mission-critical applications.
- Reviewed cost categories and configured budgets by allocating IT budget percentages to each.
- Defined workload patterns, labour costs, and the appropriate team size and composition.
- Conducted a comprehensive cost-benefit analysis to assess financial efficiency.

To gain clarity, calculate the total cost of ownership (TCO) step-by-step to analyse the cost-benefit of infusing a low-code/no-code platform into your software ecosystem compared to current cloud workloads in IaaS, PaaS, and SaaS. Budgeting and

forecasting are critical FinOps principles for cost efficiency, helping to predict future spending trends based on historical data, define cost alerts and limits to avoid cost overruns and maintain operational excellence and reliability. Ultimately, a strategic approach to cost management will determine whether these platforms truly deliver long-term value.

TCO Calculator location - https://research.amitpuri.com/calculator/tco

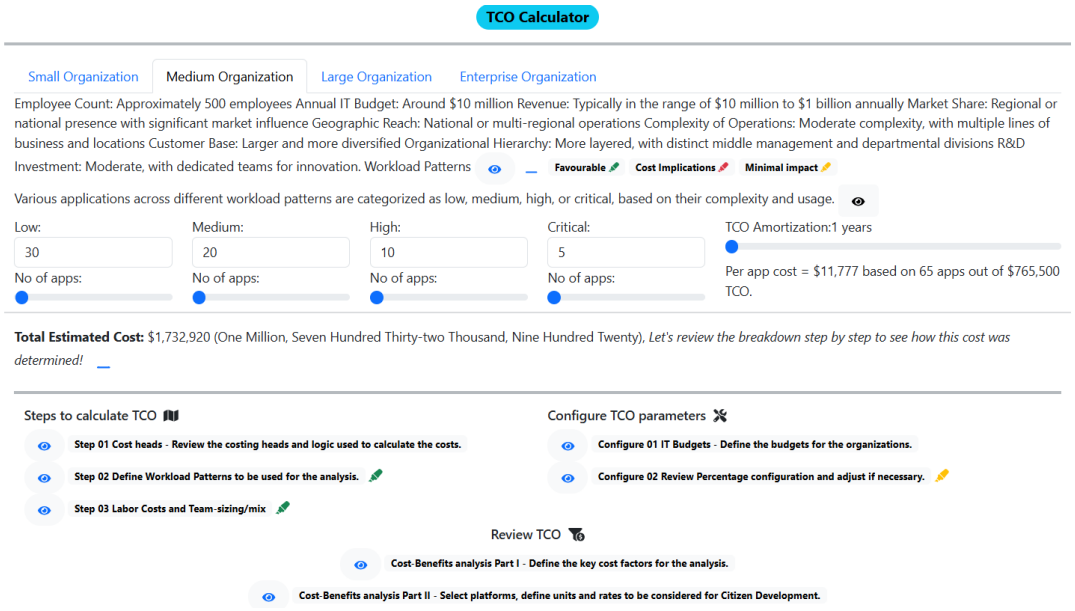


Figure 4.2 TCO Calculator

So, step 01 includes defining various cost heads of the IT budgets that are categorised into

- Costs incurred on Human Capital and Development
- Costs of Automation and Efficiency tools
- Strategic CapEx Investments
- Operational Agility Costs

The annual IT budget often serves as the overarching budget, encompassing all other budgets. It is expressed as a percentage of total revenue, ranging from 3% to 7% for

most industries and may be higher for tech-heavy sectors. Refer to the tables below to review the budget breakdown.

Budget percentages											
Org. Size	**Revenue**	**IT**	**Salary**	**Operational**	**Cloud**	**Capital**	**Dev**	**Automation**	**Contingency Funds**	**Project**	**Agile Ratio within Project**
Enterprise	1000000000	60%	65%	15%	30%	20%	10%	5%	10%	10%	0.5
Large	100000000	20%	60%	15%	30%	20%	10%	5%	25%	10%	0.5
Medium	10000000	12%	55%	15%	20%	15%	10%	5%	10%	10%	0.5
Small	1000000	7%	50%	10%	40%	15%	10%	5%	10%	10%	0.5
dependent		Revenue		IT Budgets							
				Human Capital & Development		Automation & Efficiency Tools		Strategic CapEx Investments		Operational Agility Costs	

Table 4.2 Various percentages of budgets within IT

Further, IT budgets for small, medium, large, and enterprise organisations are broken down as shown in Table 4.3.

	Human Capital & Development	Automation & Efficiency Tools	Strategic CapEx Investments	Operational Agility Costs
Enterprise	~30% to 40%	~20% to 25%	~30% to 40%	~15% to 20%

Large	~25% to 35%	~15% to 20%	~20% to 30%	~10% to 15%
Medium	~20% to 30%	~10% to 15%	~15% to 20%	~5% to 10%
Small	~20%	~5% to 10%	~ 10% to 15%	~Dev Budget

Table 4.3 The 4 Major Cost Heads in the IT Budget

Each cost head in the IT budget is further classified into tables 4.4, 4.5, 4.6 and 4.7.

Human Capital & Development	%	Dependent variable
Skilled Labor		
Salaries and Benefits	40-60%	Total cloud budget
Hiring Costs	10-20%	1-yr salary X positions
Talent & Skills Development		
Training and Certifications	2-5%	Salary budget
Workshops & Seminars	1-2%	Operational budget
Mentorship and Coaching	1-3%	HR/IT budget
Agile Workforce Management		
Agile Training	1-2%	Project budget
Change Management	5-10%	Project budget
Collaboration Tools	1-3%	IT budget

Table 4.4 The Human Capital & Development in the IT Budget

Automation & Efficiency Tools	%	Dependent variable
CI/CD Pipelines		
CI/CD Tools	5-10%	Development budget
Automation and Scripts	10-20%	Development budget
DevOps Tools		
Licenses & Subscriptions	5-15%	IT budget

Integration Costs	10-20%	Tool acquisition costs
Infrastructure as Code (IaC)		
IaC Tools	5-10%	Cloud infrastructure budget
Setup and Configuration	10-15%	Initial IaC tool costs
Robotic Process Automation (RPA)		
RPA Licenses	15-20%	Automation budget
Bot Development & Maintenance	10-15%	Automation budget

Table 4.5 The Automation & Efficiency Tools in the IT Budget

Strategic CapEx Investments	%	Dependent variable
Digital Acceleration		
Platform & Tool Acquisition	10-20%	CapEx budget
Implementation and Deployment	20-30%	Total acquisition cost
Subscription		
Long-term Subscriptions	5-15%	IT budget
Subscription Fees	102-0%	IT budget
Platform & Tooling Acquisition		
Customization & Integration	20-30%	Platform acquisition cost
Consulting & Expert Services	10-15%	CapEx budget

Table 4.6 The Strategic CapEx Investments in the IT Budget

Operational Agility Costs	%	Dependent variable
Cloud Operating Costs		
Compute & Storage	30-50%	Cloud Budget
Data Transfer & Networking	5-10%	Cloud Budget
Monitoring & Management	5-10%	Cloud Budget

Agile Process Optimisation		
Agile Tooling	10-30%	Operational Budget
Process Improvement	5-10%	Agile Project Budget
Scalability & Flexibility		
Resource Optimization	10-30%	Cloud Costs
Security & Compliance	5-15%	Cloud Budget

Table 4.7 The Operational Agility Costs in the IT Budget

This calculation can help determine the total estimated cost, as shown in Figure 4.2. It can also estimate the per-app cost by considering the total number of apps. However, a more accurate approach would be to define workload patterns based on how applications are deployed across various cloud providers. This includes app services and their rates—basic and optimal—for compute resources; database services and their rates—basic and optimal—on PaaS or VMs on IaaS; and networking, storage, and monitoring services, with their corresponding rates. Additionally, considering serverless architecture, the number of executions (in millions) and their rates—basic and optimal—can help determine a more realistic range of cloud spending, as covered in step 02, refer to figure 4.3 below.

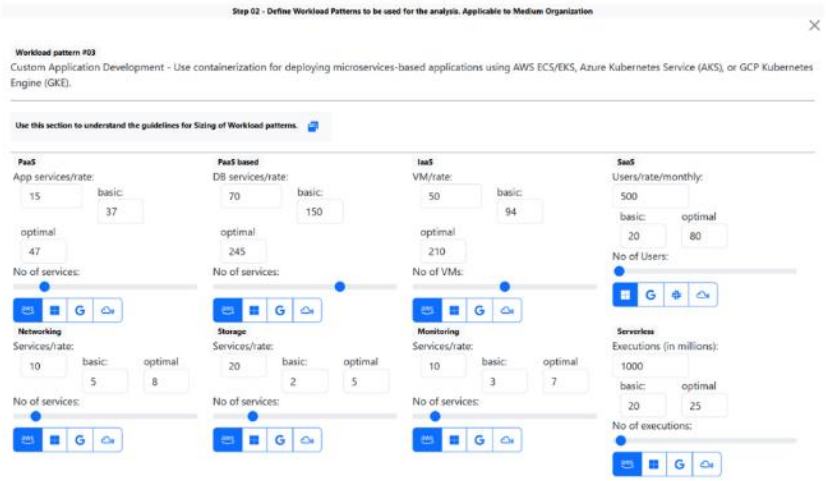
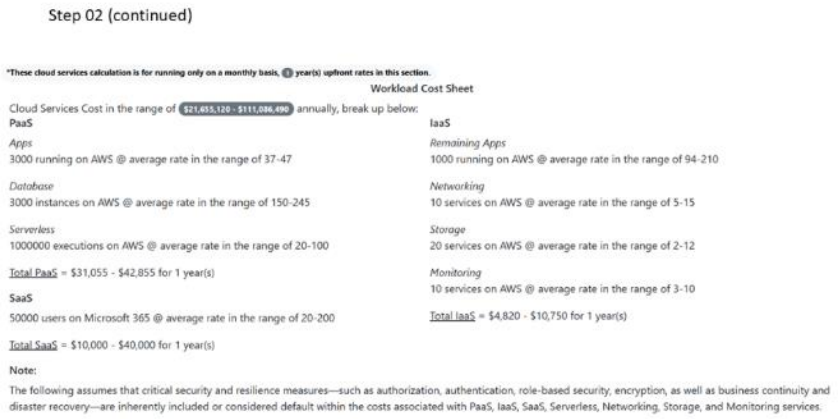


Figure 4.3 Step 02 – Define Workload Patterns

Based on the workload definition from step 02, the workload cost sheet outlines the total spending on IaaS, PaaS, and SaaS for the given workloads across various applications in a medium-sized organisation, as illustrated in Figure 4.4.



Step 02 (continued)

*These cloud services calculation is for running only on a monthly basis, 1 year(s) upfront rates in this section.

Workload Cost Sheet

Cloud Services Cost in the range of $21,653,120 - $111,086,490 annually, break up below:

PaaS

Apps
3000 running on AWS @ average rate in the range of 37-47

Database
3000 instances on AWS @ average rate in the range of 150-245

Serverless
1000000 executions on AWS @ average rate in the range of 20-100

Total PaaS = $31,055 - $42,855 for 1 year(s)

SaaS

50000 users on Microsoft 365 @ average rate in the range of 20-200

Total SaaS = $10,000 - $40,000 for 1 year(s)

IaaS

Remaining Apps
1000 running on AWS @ average rate in the range of 94-210

Networking
10 services on AWS @ average rate in the range of 5-15

Storage
20 services on AWS @ average rate in the range of 2-12

Monitoring
10 services on AWS @ average rate in the range of 3-10

Total IaaS = $4,820 - $10,750 for 1 year(s)

Note:

The following assumes that critical security and resilience measures—such as authorization, authentication, role-based security, encryption, as well as business continuity and disaster recovery—are inherently included or considered default within the costs associated with PaaS, IaaS, SaaS, Serverless, Networking, Storage, and Monitoring services.

Figure 4.4 Step 02 (continued) – Workload Cost Sheet

This addresses the non-labour costs, while step 03 will help define labour costs, team sizing, and role mix to calculate outsourcing and consulting services costs—whether offshore, nearshore, or onshore—based on the number of people in various roles and the hours they contribute over a specified period, typically annually. These efforts should encompass the development, migration, integration, testing (including security, penetration, and performance testing), project management, change management, risk and budget management, stakeholder communication, post-migration review, cloud architecture, design and strategy, vendor and tool selection, cloud provider evaluation, implementation, contract negotiation, compliance, legal and regulatory reviews, and organisational cultural change.

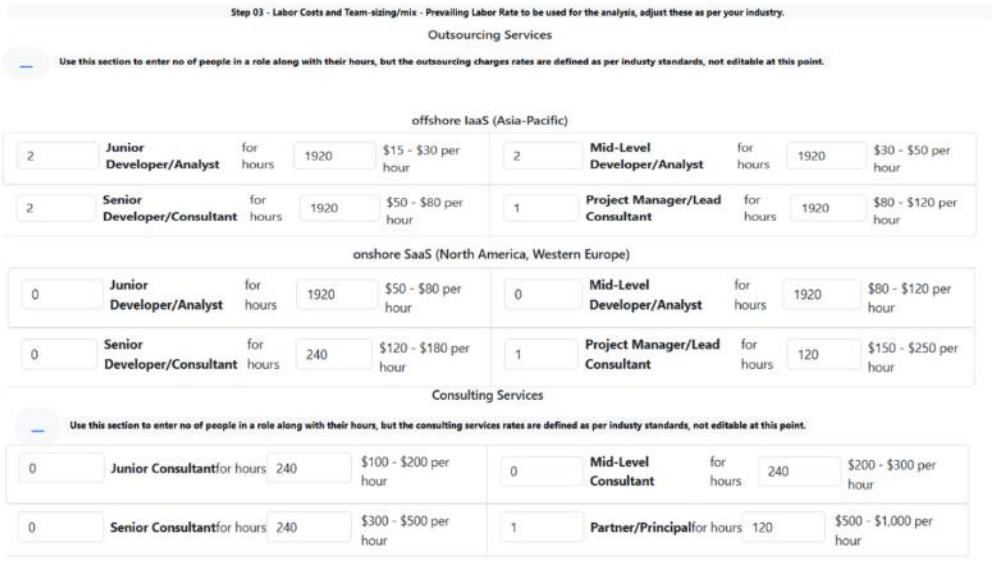


Figure 4.5 Step 03 – Labor Costs and Team-sizing/mix

Figure 4.6 Step 03b – Team sizing/mix summary and total labour cost

With this step, the TCO calculator now includes information on the total labour costs for IaaS, PaaS, and SaaS spending based on industry-standard rates, which can be leveraged to perform a comprehensive cost-benefit analysis. Part I of the cost-benefit analysis enables you to input various costs for IaaS, PaaS, SaaS, or Low-code/No-code

platforms to review the total cost of ownership. The cloud-related cost ranges calculated from steps 02 and 03 are referenced, allowing you to input the actual costs using a pointer.

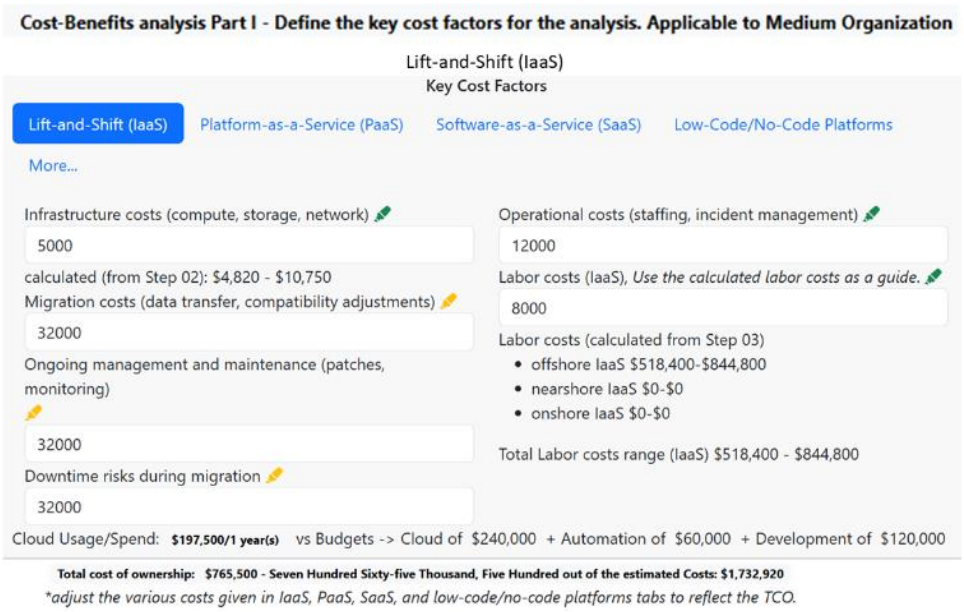


Figure 4.7 Cost-Benefits Analysis Part I – define the key cost factors – IaaS

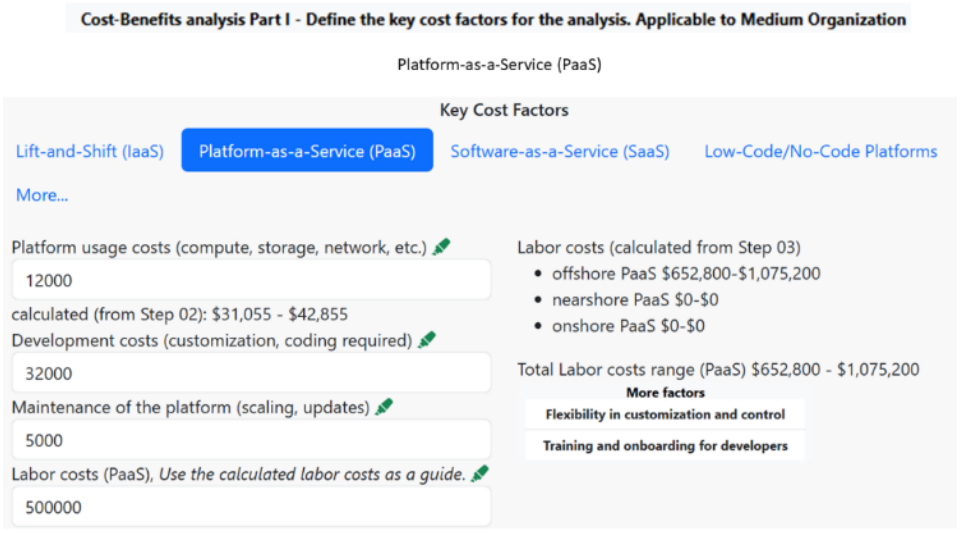


Figure 4.8 Cost-Benefits Analysis Part I – define the key cost factors - PaaS

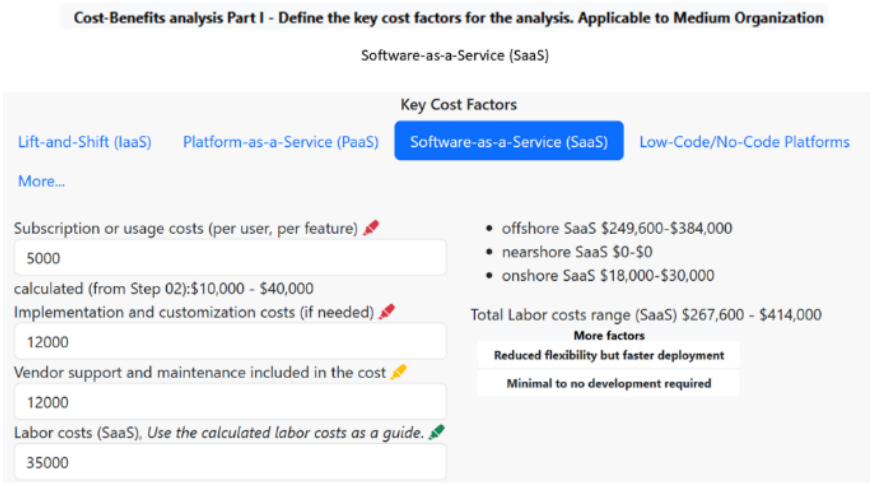


Figure 4.9 Cost-Benefits Analysis Part I – define the key cost factors - SaaS

In the low-code/no-code platforms section, it is essential to recognise that many platforms and services—such as integration and workflow automation, analytics, and CRM—may already fall under the SaaS or low-code/no-code solutions category. While these platforms may not directly support citizen development, they often provide built-in integration and workflow automation capabilities within existing cloud workloads. Therefore, these platforms should be considered within this section or classified under SaaS, as they contribute to automation and efficiency in cloud environments.

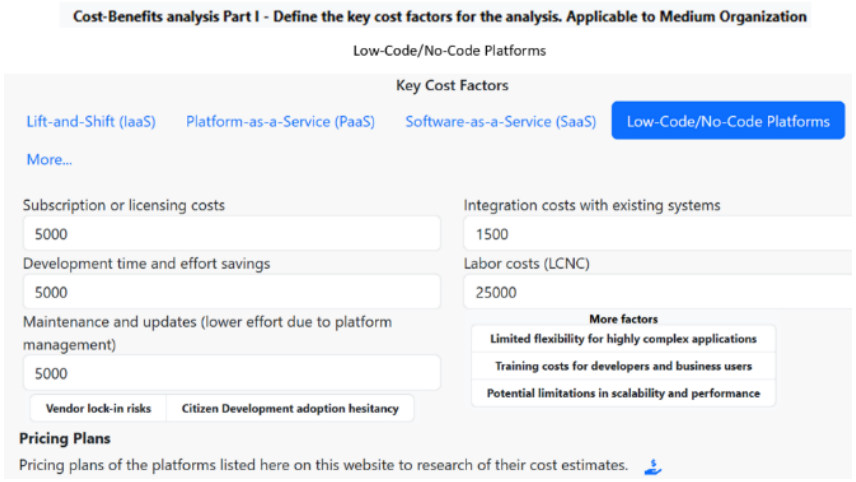


Figure 4.10 Cost-Benefits Analysis Part I – define the key cost factors - LCNC

The final step in the cost-benefit analysis (Part II) is to select the platforms considered for citizen development. In this step, choose the platform and input the required units and rates to calculate the total subscription or licensing costs for replacing applications in the low, medium, high, or critical categories. Subsequently, the costs associated with these applications should be removed from the IaaS, PaaS, or SaaS estimates. Finally, the total subscription or licensing cost is used to assess the impact on the overall total cost of ownership.

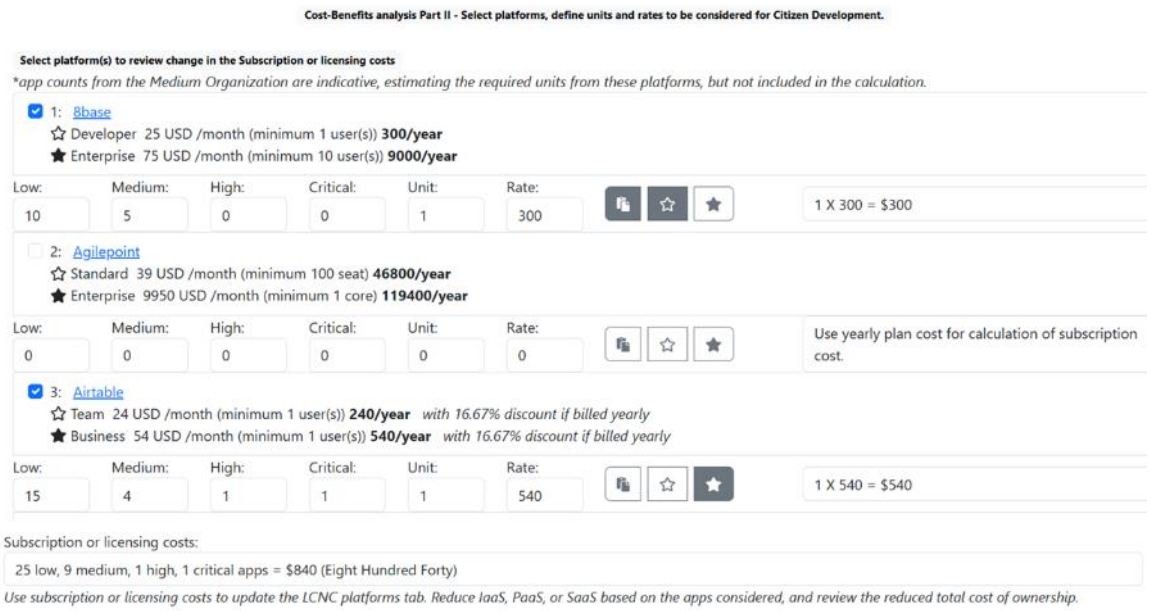


Figure 4.11 Cost-Benefits Analysis Part II – platform, unit, & rates info, review TCO

Review development costs in terms of time and effort savings in the low-code and no-code platforms section, focusing on cost reductions in computing, storage, networking, monitoring, and management. Additionally, account for maintenance savings and accelerated time-to-market benefits where applicable.

Low-code, No-code platforms offer substantial cost savings across several areas, including development time, labour costs, maintenance, infrastructure and computing, storage, networking, monitoring, management, software and licensing, time-to-market, integration, security, compliance, and testing. By reducing the need for extensive development, training, upskilling, and manual processes, low-code and no-code platforms enable organisations to optimise efficiency while minimising costs.

Features like pre-built connectors, automated testing tools, and reduced debugging and error handling further enhance cost-effectiveness. Incorporating these savings into a total cost of ownership (TCO) calculator allows for a thorough review of the cost-benefit analysis, providing a clearer picture of potential gains from adopting low-code and no-code platforms.

Platform	Plan	$ Cost/year/minimum users
8base	Developer plan/user	300
	Enterprise plan/10 users	9000
Agilepoint	Standard edition/100 users	46800
	Enterprise edition/user	119400
Airtable	Team plan/user	240
	Business plan/user	540
Alpha Anywhere	Professional plan/user	1188
	Business plan/user	9000
Appsmith	Business plan/user	480
AuraQuantic	Starter plan/20 users	7440
	Standard plan/20 users	10560
OutSystems	Developer Cloud(ODC)/user	36300

Table 4.8 Low-code No-code Platforms Annual Cost

The table shows that Low-code and No-code platforms offer different pricing structures, ranging from basic developer plans to more comprehensive enterprise editions.

While these platforms provide significant savings in development time, labour, and maintenance, it is essential to acknowledge that there are no 'free lunches'—costs can accumulate, especially at scale. For example, Prices range from $240 per user for Airtable's Team plan to $119,400 annually for Agilepoint's Enterprise edition.

Organisations must carefully evaluate each plan's features and pricing to meet their needs. These platforms offer free or trial plans, allowing businesses to test capabilities before committing. Making an informed decision by thoroughly assessing total costs, platform features, and critical aspects like security and compliance (FedRAMP, HIPAA, SOC1-3, SLAs with five to six nines of availability, etc.) is essential for optimising cost efficiency and ensuring platform effectiveness.

Let us examine an illustration of medium-sized organisations re-platforming a sample set of cloud workloads.

Cost Head	Range	
Cloud Usage (step 02)	$45,875	$93,605
Labour Costs (step 03)	$1,498,800	$2,454,000
Other costs		
IaaS		
Migration costs		$32000
Ongoing management and maintenance		$32000
Downtime risks during migration		$32000
Operational costs (staffing, incident management)		$12000
PaaS		

Development costs (customisation, coding, testing)	$32000
Maintenance of the platform (scaling, updates)	$5000
SaaS	
Subscription or usage costs (per user, per feature)	$5000
Implementation and customisation cost (if needed)	$12000
Vendor support and maintenance (included in the cost)	$12000
Existing Low-code / No-code platforms	
Integration costs with existing systems	$1500
Total Cost of Ownership	$17,20,175 - $27,23,105 for a year

Table 4.8 Existing costs of the cloud workloads in Medium-sized organisation

Considering the cloud workload size of medium-sized organisations, as in Table 4.9

Low	30
Medium	20
High	10
Critical	5
Total	65

Table 4.9 Sample App Size of medium-sized organisation

The per-app cost may be around $11,777.

Assumption: The re-platforming involves 10 low-complexity apps, 20 medium-complexity apps, and 5 high-complexity apps migrating to these platforms:

Airtable	Business plan/user	$540
OutSystems	Developer Cloud(ODC)/user	$36,300

Agilepoint	Enterprise edition/user	$1,19,400
Subscription or licensing costs for 35 apps		$1,56,240

Table 4.10 Platform subscription or licensing costs

The re-platforming efforts are shown in Table 4.11

App Complexity	No. of Apps	Re-platforming Labour Costs (per app)
Low Complexity	10	1 x 160 hrs. @ $80/hour = $12,800
Medium Complexity	20	2 x 320 hrs. @ $120/hour = $76,800
High Complexity	5	2 x 160 hrs. @ $120/hour = $38,400
Total	35	$1,28,000

Table 4.11 Re-platforming Labour costs

The cost savings in this effort are shown in Table 4.12

The total new cost of 35 apps	$2,84,240
vs initial cost of 35 apps	$4,12,196
Cost Savings	$1,27,955

Table 4.12 Cost savings

If we aggregate these cost savings across the key budget categories—Human Capital and development, Automation and efficiency Tools, Strategic CapEx Investments, and Operational Agility Costs in IT budgets—we can expect a positive impact on various sub-cost areas within each. This includes potential reductions in workforce requirements for development, optimised tooling costs, more efficient capital allocation toward scalable technologies, and enhanced flexibility in operational expenses, all contributing to overall cost management and resource efficiency. However, this process may not be entirely straightforward, as re-platforming efforts can lead to additional expenses, including new

UX design, usability testing, user training and communication, ongoing maintenance, and potential upgrade costs if the platform mandates updates.

In summary, Low-code, No-code platforms can offer a cost-effective alternative to traditional cloud models like IaaS or PaaS. However, it is crucial to carefully review your specific needs against the features available on these platforms. Weigh these considerations over a 3-or 5-year period to fully understand their long-term impact on the total cost of ownership. Once investments in strategy and development are made, they can be challenging to reverse, especially given the vendor-specific nature of many platforms. Not carefully aligned with business goals can lead to inefficiencies or waste.

To mitigate these risks, use free trials or limited-time offers to evaluate the platform's suitability and consider a phased adoption strategy—starting with more minor, non-critical applications to assess strengths and weaknesses. Ensure the platform provides robust security, compliance features, and scalability to meet long-term requirements. Performing a thorough cost-benefit analysis, factoring in both initial and ongoing costs, will help optimise the value of your Low-code/No-code investment over time.

The first research question (RQ1) examines the core theory that low-code and no-code platforms can significantly reduce costs by accelerating development and simplifying processes. The findings highlight several advantages:

- Low-code No-code platforms enable faster application development by utilising pre-built components and drag-and-drop interfaces, reducing reliance on traditional coding.
- These platforms reduce the need for specialised development teams by empowering non-technical users to create applications.

However, there are potential drawbacks to consider:

- Learning curve: Despite being designed for non-technical users, transitioning from traditional development methods may present a learning curve.

- Complex Customizations: Low-code No-code platforms may lack the flexibility required for highly complex applications, making traditional development more appropriate in the long run. While low-code and no-code platforms are great for rapid growth, they may face challenges scaling large enterprises with intricate systems.

In parallel, cloud workload rationalisation involves assessing applications and services across cloud, on-premise, and hybrid environments to optimise costs, rightsizing, and performance. Businesses can modernise workloads while ensuring governance and management to streamline operations through refactoring, re-platforming, consolidating, and rehosting. These initiatives strategically align with business objectives, supporting scalability, development, and innovation for the future.

Note: These cost drivers and cloud expenditures mentioned in this section are based on insights from reputable sources, including Bersin, CompTIA, Flexera, Forbes, Forrester, Gartner, IDC, McKinsey, Prosci, RightScale, Scaled Agile, and SHRM.

4.2 Research Question Two

RQ2 - What are the possible scenarios, specific use cases, or application types in which organisations looking to optimise their cloud solutions might consider low-code and no-code platforms to achieve a quick turnaround?

We will approach this question by focusing on Rapid Prototyping, Time-to-market, Legacy modernisation, workflow automation and customisation flexibility.

Scenarios

1. Tech Debt Management: Many organisations face challenges with tech debt due to legacy systems. Refactoring using architectural and design patterns and cloud adoption frameworks can mitigate this. Leveraging low-code and no-code platforms allows phased modernisation, such as creating new customer portals while gradually updating back-end systems.

2. Rapid Prototyping & MVP: Low-code and no-code platforms are particularly effective for startups needing to launch minimum viable products (MVPs) quickly. These platforms facilitate quick iterations, enabling fast user feedback cycles.

3. Integration & Data Management: Low-code and no-code platforms benefit organisations that need to integrate different systems. They allow for creating integration layers that seamlessly manage data flows across multiple applications.

Low-code and no-code platforms are particularly advantageous for:

- Business Process Automation: Automating routine tasks, reducing manual errors, and improving efficiency.
- Internal Tools & Dashboards: Custom applications frequently updated by non-technical users.

- CRM Customization: Tailoring customer relationship systems to fit unique business needs.
- SME E-commerce: Quickly setting up e-commerce platforms and retail solutions for small and medium-sized enterprises.
- Event Management & Booking Systems: Facilitating scheduling, booking, and event handling applications.
- Educational & Training Tools: Building digital learning platforms and resources.
- Marketing Campaign Management: Streamlining marketing workflows and performance tracking.
- Mobile Apps: Developing mobile applications for non-critical business functions.
- Cloud Service Integration: Connecting disparate cloud services and managing APIs.
- Customer Portals: Creating user-friendly, self-service customer portals.
- Data Collection & Survey Tools: Simplifying the design and distribution of surveys and forms.
- Project Management: Enabling custom project management and collaboration tools.
- Compliance & Regulatory Reporting: Automating compliance checks and generating reports.

Specific Use Cases for Low-code No-code Platforms:

- Legacy System Modernization: Developing modern interfaces and microservices without completely overhauling back-end systems.
- Customer Experience: Enhancing customer engagement through personalised applications like self-service apps and web portals.
- Regulatory Compliance: Automating compliance processes with low-code and no-code tools, reducing the risk of errors and improving reporting efficiency.

- Integration & Workflow Automation: Tools like Zapier, Power Automate, and Integromat help automate workflows and integrate applications.
- Work & Collaboration: Platforms like Monday.com, Asana, and Trello offer project management and collaboration solutions.
- No-Code AI: Democratising AI with Google Cloud AI, Microsoft Azure AI, Salesforce Einstein AI or AWS AI Services.
- Content Management Systems (CMS): WordPress, Webflow, and Contentful provide easy-to-use tools for managing and publishing content.
- Website Builders: Platforms like Wix, Squarespace, and Weebly allow users to create websites without coding.
- Analytics: Google Data Studio and Power BI offer data visualisation and reporting without needing advanced skills.
- eCommerce: Shopify and WooCommerce enable the quick setup of online stores and product management.
- Search & Recommendation: Algolia and Amazon Personalize simplify the implementation of search functions and recommendation engines.
- Domain-Specific Platforms: Solutions like nCino for banking or Epic for healthcare cater to industry-specific needs.

Low-code and no-code platforms can rapidly streamline cloud-based tasks. For example:

- Finance: Mobile banking apps and CRM tools enhance customer interactions and streamline audits.
- Government: Public health surveillance and e-government services can be efficiently managed using low-code and no-code platforms.

Low-code, no-code platforms provide businesses with rapid development capabilities, allowing for cost reduction, innovation, and operational efficiency. They are instrumental

in scenarios requiring quick adaptation to changing business needs, such as mobile app development, cloud-based process automation, and customer portal enhancements.

- Consider adopting templates provided by Low-code and no-code platforms to expedite development.
- Explore tools integrating Figma or similar design platforms to simplify the UI/UX development process.
- Pre-built components for rapid deployment, especially for industries with stringent compliance needs. Platforms like OutSystems and Microsoft Power Platform offer robust features for enterprises looking to scale their operations.

There are two critical angles to explore in RQ2 when evaluating low-code and no-code platforms for cloud workload optimisation.

Firstly, many low-code and no-code platforms offer pre-built templates for specific use cases, such as CRM systems, e-commerce platforms, and workflow automation. These templates serve as an excellent starting point for rationalising cloud workloads that are already operational. In this case, a cost-benefit analysis is essential to assess the viability of redesigning the existing solution on a low-code and no-code platform versus retaining it. The study should consider whether the current workload continues to deliver value, whether shifting to a low-code and no-code platform could reduce operational costs, and whether the move would offer improvements in scalability, flexibility, or ease of updates. If the existing workload is still cost-effective and meets long-term goals, keeping it as-is might be the best action. However, suppose cost savings, improved efficiency, or better integration opportunities are identified. In that case, leveraging low-code and no-code templates to rebuild the solution can result in a more streamlined, cost-efficient operation while minimising dependency on highly skilled developers.

Secondly, some cloud workloads may no longer be cost-effective, leading organisations to consider low-code and no-code platforms as an alternative solution. This is particularly relevant for workloads that are inefficient, costly, or constrained by legacy technologies. Conducting a cost-benefit analysis in these scenarios is crucial. This analysis should compare the current cloud solution's operational costs with the projected costs of migrating or rebuilding the workload using a low-code and no-code platform. It is also essential to account for savings regarding reduced development time, maintenance expenses, and optimised infrastructure usage. Additionally, the scalability and flexibility offered by low-code and no-code platforms can provide long-term benefits, especially when integrating new systems or technologies without requiring large-scale overhauls. In this context, low-code and no-code platforms enable the rapid deployment of more agile and adaptive solutions, making them an attractive alternative.

A good starting point is to identify and evaluate workloads consuming more resources than necessary. Once these workloads are identified, assess whether rebuilding them on a low-code and no-code platform could offer significant cost savings and operational efficiency. A prudent approach would be to test the waters with smaller, non-critical workloads before moving larger, mission-critical systems onto low- and no-code platforms. By starting small, you can gauge the platform's performance, cost efficiency, and ease of use.

An iterative approach is highly recommended, beginning with a pilot project to test the viability of using low-code and no-code templates for cloud workload optimisation. This allows for carefully comparing performance and costs against the existing cloud setup. Choosing the right low-code and no-code platforms is equally important and should be based on the specific workload or industry requirements. For example, OutSystems and

Microsoft Power Platform are known for their enterprise-grade security, scalability, and compliance, making them ideal for larger organisations. It is also essential to be mindful of hidden costs, such as platform-specific training needs or potential vendor lock-in, which could affect long-term flexibility. Finally, consider the long-term viability of the chosen low-code and no-code platform, ensuring that it can support future growth and meet regulatory standards, predominantly if your organisation operates in a highly regulated industry like healthcare or finance.

To assess cloud workloads, the underlying architecture, and the tech stack while considering the adoption of low-code and no-code platforms, an effective questionnaire can guide decision-making on various costs, efforts, and operational overhead. Here are key questions to include in your assessment:

1. Current Cloud Workload Efficiency:

- What are the performance and resource utilisation metrics for existing workloads (e.g., CPU usage, memory, and storage consumption)?
- How scalable are the current workloads? Can they easily handle increased traffic or business demand?
- Are there frequent performance bottlenecks or downtime issues? If so, what are the leading causes (e.g., network latency, inefficient code)?

2. Cost Analysis:

- What are the total operating costs of the current cloud workloads (compute, storage, network, etc.)?
- How much are you spending on infrastructure provisioning, maintenance, and scaling?
- Are ongoing licensing costs or third-party services used with the current tech stack?

- How much does your organisation spend developer hours or external contractors to maintain or expand current workloads?

3. Technical Debt and Complexity:

- Are there legacy systems or outdated components in your current architecture? What challenges do they present (e.g., integration issues, complex maintenance)?

- What is the current level of technical debt in terms of unoptimised code, poor documentation, or unsupported technologies?

- How much effort would be required to refactor or modernise existing workloads if they continue on the current platform?

4. Operational Overhead:

- How many hours per week/month does your team spend managing, monitoring, and troubleshooting cloud workloads?

- Is there a need for specialised technical skills to maintain the current architecture? How dependent are you on those skills?

- How frequently do you need to update or patch systems, and what operational impact do those updates have (e.g., downtime, performance degradation)?

5. Development and Iteration Speed:

- How long does developing and deploying new features or updates on your current tech stack take?

- Are there bottlenecks or delays in the development cycle due to complex workflows, manual integration steps, or dependency issues?

- How does the current system support rapid prototyping and iteration of new business ideas or features?

6. Security and Compliance:

- Does the current tech stack meet your organisation's security and compliance requirements (e.g., GDPR, HIPAA, SOC 2)?

- How much effort is required to consistently meet data protection, encryption, and compliance standards?

- Are security patches and vulnerability management processes automated or manual?

7. Integration and Extensibility:

- How easily can you integrate existing workloads with third-party services, APIs, or cloud providers?

- How much effort is required to add new features or services to the current tech stack?

- Are there any limitations on the flexibility to integrate modern tools (e.g., AI/ML, analytics) into the current architecture?

8. Skillset and Resource Availability:

- Does your team have the necessary skills to maintain and develop new features on the current tech stack? Is there a gap in expertise that could lead to inefficiencies or delays?

- How dependent is your organisation on external contractors or consultants for ongoing cloud workload management?

9. Long-Term Sustainability and Innovation:

- Will the current architecture and tech stack support your long-term business goals and future growth (e.g., multi-cloud support, new markets, expanding products)?

- How difficult would it be to migrate or refactor workloads if significant technological shifts (e.g., cloud-native, serverless) are required in the future?

10. Low-Code/No-Code Platform Suitability:

- Do existing low-code and no-code platform templates closely match your current workloads or business processes?

- How much effort would it take to transition from your current tech stack to a low-code and no-code platform (e.g., re-building workflows, integrations, and data migration)?

- Would adopting a low-code and no-code platform reduce the time to market for new features or products? How would it impact your ability to innovate?

- What are the cost implications of using a low-code and no-code platform (e.g., subscription fees, platform limitations)?

- How easily can the low-code and no-code platforms scale with your business needs, and what are their complexity, customisation, and extensibility limitations?

- Do the low-code and no-code platforms align with your security, compliance, and data governance requirements?

11. Cost-Benefit of Low-Code/No-Code vs. Current Stack:

- What are the potential operational cost savings (e.g., infrastructure, developer hours) by adopting low-code and no-code platforms?

- How does the development and maintenance effort compare your current tech stack and low-code and no-code platforms?

- Can adopting low-code and no-code platforms reduce dependency on highly specialised skill sets or hard-to-find resources?

- Are there areas where the current system's complexity and customisation offer advantages over low-code and no-code solutions?

Addressing these questions will help your organisation assess whether adopting low-code and no-code platforms offers a cost-effective, efficient, and scalable alternative to cloud workloads. Focusing on operational overhead, long-term viability, and specific cost-benefit analysis will guide informed decisions about low-code and no-code adoption.

Organisations can make informed decisions that lead to more efficient, scalable, and cost-effective cloud solutions by carefully balancing the benefits of template-driven

rationalisation for existing workloads with the cost-benefit analysis of transitioning non-optimized workloads to low-code and no-code platforms. This strategy allows businesses to enhance their agility, reduce costs, and drive innovation while ensuring they make the most of modern cloud platforms.

4.3 Research Question Three

RQ3 What attributes of low-code and no-code platforms should be considered when assessing workloads against these platforms' features for investing in modernisation or rationalisation efforts?

Organisations increasingly explore low-code and no-code platforms in the evolving digital transformation landscape as viable solutions for modernising or rationalising their workloads. Understanding which attributes of these platforms to prioritise is essential for making informed investment decisions. This study assesses the critical attributes of low-code and no-code platforms, particularly focusing on integration capability, scalability, security features, user accessibility, and customisation options. To set a foundation for this analysis, we identify critical differences between low-code and no-code platforms, presented in a comparative table of fundamental attributes. This comparison serves as a precursor to uncovering additional factors that further differentiate these platforms in subsequent stages of the research.

Platform Attributes	**Low-Code**	**No-Code**
Target Users	Developers, IT Professionals, technically proficient business users	Business Users, non-developers, citizen developers

Ease of use	IDE or Visual development tool, drag-and-drop interfaces for rapid application development with WYSIWYG run and preview	Visual and intuitive drag-and-drop interfaces with WYSIWYG run and preview
Customisation & flexibility	Use popular languages like JS, SQL, REST, GraphQL, JSON, YAML or platform-specific languages like OutSystems Expressions, Appian Expressions, Zoho Creator's Deluge, Power Fx and Salesforce Apex	No custom code is allowed
Integration Capabilities	Third-party platforms, databases and APIs	Out-of-the integrations
Application Scope	Large-scale enterprise applications	Simple to moderately complex app development
Development Process	Source control integration, Collaboration tool integration	Collaboration tool integration
Deployment & Scalability	CI/CD integrations, automated testing, robust and scalable for enterprise-level applications	Easy deployment
Learning curve	Required coding skills	Extremely low learning curve. Kickstart with templates
Governance & Control	It allows more control over application development lifecycle management. The administration console is available to enable/disable feature usage in dev, test, and production environments with IAM.	Governance features are used to ensure organisational standards like security, compliance, and auditing. Governance is mostly simplified but may require oversight for compliance.
Development Speed	Higher than traditional methods, depends on the application complexity	Relatively high to enable rapid prototyping, often within hours or days
UX with the design process and workflow efficiency	Leading platforms support UX native integrations, handoffs of design specifications or imports of designs from Figma, Sketch, UXPin, Zeplin,	Optionally available to fast-track UX design and work with designers.

	Invision, and Creately, and collaboration features with designers.	

Table 4.13 Low-code and No-code Platform Attributes

Several critical attributes must be considered when assessing platforms for cloud workloads in modernisation and rationalisation efforts. These include platform extensibility, business logic implementation, availability of APIs and connectors, and interoperability to ensure smooth data exchange and workflow integration. Additionally, business process optimisation, data security (both in transit and at rest), and regulatory compliance certifications are essential factors in this evaluation.

Customer success stories across various platforms highlight these aspects:

- Compliance appeared in 326 keyword occurrences across 105 case studies.
- Security was mentioned 638 times across 272 case studies.
- Data security appeared 13 times in 18 case studies.
- Trust was a focus in 103 keyword occurrences across 70 case studies.
- Integration was a central theme, with 1,294 occurrences across 331 case studies.
- Reliable was mentioned 58 times in 46 case studies, reliably ten times in 8 case studies, and reliability 40 times in 18 case studies.
- Scalability was noted 55 times across 48 case studies.
- Automation stood out with 1,025 occurrences across 197 case studies.
- Governance was highlighted 80 times across 16 case studies.

These findings underscore the importance of key attributes like compliance, security, integration, scalability, and automation when evaluating platforms for modernising and rationalising cloud workloads.

Integration Capabilities—When evaluating the integration capabilities of low-code/no-code platforms, their ability to seamlessly connect with both legacy and modern systems,

APIs, and databases is crucial for ensuring efficient data flows and operational productivity.

OutSystems excels in this area with its Integration Hub, offering pre-configured connectors for widely used systems like SAP and Salesforce, which minimises setup time. Additionally, it supports custom API integration through REST, SOAP, and OData, making it adaptable for organizations with diverse technology stacks. OutSystems also offers real-time data synchronisation, ensuring smooth information flow across multiple platforms.

Similarly, Mendix is notable for its open and extensible integration model, supporting REST, SOAP, and OData services. With a wide array of pre-built connectors in the Mendix App Store, businesses can quickly integrate with external services and internal applications. This is particularly beneficial for enterprises seeking adaptable integration with both cloud-based services and on-premise systems.

Microsoft Power Platform stands out with its integration ecosystem of over 600 pre-built connectors through Power Automate, enabling seamless integration with Azure, Dynamics 365, and third-party applications such as Google and Twitter. This makes Power Platform a strong choice for cross-platform workflows, connecting diverse cloud environments while maintaining data consistency.

Appian's Integration Designer provides native support for enterprise systems like Salesforce, SAP, and Oracle and comprehensive API support for custom integrations. With its flexible architecture, Appian’s strength lies in supporting large-scale enterprise integration projects.

Salesforce acquired MuleSoft to offer one of the most powerful integration frameworks in the market. MuleSoft bridges on-premise systems and cloud-based services, enabling

seamless data synchronization across different environments. This makes it a top choice for organisations focused on unified data management across diverse infrastructures.

8base offers a GraphQL-based architecture for modern integration protocols that facilitates smooth and dynamic integration with external services and databases. Its API-first approach is well-suited for developers managing large datasets and multiple services.

AgilePoint and AuraQuantic also provide robust integration capabilities. AgilePoint integrates with enterprise systems like Microsoft Dynamics, Salesforce, and SAP. At the same time, AuraQuantic focuses on seamless integration with core enterprise systems such as Microsoft Dynamics and Oracle, ensuring smooth business processes and data flow.

Platforms like Zapier and Boomi complement these platforms by offering versatile integration solutions. Zapier, with its ability to connect over 5,000 apps, is ideal for quickly automating workflows between SaaS products without deep technical expertise. On the other hand, Boomi—an iPaaS from Dell Technologies—is tailored for large enterprises requiring deep, scalable integrations across on-premise and cloud environments.

These platforms offer a spectrum of integration capabilities, from rapid SaaS-based solutions like Zapier to enterprise-grade platforms like Boomi and MuleSoft. This makes them indispensable for businesses connecting diverse systems and ensuring seamless data flow across digital ecosystems.

Scalability—Low-code/no-code platforms are designed for scalability, allowing businesses to expand their applications as usage or demand increases.

Platforms such as OutSystems, Mendix, and Power Platform leverage cloud-native architectures, enabling organisations to scale vertically through increased resource allocation or horizontally across multiple servers. The flexibility offered by these platforms ensures they can handle high-traffic, high-performance applications.

8base and Bubble also provide cloud-first scalability, while platforms like AgilePoint and AuraQuantic support enterprise-wide workflow automation and business process management (BPM), ensuring smooth scalability across departments and regions.

Security remains a top priority for low-code/no-code platforms, especially for industries requiring strict regulatory compliance.

OutSystems offers comprehensive security features, including end-to-end encryption, role-based access control (RBAC), multi-factor authentication (MFA), and single sign-on (SSO). Its compliance with GDPR and HIPAA makes it suitable for industries with stringent security regulations.

Mendix provides a similarly robust security framework with SOC 2 and ISO 27001 compliance, advanced encryption, and robust authentication methods.

Power Platform integrates with Azure Active Directory (AAD) for user authentication and access control, aligning with existing security frameworks and global regulations such as GDPR and CCPA.

Appian, compliant with FedRAMP and SOC 2, offers built-in encryption, audit trails, and governance features, making it ideal for industries with high security and compliance requirements.

Customisability allows organisations to tailor applications to their unique needs without compromising flexibility.

OutSystems offers deep customisation capabilities, enabling the creation of custom workflows, UI/UX designs, and app extensions with custom code in languages such as JavaScript or CSS. Mendix and Bubble also provide significant customisation options. Mendix supports custom Java actions and UI/UX components, while Bubble's visual programming interface allows users to build sophisticated workflows and UIs without coding expertise. For more developer-centric customisation, 8base enables the

implementation of custom business logic and workflows using GraphQL, offering complete control over how applications interact with data.

Platforms like AuraQuantic and AgilePoint further extend customisation capabilities by offering intricate BPM solutions, making them highly adaptable for complex workflow automation.

User accessibility is a crucial consideration when evaluating low-code/no-code platforms, especially for businesses with both technical and non-technical users. Power Platform, Airtable, and Bubble provide intuitive interfaces, enabling users with little coding experience to create applications and workflows. Power Platform's drag-and-drop interface and integration with the Microsoft ecosystem make it accessible to a wide range of users, from business analysts to developers. Similarly, Airtable allows users to quickly build databases and collaborative tools, while Bubble provides a robust visual programming interface for creating fully functional web applications.

As enterprises scale across cloud environments, Identity and Access Management (IAM) becomes essential for ensuring security and control. Ideally, low-code/no-code platforms should support deployment to cloud providers like AWS, Azure, or GCP, leveraging their centralised IAM capabilities for role-based access control and multi-factor authentication. For instance, Power Platform integrates with Azure AD, centralising IAM for all workloads created using the platform. This centralised IAM approach enhances security and governance, ensuring all applications follow consistent access policies regardless of platform complexity.

In conclusion, when assessing workloads for modernisation or rationalisation using low-code/no-code platforms, key attributes such as integration capability, scalability, security features, customisation options, and user accessibility must be considered. These platforms offer varying extensibility, automation, and compliance levels, making aligning platform

capabilities with business needs crucial. By focusing on these attributes, organisations can ensure that their investment in low-code/no-code platforms drives agility, scalability, and long-term sustainability in a rapidly evolving digital landscape.

4.4 Research Question Four

RQ4 How are low-code and no-code platforms different from SaaS offerings, and how does integration with a SaaS platform affect the choice of low-code and no-code platforms? We will approach this question by focusing on customisation flexibility, vendor lock-in, integration complexity, feature extensibility, and deployment control. In terms of these factors, low-code and no-code platforms differ primarily from SaaS offerings.

Customisation flexibility in low-code/no-code platforms allows users to build tailored applications and workflows with minimal or no coding, making them more adaptable to unique business needs. In contrast, SaaS offerings provide pre-built software solutions with limited customisation options, typically constrained to configuration settings. Low-code platforms like OutSystems, Mendix, and Agilepoint allow greater flexibility in designing applications and automating workflows across diverse ecosystems. In contrast, SaaS products are typically designed to solve specific business functions (e.g., CRM, ERP) with pre-set features. In terms of vendor lock-in, SaaS offerings tend to embed customers into specific ecosystems (e.g., Salesforce or Microsoft 365) due to heavy reliance on proprietary services. Low-code/no-code platforms offer more flexibility, but platforms like Power Platform and Salesforce Lightning are tightly integrated into their ecosystems, which can create a similar lock-in. However, platforms like OutSystems and Mendix counteract this by supporting multi-cloud deployments and integrating various services, offering businesses more freedom to avoid total dependency on one vendor.

Integration complexity is another differentiator. Low-code/no-code platforms are designed to integrate seamlessly with multiple systems, allowing businesses to orchestrate workflows across various SaaS platforms. For example, Power Automate integrates Microsoft services, while Appian and AuraQuantic excel in process automation across multiple platforms like Salesforce and Microsoft 365. On the other hand, SaaS platforms often offer pre-configured integrations but lack the deep flexibility that low-code/no-code platforms provide in connecting multiple systems and data sources.

Feature extensibility is greater in low-code platforms since they allow users to build custom applications beyond the limitations of SaaS solutions. Platforms like Bubble and Betty Blocks enable the creation of full-scale applications without requiring extensive coding skills. In contrast, SaaS offerings are generally limited to extending existing features or purchasing third-party apps. Low-code platforms thus enable businesses to innovate and develop more complex, tailored solutions compared to the predefined functionality of SaaS. Deployment control is another significant distinction. Low-code platforms often offer hybrid or multi-cloud deployment options, giving businesses control over where their applications are hosted and how they are scaled. For example, OutSystems and Alpha Anywhere provide deployment flexibility across cloud or on-premises environments. In contrast, the vendor generally hosts and manages SaaS offerings, limiting customers' control over the underlying infrastructure and deployment flexibility.

In conclusion, integrating a SaaS platform affects the choice of low-code/no-code platforms by determining the ease of connectivity, customisation, and extensibility. Platforms with robust integration capabilities and flexible deployment options, such as Agilepoint, Mendix, and OutSystems, provide greater freedom and reduce the risks of vendor lock-in. At the same time, those tightly coupled with ecosystems like Power

Platform or Salesforce may limit flexibility but offer seamless integration within their environments.

4.5 Research Question Five

RQ5 How does support for open standards and integration with other cloud models, such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Function as a Service (FaaS) — often associated with serverless computing — and Container as a Service (CaaS), affect the choice of low-code and no-code platforms?

We will approach this question by focusing on interoperability, vendor neutrality, deployment flexibility, ecosystem compatibility, and scalability options.

Support for open standards and integration with cloud models like IaaS, PaaS, FaaS, and CaaS play a pivotal role in shaping the choice of low-code/no-code platforms. Platforms that embrace open standards such as REST, OpenAPI, OAuth 2.0, CNCF standards (e.g., OCI for containers, CNCF CloudEvents), and machine learning frameworks (ONNX, MLflow, TFX, PMML) allow for seamless interoperability across diverse systems and services. This openness is crucial for organisations leveraging different cloud models, ensuring easy integration of cloud-native services, enterprise systems, and external APIs. REST and OpenAPI enable platforms to interact with various cloud services, allowing apps to use cloud compute, storage, or event-driven functions efficiently.

Open standards directly influence vendor neutrality by ensuring that applications can be moved between cloud providers with minimal reconfiguration. For example, OCI standards for containers and Kubernetes orchestration (CNCF) allow developers to deploy applications in any cloud environment without vendor lock-in. OAuth 2.0 ensures secure, standardised authentication across platforms, while CNCF CloudEvents simplifies event-

driven architectures for serverless (FaaS) integration, making platforms more adaptable to multi-cloud and hybrid environments.

Standards like OCI enhance deployment flexibility, allowing containers to be deployed consistently across environments, whether on-premise, in public clouds, or across hybrid infrastructures. By adhering to CNCF and containerisation standards, platforms like AgilePoint and OutSystems can offer this deployment flexibility, ensuring that applications can be hosted where they make the most sense for performance, cost, or regulatory reasons. Open standards further strengthen ecosystem compatibility. Platforms that integrate with enterprise systems through REST, OpenAPI, and OAuth 2.0 can seamlessly connect to CRMs, ERPs, and cloud services. Moreover, standards in AI/ML like ONNX, MLflow, and PMML enable low-code platforms to integrate machine learning models from diverse frameworks, fostering the development of AI-powered applications. This broadens the ecosystem compatibility, allowing low-code/no-code platforms to integrate with modern machine learning pipelines and cloud-native AI services.

Scalability is directly impacted by the platform's adherence to open standards. Cloud-native standards like CNCF CloudEvents allow platforms to leverage serverless computing models (FaaS) for event-driven scalability, enabling applications to handle fluctuating workloads without over-provisioning resources. IaaS and PaaS integration based on standards like OpenAPI or REST further support scalability, allowing low-code/no-code applications to scale across different cloud services depending on demand dynamically.

In summary, open standards like REST, OpenAPI, OAuth 2.0, OCI, CNCF standards, and AI/ML frameworks (ONNX, PMML) enhance low-code/no-code platforms' ability to offer interoperability, vendor neutrality, deployment flexibility, ecosystem compatibility, and scalability. By supporting these standards, platforms become more versatile, allowing

businesses to integrate, scale, and deploy applications across a wide range of cloud models, maximising efficiency and avoiding vendor lock-in.

A critical gap in the low-code/no-code ecosystem is that many platforms still lag behind in fully embracing open standards and vendor-agnostic deployment models. While some outliers have made strides in this direction, the broader industry requires more focused research and development to build sophisticated standards that ensure true vendor neutrality and flexibility.

A concerted effort to establish open, interoperable standards across low-code/no-code platforms would enable customers to avoid vendor lock-in and maximise their deployment options. By creating a marketplace on major cloud providers, platforms could offer a more competitive landscape, benefiting users and vendors. This would foster an environment where platforms can integrate seamlessly with IaaS, PaaS, FaaS, and CaaS models, giving customers the flexibility to choose the best tools and services for their needs. The evolution of such standards could unlock the full potential of the citizen development revolution, allowing platforms, partners, and customers to innovate, collaborate, and grow without the constraints of proprietary ecosystems.

Therefore, the industry should prioritise the development of common frameworks and standards that facilitate cross-platform integration, multi-cloud deployments, and scalability across diverse cloud models. This will benefit customers through increased flexibility and choice, driving innovation and growth across the low-code/no-code ecosystem.

4.6 Summary of Findings

This extensive study evaluates over 300 platforms for citizen developers, focusing on low-code, no-code, and related platforms such as workflow automation, integration, AI, and

more. The research encompasses 415 platforms, broken down into categories like low-code/no-code (123 platforms), integration and workflow automation (51), no-code AI (21), website builders (12), AI startups (106), and other tools.

Key insights derived from the study of over 2,000 social media posts and 800+ case studies highlight the following:

- Development Efficiency: Platforms with visual interfaces and drag-and-drop features claim to accelerate development by 20-30 times and reduce time to market by 70-80%. They offer templates, integration options, and pre-built components that streamline the creation of applications across multiple devices.
- Platform Usage: These platforms have built over 3-4 million applications, supporting 1.5K to 1 billion monthly users across industries such as banking, healthcare, and government.
- Workflow Automation: Integration with APIs and automation capabilities is widespread, reducing operational inefficiencies and improving business processes.
- Generative AI: GenAI integration has become prominent, enabling platforms to offer conversational assistants, real-time collaboration, and enhanced user experiences for developers and end-users.
- Cost Savings: Low-code/no-code platforms reduce development costs, increase productivity, and contribute to legacy system modernisation. Organisations report significant reductions in time, cost, and errors.
- TCO and ROI: Platforms can evaluate savings on cloud workloads using Total Cost of Ownership (TCO) calculators and ROI models. These tools help calculate cost heads like human capital, automation tools, and operational agility.

4.7 Conclusion

Low-code/no-code platforms provide a powerful alternative to traditional cloud workloads, offering significant development speed, cost efficiency, and adaptability advantages. By empowering non-technical users to build applications, they reduce reliance on specialised development resources, streamline business processes, and shorten time-to-market for digital solutions.

However, while these platforms excel in efficiency, cost reduction, and accessibility, limited customisation is the most commonly cited drawback across user reviews on platforms like Gartner Peer Insights, G2, and Capterra. Many users report that while low-code/no-code tools are highly effective for simple or moderately complex applications, they often lack the flexibility for intricate or highly specialised systems. This limitation can become a bottleneck for businesses requiring extensive customisation or advanced functionalities, forcing them to supplement with traditional development or face scalability and feature depth restrictions.

In light of these concerns, organisations must carefully weigh the trade-offs between the rapid development benefits of low-code/no-code platforms and the customisation constraints. These platforms may require additional developer input or hybrid approaches to overcome customisation limitations for businesses with highly specialised workflows or complex integrations.

Ultimately, low-code/no-code platforms remain valuable for many use cases. However, they are best leveraged where simplicity, speed, and efficiency are prioritised over deep customisation and control. To maximise their potential, businesses should consider integrating them with traditional development workflows or utilising them for non-critical or less complex applications while considering long-term flexibility and scalability needs.

CHAPTER V:

DISCUSSION

5.1 Discussion of Results

Cloud transformation is frequently impeded by lengthy development cycles and rigid processes, which make it challenging to experiment and iterate rapidly. In this context, low-code/no-code (LCNC) platforms are valuable instruments that facilitate rapid prototyping. These platforms enable non-technical users to develop functional applications and collect real-time feedback, accelerating innovation. Businesses must deliver new solutions rapidly while maintaining a competitive edge in fast-changing cloud environments, which is where this ability to pivot and evolve ideas is particularly advantageous.

A fundamental difficulty in cloud transition is the risk involved with large-scale migrations, which frequently become too complex to handle. Low-code, no-code platforms provide an alternate way by enabling firms to create small, modular applications that can be grown and iterated incrementally. This corresponds with Agile techniques, which enable businesses to provide value gradually while minimising risk. The reduced complexity of low-code, no-code platforms-based solutions helps minimise transformation fatigue, allowing organisations to experience substantial results early in their cloud journey.

Low-code, no-code platforms offer a middle ground in the decision-making process between building in-house solutions or adopting third-party services. These platforms empower organisations to customise pre-built solutions with minimal development effort, allowing them to blend building and buying decisions based on specific needs. This flexibility accelerates time-to-market without sacrificing control, a critical factor for companies navigating cloud adoption and making pivotal transition decisions.

One of the most significant barriers to cloud transformation is the shortage of skilled talent, particularly cloud and software engineers. Low-code, no-code platforms alleviate this challenge by democratising application development, allowing non-technical users to contribute as "citizen developers." By distributing development tasks across a broader group, businesses can relieve the pressure on IT teams and expand the number of hands available to support transformation efforts. This democratisation of development fosters a collaborative culture where innovation is no longer restricted to technical experts.

Moreover, Low-code, no-code platforms promote a culture of innovation, collaboration, and problem-solving across organisations. By lowering the technical barriers to building applications, these platforms encourage teams from various departments to participate in the creation process. This helps nurture an environment where innovation thrives, with employees taking ownership of problem-solving initiatives. The rapid prototyping capabilities also support internal testing and dogfooding, ensuring that solutions are refined before being widely deployed.

Finally, as cloud transformations increasingly focus on data-driven decision-making, low-code and no-code platforms support a data-centric mindset. These platforms simplify data integration and governance, making it easier for organisations to manage and analyse data across various systems. By aligning business operations with modern data management practices, Low-cod, no-code platforms ensure that cloud transformation efforts are effective and sustainable.

In conclusion, the inherent challenges of cloud transformation—ranging from talent shortages to integration complexities—create a favourable environment for the rise of low-code/no-code platforms. These platforms provide the flexibility, scalability, and

democratisation needed to accelerate cloud initiatives while fostering a culture of innovation, collaboration, and problem-solving within organisations.

5.2 Discussion of Research Question One

Low-code and no-code platforms have emerged as viable alternatives for rationalising cloud workloads, offering significant potential to reduce costs associated with cloud services. This discussion focuses on key factors such as cost efficiency, scalability, development speed, complexity management, and resource utilisation, which were the primary criteria for evaluating these platforms in the context of citizen development as an alternative.

Cost Efficiency

Low-code/no-code platforms demonstrate cost efficiency primarily by reducing the need for highly specialised development teams and streamlining the development process. Businesses can significantly lower upfront infrastructure, labour, and maintenance costs. By shifting workloads from traditional IaaS or PaaS models to low-code/no-code platforms, organisations can benefit from consumption-based pricing models that eliminate unnecessary resource allocation, effectively reducing cloud expenditure. However, long-term cost-effectiveness needs careful evaluation, particularly in scaling and subscription models, as costs may rise if platform usage grows or advanced features are required.

Scalability

While scalability is a common strength of cloud platforms, low-code/no-code platforms offer their own scalability options. Many of these platforms support cloud-native architectures that allow applications to scale dynamically based on demand. OutSystems and Mendix, for example, provide robust scalability options, allowing large-scale

enterprise applications to be developed and deployed. However, the limited customisation noted by users on forums like Gartner Peer Insights and G2 can hinder scalability in cases where highly specialised workflows or integrations are required. Despite this, platforms increasingly incorporate features like serverless computing and microservices, which add to their scalability capabilities.

Development Speed

The most prominent advantage of low-code/no-code platforms is their ability to speed up the development process. Through visual development interfaces, pre-built templates, and drag-and-drop components, these platforms allow applications to be developed up to 20-30 times faster than traditional development methods. This acceleration in development speed enables businesses to respond quickly to changing market conditions and customer needs, allowing for a faster time-to-market. Additionally, the ability to quickly iterate and prototype solutions with minimal coding reduces the risk and cost of experimentation, further supporting their role as a viable alternative for cloud workload rationalisation.

Complexity Management

Low-code/no-code platforms simplify complexity management by abstracting much of the technical intricacies of development. For instance, businesses can focus on building business logic and user experiences without worrying about underlying infrastructure or integration complexities. This simplification is especially beneficial for non-technical teams involved in application development. However, the complexity increases when organisations need custom features, complex business rules, or highly tailored workflows, which these platforms may not fully support. This often necessitates supplementary traditional development, which can offset some cost and speed benefits.

Resource Utilisation

Resource utilisation is a crucial consideration when evaluating the role of low-code/no-code platforms in reducing cloud workload costs. These platforms provide efficient resource allocation by automatically optimising compute, storage, and networking resources. Moreover, many platforms allow for flexible deployment across multiple cloud environments, which is helpful for organisations leveraging hybrid or multi-cloud strategies. By allowing non-technical users to develop applications, these platforms also free up skilled developers for more complex, high-value tasks, enhancing overall resource utilisation within the organisation.

Challenges and Considerations

Despite the clear benefits, organisations need to address some limitations and challenges when using low-code/no-code platforms for cloud workload rationalisation. The most notable challenge is customisation, as these platforms may not always support the deep customisations needed for complex enterprise applications. In addition, vendor lock-in is a concern, especially when choosing platforms tightly integrated into specific ecosystems (e.g., Salesforce, Microsoft). Organisations need to assess the long-term implications of relying on low-code/no-code platforms in terms of both flexibility and cost management.

Moving Forward

To further this discussion, it is essential to analyse real-world case studies demonstrating how low-code/no-code platforms have been successfully deployed to rationalise cloud workloads. These examples can offer deeper insights into the practical applications of such platforms and highlight potential challenges that were overcome. A comparative analysis of different platforms for scalability, customisation options, and integration capabilities can provide a more nuanced understanding of when and how to leverage these tools effectively.

Further research should also evaluate the total cost of ownership (TCO) of low-code/no-code platforms over an extended period, especially as businesses scale and their requirements evolve. This will help determine whether these platforms remain cost-efficient and flexible long-term or if traditional development practices are still needed for specific complex cloud workloads.

5.3 Discussion of Research Question Two

Low-code and no-code platforms are particularly suited for scenarios that require rapid development, deployment, and easy integration, such as mobile apps, web portals, and internal business tools. These platforms allow businesses to iterate quickly and deploy solutions without needing large development teams. Their ease of use makes them a strong choice for small to medium-sized businesses (SMBs), startups, and organisations in their digital transformation journey (v1 products).

However, except for a few outliers, such as OutSystems, Mendix, and Power Platform, which offer enterprise-grade scalability, customisation, and deeper integration capabilities, most low-code/no-code platforms still cater primarily to SMBs and startups. These outliers are exceptions regarding their ability to handle complex, large-scale, and highly customised enterprise applications. Most platforms in this space are most effective for more uncomplicated use cases and rapid prototyping, making them ideal for smaller organisations and startups looking for fast and cost-effective solutions.

Scenarios for Low-code/No-code Platform Adoption

1. Tech Debt Management and Legacy Modernization

Low-code/no-code platforms allow businesses to modernise their legacy systems incrementally. Companies can build new customer-facing applications, mobile portals, or

dashboards while maintaining their existing legacy back-ends. For example, OutSystems and Mendix are frequently used in this phased modernisation approach because they offer powerful integration tools and the ability to interface with older systems.

2. Rapid Prototyping & MVP Development

Startups often need to quickly create Minimum Viable Products (MVPs) to validate their ideas in the market. Platforms like Bubble, Betty Blocks, and Adalo provide the perfect environment to develop, launch, and iterate MVPs in weeks rather than months. These platforms allow startups to experiment with different features, gather user feedback, and adjust their products accordingly.

3. Integration & Data Management

Platforms like Power Automate, Make, and Appian excel in automating workflows and managing data across different systems. These platforms benefit organisations that need to connect multiple data sources, integrate with third-party services, or automate routine tasks. The simplicity of setting up automated workflows without significant coding knowledge helps businesses operate more efficiently.

Critical Use Cases for Low-code/No-code Platforms

1. Mobile Apps Development

Mobile apps can be developed on platforms like Adalo, AppMaster, Alpha Anywhere, and OutSystems. These platforms benefit organisations looking to launch internal apps or customer-facing solutions that must be responsive across devices. For example, a startup could use Alpha Anywhere to quickly deploy a mobile app that integrates with a back-end database.

2. Customer Portals & Self-Service Platforms

Creating user-friendly web portals is critical for the banking, insurance, and healthcare industries. OutSystems and Appian offer robust tools to build customer portals and self-service platforms. These platforms enable organisations to create applications that improve customer engagement and reduce dependency on manual processes.

3. Business Process Automation (BPA)

Platforms allow businesses to automate various internal processes, such as HR onboarding, payroll processing, procurement, and supply chain management. These platforms allow businesses to build custom workflows without complex coding, improving operational efficiency.

4. CRM Customization

Platforms enable businesses to customise and extend their CRM systems to fit unique business workflows. This customisation allows organisations to tailor CRM solutions to match specific sales processes, marketing campaigns, or customer support functions, providing an enhanced customer experience.

5. E-commerce Solutions

Tools provide an easy-to-use interface for building online stores for small to medium-sized businesses (SMEs) looking to set up e-commerce platforms quickly. These platforms are ideal for SMEs needing a fast turnaround to sell products online while integrating with existing business processes like inventory management and payment gateways.

Application Types Tied to Various Platforms

- Mobile Applications: Platforms are well-suited for quickly creating mobile apps that can be deployed across multiple devices.
- Web Portals: Tools allow businesses to build scalable customer-facing web portals ideal for finance, healthcare, and insurance industries.

Internal Tools & Dashboards: Platforms are famous for creating internal management dashboards and collaboration tools. These platforms enable business teams to track performance metrics, monitor tasks, and manage workflows in real-time.

- Business Automation Tools: Tools allow organisations to automate repetitive tasks like processing customer requests or managing workflows across multiple systems.

Low-code/no-code platforms offer significant value in speed, flexibility, and cost-efficiency for startups. They allow founders and small teams to build MVPs, automate business processes, and integrate third-party services without hiring an entire development team. These platforms enable startups to:

- Launch products quickly,
- Iterate on customer feedback,
- Focus resources on business growth rather than development.

While most low-code/no-code platforms are designed for startups and smaller-scale applications, outliers like OutSystems, Mendix, and Power Platform are more suited to larger, more complex use cases and can handle enterprise-scale development. These platforms offer deeper customisation, better scalability, and integration with complex systems, making them an exception among low-code/no-code platforms.

Low-code/no-code platforms are invaluable for startups and small—to medium-sized businesses, mainly when rapid prototyping, business process automation, and seamless integration are priorities. Except for a few outliers, most of these platforms excel in building simple to moderately complex applications but may fall short in deeply customised or highly complex enterprise environments.

These platforms provide a critical advantage in speed, flexibility, and cost reduction for startups and early-stage companies. Larger enterprises, on the other hand, will benefit from a hybrid approach, combining low-code/no-code platforms with traditional development

for complex or mission-critical systems while using low-code tools for less critical applications where agility and speed are paramount.

Low-code/no-code platforms are an excellent choice for businesses looking for quick and cost-effective solutions. However, they need careful consideration when scaling or requiring advanced customisation.

5.4 Discussion of Research Question Three

When evaluating low-code and no-code platforms for workload modernisation or rationalisation, several key attributes are essential, including integration capability, scalability, security features, customisation options, and user accessibility. These attributes help determine whether a platform can meet the organisation's complex needs while supporting agile development, governance, and compliance. This discussion explores these attributes in the context of modernising cloud workloads and driving digital transformation initiatives.

Integration capabilities are crucial, especially for organisations with legacy systems and third-party applications. A platform's ability to seamlessly connect to both existing legacy systems and modern cloud services can significantly influence its viability for modernising workloads. The most effective platforms offer pre-built connectors for widely used services and custom API support, which allow businesses to automate workflows and synchronise data between different environments efficiently. Platforms with strong integration capabilities help reduce technical overhead and enable organisations to achieve better workflow automation and data synchronisation across systems.

Conversely, platforms with limited integration features can restrict modernization efforts, particularly for enterprises with complex IT ecosystems. Therefore, prioritising platforms

with robust integration frameworks is essential to ensure smooth data exchange and workflow automation.

Scalability is vital for businesses with growing demands or those operating in environments where workload fluctuations are common. The ability to scale applications horizontally and vertically is critical for handling larger volumes of data and users.

Some platforms are designed for scalability, leveraging cloud-native architectures to enable auto-scaling based on resource demand. This flexibility ensures that organisations can accommodate increased traffic or processing needs without extensive reconfiguration. Additionally, platforms that offer business process automation help organisations scale operations across departments or locations, further enhancing their capability to support growing demands.

However, scalability becomes a concern with platforms that are focused on smaller use cases or simpler applications, as they may not support the complex scaling needs of larger organisations. Platforms that provide enterprise-grade scalability are generally better suited for handling large-scale modernisation efforts.

Leading platforms offer comprehensive security features, including end-to-end encryption, role-based access control, and compliance with industry standards such as GDPR, HIPAA, and SOC 2. Additionally, integration with identity management systems for user authentication and access control enhances security across the organisation’s ecosystem. Such features are crucial for businesses in highly regulated sectors to ensure compliance with data protection and governance requirements.

It is essential to ensure that the platform selected offers the necessary security certifications and features to safeguard sensitive data and manage user access effectively.

Customisation is another critical attribute, particularly for organisations with unique business processes or workflows that pre-built templates or components cannot easily accommodate. The ability to tailor applications to fit specific needs is vital for ensuring that the platform can support long-term modernisation efforts.

Platforms that allow for deep customisation through the use of custom business logic and code enable organisations to create sophisticated applications that go beyond basic templates. This flexibility is crucial for companies with specialised needs requiring more than simple workflows or automated processes.
Platforms that limit customization may be sufficient for simpler use cases, but for businesses requiring more complex applications, it is essential to select a platform that supports deeper customisation and flexibility.

User accessibility and ease of use are central to the success of low-code/no-code platforms, especially for organisations that aim to empower non-technical users to contribute to application development. Platforms with intuitive drag-and-drop interfaces allow non-technical users to build and modify applications quickly, enabling business users to automate workflows and create custom solutions without the need for extensive technical expertise.
Empowering non-technical users helps organizations reduce their dependency on IT teams and fosters a culture of innovation by allowing more employees to contribute to application development. However, platforms designed for ease of use may sacrifice flexibility in

terms of customisation and advanced functionality, so businesses must balance accessibility and the ability to create complex solutions.

Governance and control features are critical for enterprises that need to manage the lifecycle of their applications, including version control, access management, and compliance with internal policies. Platforms with solid governance features provide centralised management consoles, role-based permissions, and audit trails, allowing organisations to maintain control over their application development processes.

These capabilities are essential for large organisations with multiple teams and departments working on various applications. They ensure that applications are developed in compliance with security and governance standards. Robust governance tools help businesses scale their application development efforts while maintaining oversight and control over critical security and compliance.

In conclusion, selecting a low-code/no-code platform for modernising or rationalising cloud workloads requires careful consideration of critical attributes such as integration capability, scalability, security features, customisation options, and user accessibility. Platforms that offer a balanced combination of flexibility, scalability, and enterprise-grade security are better suited for organisations with complex modernisation needs.

Organisations looking to empower non-technical users while maintaining control over security and customisation should focus on these attributes to ensure long-term success. By aligning the platform’s capabilities with the organisation's specific workload requirements, businesses can drive innovation, improve agility, and achieve more efficient governance and compliance.

5.5 Discussion of Research Question Four

Low-code/no-code platforms and SaaS differ in critical areas such as customisation flexibility, vendor lock-in, integration complexity, feature extensibility, and deployment control. These differences significantly influence an organisation's choice between the two, especially when integrating with SaaS solutions.

Customisation Flexibility

SaaS platforms offer limited customisation, primarily through configuration settings. They cater to specific business functions (e.g., CRM or HR). Low-code/no-code platforms, on the other hand, allow for greater customisation, enabling businesses to build tailored applications that meet unique needs. This flexibility is critical for organisations with complex workflows or specialised processes.

Vendor Lock-in

SaaS solutions often create vendor lock-in due to proprietary ecosystems and data formats, making migration difficult. Low-code/no-code platforms generally offer more flexibility, allowing businesses to choose deployment environments (cloud, on-premises, hybrid), though some platforms are still tied to specific cloud ecosystems.

Integration Complexity

SaaS platforms typically provide limited integration options, making them less adaptable to complex IT environments. In contrast, low-code/no-code platforms excel in integrating with legacy systems and third-party services, offering versatile connectors and APIs. This

makes them ideal for businesses needing seamless data exchange and process automation across systems.

Feature Extensibility

SaaS is constrained by its built-in feature set, with limited extensibility through plugins or extensions. Low-code/no-code platforms enable the creation of custom features and workflows, making them more suitable for businesses needing highly tailored solutions beyond what SaaS offers.

Deployment Control

SaaS platforms are vendor-hosted, and businesses have little control over deployment. In contrast, low-code/no-code platforms offer flexibility in deployment, including cloud, on-premises, and hybrid options, giving businesses more control over performance, security, and compliance.

Impact of SaaS Integration

Low-code/no-code platforms enhance SaaS by allowing businesses to extend and automate workflows across multiple SaaS tools. They provide pre-built integrations with popular SaaS platforms and custom connectors, enabling businesses to create unified processes, optimise operations, and bridge gaps in SaaS functionality.

Low-code/no-code platforms offer more customisation, integration, and control than SaaS, making them better suited for businesses with complex needs. Their ability to integrate and extend SaaS functionality is a significant advantage, especially for organisations looking to automate and optimise workflows across diverse systems. The choice between these

platforms should consider the balance between flexibility, integration requirements, and vendor lock-in.

5.6 Discussion of Research Question Five

Support for open standards and integration with cloud models like IaaS, PaaS, FaaS, and CaaS is essential in selecting low-code/no-code platforms. Standards such as REST, OpenAPI, OAuth 2.0, and CNCF guidelines (e.g., OCI for containers) ensure seamless interoperability across various systems, enabling smooth integration of cloud-native services and external APIs. This fosters vendor neutrality, allowing applications to move between cloud providers without lock-in, and enhances deployment flexibility across environments. Furthermore, adherence to AI/ML standards like ONNX or PMML enables platforms to incorporate machine learning models efficiently. Open standards increase ecosystem compatibility, scalability, and flexibility, making low-code/no-code platforms highly adaptable to diverse cloud infrastructures. A critical gap in the low-code/no-code ecosystem lies in the lagging adoption of open standards and vendor-agnostic deployment models. While specific platforms have progressed, much of the industry still lacks the sophistication to ensure genuine vendor neutrality and deployment flexibility. Without a more concerted push towards open, interoperable standards, customers risk being locked into proprietary ecosystems, limiting their ability to leverage the full range of cloud services.

Establishing universal standards across low-code/no-code platforms would empower customers to avoid vendor lock-in while expanding their deployment options. By fostering open marketplaces on major cloud providers, platforms can create a more competitive and dynamic environment, benefiting users and vendors. This would allow seamless integration

with IaaS, PaaS, FaaS, and CaaS models, allowing customers to choose the tools and services that best fit their needs.

In the long term, developing such open standards is essential for unlocking the full potential of the low-code/no-code movement. It would enable broader collaboration, innovation, and scalability, allowing platforms, partners, and customers to thrive without being restricted by closed, proprietary systems. To achieve this, the industry must prioritise building common frameworks that facilitate multi-cloud deployments and cross-platform integration, driving sustained growth and innovation across the ecosystem.

CHAPTER VI:

SUMMARY, IMPLICATIONS, AND RECOMMENDATIONS

6.1 Summary

The research highlights several key challenges and opportunities within the cloud transformation landscape, particularly with low-code/no-code (LCNC) platforms. LCNC platforms address significant hurdles in cloud transformation, such as the need for rapid prototyping, modular development, and reducing reliance on skilled developers. They promote innovation, collaboration, and problem-solving by empowering more users to contribute to application development. However, a critical gap remains in the ecosystem: the lack of widespread adoption of open standards and vendor-agnostic deployment models. While some platforms have made strides, the industry still needs more concerted effort to ensure vendor neutrality and multi-cloud integration.

6.2 Implications

The research suggests that LCNC platforms are well-positioned to tackle many challenges cloud transformation poses. However, for their full potential to be realised, the ecosystem needs to prioritise the development of open standards. By embracing standards such as REST, OpenAPI, OAuth 2.0, and CNCF guidelines, platforms can provide users greater deployment flexibility, scalability, and the ability to avoid vendor lock-in. This shift would foster a more competitive and innovative marketplace, benefiting customers and vendors. Adopting these standards would enable platforms to integrate seamlessly with IaaS, PaaS, FaaS, and CaaS models, allowing organisations to leverage the best tools and services available. Overall, the success of cloud transformation efforts will increasingly depend on

the industry's ability to support open, interoperable solutions that promote long-term innovation and collaboration.

6.3 Recommendations for Future Research

It is essential to explore the evolving role of AI in enhancing platform capabilities, the impact of robust component libraries and templates on user adoption, and the importance of deployment flexibility in preventing vendor lock-in. Additionally, understanding the market dynamics contributing to vendor consolidation and examining strategies to mitigate long-term vendor dependency and cost overrun risks will provide valuable insights for businesses and developers looking to leverage these platforms effectively. Here are a few questions for future research work:

- How is AI driving recent advancements in low-code/no-code platforms, and what specific features or improvements does it enable across key areas like onboarding, documentation, developer experience, and end-user interaction, particularly through AI-powered tools like chatbots?
- How are No-Code AI platforms driving the growth of citizen development across various industries, and what are the key factors contributing to their increasing adoption?
- What are No-Code AI platforms' key benefits and limitations in enabling citizen developers to build scalable, enterprise-ready AI solutions?
- In what ways do component libraries and pre-built templates contribute to the widespread adoption of low-code/no-code platforms, and how do they enhance development speed and accessibility?

- How does deployment flexibility in low-code/no-code platforms help mitigate vendor lock-in, and what are the standard deployment models (e.g., on-premise, hybrid, multi-cloud) available to users?
- What key factors contribute to the dominance of a few vendors in the low-code/no-code market, and how can organisations avoid long-term vendor dependency or unforeseen cost overruns?

In summary, continued exploration of these critical areas will drive the evolution and effective implementation of low-code/no-code platforms, fostering more significant innovation and accessibility.

6.4 Conclusion

Low-code/no-code platforms have emerged as powerful tools to address many of the challenges associated with cloud transformation, offering rapid prototyping, modular development, and greater accessibility for non-technical users. These platforms are instrumental in fostering innovation, collaboration, and scalability, making them essential to modern enterprise strategies. However, the broader adoption of open standards and vendor-neutral deployment models remains a critical gap in the ecosystem. For low-code/no-code platforms to fully realise their potential, the industry must prioritise developing interoperable frameworks that ensure flexibility, multi-cloud compatibility, and freedom from vendor lock-in. By doing so, the industry will enhance user choice and adaptability and drive sustainable growth, collaboration, and innovation in the evolving digital landscape.

REFERENCES

Bommadevara, N., Miglio, A. D., & Jansen, S. (2018, Apr 12). Cloud adoption to accelerate IT modernisation. p. 8. Retrieved from https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/cloud-adoption-to-accelerate-it-modernization

Chee, Y. M., Zhou, N., Meng, F. J., Bagheri, S., & Zhong, P. (2011). A Pattern-Based Approach to Cloud Transformation. *IEEE 4th International Conference on Cloud Computing*, 388-395. doi:10.1109/CLOUD.2011.86.

Digital transformation: Rewiring for digital and AI. (2023). Retrieved from McKinsey: https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/leadership-and-digital-transformation

Frank F. Xu, U. A. (2022, Feb 26). A Systematic Evaluation of Large Language Models of Code. *arXiv*, 13. doi:10.48550/arXiv.2202.13169

Gartner. (2023). *Enterprise Low-Code Application Platforms Reviews and Ratings*. Retrieved from Gartner: https://www.gartner.com/reviews/market/enterprise-low-code-application-platform

Humby, C. (2006). *https://en.wikipedia.org/wiki/Clive_Humby*. Retrieved from Wikipedia: https://en.wikipedia.org/wiki/Clive_Humby

Josef Spillner, Y. B.-P. (2018). Co-Transformation to Cloud-Native Applications Development Experiences and Experimental Evaluation.

Juncal Alonso, L. O.-E. (2013). Cloud modernisation assessment framework: Analyzing the impact of a potential migration to Cloud. *2013 IEEE 7th International Symposium on the Maintenance and Evolution of Service-Oriented and Cloud-Based Systems, Eindhoven, Netherlands*, 64-73. doi:10.1109/MESOCA.2013.6632736

Keith Conway, A. S. (2023, Jan 18). The FinOps way: How to avoid the pitfalls of realising the value of the cloud. *McKinsey*, 6. Retrieved from https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-finops-way-how-to-avoid-the-pitfalls-to-realizing-clouds-value

Khoshafian, D. S. (2020). *How to Alleviate Digital Transformation Debt: post-COVID-19"*.

Kryder, M. H. (2005). Retrieved from https://en.wikipedia.org/wiki/Mark_Kryder

Leila Abdollahi Vayghan, M. A. (2019, Jan 15). Kubernetes as an Availability Manager for Microservice Applications. *arXiv*, 10. doi:10.48550/arXiv.1901.04946

Mahdi Fahmideh Gholami, F. D. (2016). Cloud migration process—A survey, evaluation framework, and open challenges. *Journal of Systems and Software*, 31-69. doi:10.1016/j.jss.2016.06.068.

Manuela Dalibor, M. H. (2022, June). Generating customised low-code development platforms for digital twins. *Journal of Computer Languages*, 1-10. Retrieved from https://doi.org/10.1016/j.cola.2022.101117

Marita Canina, C. B. (2022, June). Digital Maturity. *Springer*, 1-8. Retrieved from https://doi.org/10.1007/978-3-319-98390-5_270-1

Mark Chen, J. T. (2021, Jul 07). Evaluating Large Language Models Trained on Code. *arXiv*. doi:10.48550/arXiv.2107.03374

Michael Chui, E. H. (2023, Jun 14). The economic potential of generative AI: The next productivity frontier. *McKinsey*, 68. Retrieved from https://www.mckinsey.com/capabilities/mckinsey-digital/our-insights/the-economic-potential-of-generative-ai-the-next-productivity-frontier

Nane Kratzke, P.-C. Q. (2017, April). Understanding cloud-native applications after ten years of cloud computing - A systematic mapping study. *Journal of Systems and Software*, 1-16. Retrieved from https://doi.org/10.1016/j.jss.2017.01.001

Oleksandr Matvitskyy, K. I. (2023). *Magic Quadrant for Enterprise Low-Code Application Platforms.* Gartner. Retrieved from https://www.gartner.com/doc/reprints?id=1-2F7NELJY&ct=231004&st=sb

Park, S. (2017). Digital Fluency. *Palgrave Macmillan, London*, 129-160. doi:10.1088/1742-6596/1529/5/052092

Petrosyan, A. (2023). *Internet usage worldwide - Statistics & Facts.* Statista. Retrieved from https://www.statista.com/topics/1145/internet-usage-worldwide/

Raquel Sanchis, Ó. G.-P. (2020). Low-Code as Enabler of Digital Transformation in Manufacturing Industry. *MDPI Applied Sciences*, 10-12. Retrieved from https://doi.org/10.3390/app10010012

SafonovaIrina, I., & Bamigbala, T. (2022, Dec 31). DIGITAL TRANSFORMATION OF SOFTWARE ENGINEERING OPERATIONAL MODELS Tools, platforms, and cloud services. 36. doi:10.13140/RG.2.2.29221.17129

Shazibul Islam Shamim, J. A. (2022, Nov 13). Benefits, Challenges, and Research Topics: A Multi-vocal Literature Review of Kubernetes. *arXiv*, 79. doi:10.48550/arXiv.2211.07032

Shuiguang Deng, H. Z. (2023, Jul 26). Cloud-Native Computing: A Survey from the Perspective of Services. *arXiv*, 32. doi:10.48550/arXiv.2306.14402

Woo, M. (2020, Sept). The Rise of No/Low Code Software Development—No Experience Needed? 960-961. Retrieved from https://doi.org/10.1016/j.eng.2020.07.007

Xiaoteng Zhu, S. G. (2021, Dec). Digital transformation: A systematic literature review. *Computers & Industrial Engineering*, 17. doi:10.1016/j.cie.2021.107774

Yajing Luo, P. L. (2021). Characteristics and Challenges of Low-Code Development: The Practitioners' Perspective. *arXiv*, 11-12. Retrieved from https://arxiv.org/abs/2107.07482v1

Yan, Z. (2021, Dec 28). The Impacts of Low/No-Code Development on Digital Transformation and Software Development. *arXiv*, 7. doi:10.48550/arXiv.2112.14073

Yue Wang, H. L. (2023, May 13). CodeT5+: Open Code Large Language Models for Code Understanding and Generation. *arXiv*, 26. doi:10.48550/arXiv.2305.07922

APPENDIX A:

PLATFORM ASSESSMENT CRITERIA

Core Criteria – Rating Scale

Complexity Management

Does this platform simplify the management of complex cloud workloads, potentially reducing the need for specialized resources.

- 0: "No management of complexity; overwhelming to users."
- 1: "High complexity, minimal tools to manage it."
- 2: "Moderate complexity with some management features."
- 3: "Balanced complexity; adequate tools to manage it."
- 4: "Low complexity; advanced features to simplify management."
- 5: "Excellent complexity management; minimal user effort required."

Cost Efficiency

The potential of this platform to lower development and maintenance costs compared to traditional cloud-based solutions.

- 0: "Extremely high cost; poor value for investment."
- 1: "High cost with marginal value."
- 2: "Moderate cost but still expensive for features provided."
- 3: "Fair cost for the features and benefits."
- 4: "Low cost; good balance between cost and value."
- 5: "Very cost-effective; excellent value for the price."

Customization Flexibility

Does this platform offer customizable solutions that can be tailored rapidly to meet specific business needs?

- 0: "No customization options available."

- 1: "Very limited customization with predefined templates."
- 2: "Basic customization options."
- 3: "Moderate flexibility with customization available for key features."
- 4: "High degree of customization; most features can be tailored."
- 5: "Full customization across all features with high flexibility."

Integration Capability

The ability of this platform to integrate with existing systems, applications, and data sources, enabling users to leverage their existing investments and resources, reducing migration, integration costs for seamless Modernization efforts.

0: "No integration capabilities."

1: "Very limited integration with third-party tools."

2: "Basic integration options with select tools."

3: "Moderate integration capability with several platforms."

4: "Strong integration with most common third-party tools."

5: "Full integration capability with broad, seamless third-party support."

Interoperability

The ability of this platform to work with other cloud services, tools, and technologies, enabling users to avoid vendor lock-in, reduce dependency risks, and increase flexibility and choice.

- 0: "No interoperability; works only in isolated systems."
- 1: "Very limited interoperability across platforms."
- 2: "Basic interoperability with select platforms."
- 3: "Moderate interoperability across some platforms."
- 4: "High interoperability with most platforms."
- 5: "Fully interoperable with all platforms and technologies."

Legacy Modernization

The ability of this platform to support the modernization of legacy systems, applications, and infrastructure, enabling users to improve agility, scalability, and efficiency, reducing technical debt and operational costs.

- 0: "No support for legacy system modernization."
- 1: "Very limited support for modernizing legacy systems."
- 2: "Basic modernization tools for legacy systems."
- 3: "Moderate support for legacy modernization."
- 4: "Strong modernization capabilities for legacy systems."
- 5: "Full support for modernizing legacy systems into modern platforms."

Scalability

The ability of this platform to scale up or down based on demand, enabling users to handle peak workloads, reduce costs, and improve performance.

- 0: "No scalability; limited to small workloads."
- 1: "Very limited scalability; unable to handle large workloads."
- 2: "Basic scalability, suitable for moderate growth."
- 3: "Moderate scalability; capable of handling increased demand."
- 4: "High scalability with flexible growth options."
- 5: "Fully scalable; able to grow indefinitely without limitations."

Cloud Vendor Lock-In

The risk of dependency on a single vendor, potentially limiting flexibility, choice, and control.

- 0: "No vendor lock-in; fully portable across cloud providers."
- 1: "Minimal vendor lock-in; flexible with some reconfiguration."
- 2: "Low vendor lock-in; moderate effort required to switch."

- 3: "Moderate vendor lock-in; significant re-engineering required to switch."
- 4: "High vendor lock-in; deeply integrated with one provider."
- 5: "Complete vendor lock-in; switching would be nearly impossible."

Time-to-Market

The speed at which this platform enables users to bring new products, services, and features to market, enabling rapid innovation, reducing time-to-market, and increasing competitiveness.

- 0: "Very slow time-to-market; major delays."
- 1: "Slow time-to-market with frequent bottlenecks."
- 2: "Average time-to-market with occasional delays."
- 3: "Moderate time-to-market with good responsiveness."
- 4: "Fast time-to-market with quick delivery."
- 5: "Extremely fast time-to-market; accelerated launch cycles."

Rapid Prototyping

The speed at which users can create prototypes, proofs of concept, and minimum viable products on this platform, enabling rapid experimentation, validation, and iteration, reducing time-to-market and development costs.

- 0: "No capability for rapid prototyping."
- 1: "Very slow prototyping process."
- 2: "Basic prototyping capabilities but slow."
- 3: "Moderate speed for prototyping with decent tools."
- 4: "Fast prototyping with advanced tools and templates."
- 5: "Extremely rapid prototyping; built-in tools for instant iterations."

Feature Extensibility

The ability of this platform to support the integration of new features and functionalities.

- 0: "No extensibility; features are fixed."
- 1: "Very limited feature extensibility."
- 2: "Basic ability to extend or add features."
- 3: "Moderate extensibility for adding new features."
- 4: "High degree of extensibility with customizable features."
- 5: "Full feature extensibility; users can create new features seamlessly."

UI Visual Development

Measures the platform's ability to enable visual development of user interfaces using drag-and-drop features, customizability, pre-built components, templates, and support for responsive design. It assesses ease of use, flexibility, and the quality of the visual development tools.

- 0: "No visual development tools; requires manual coding for UI creation."
- 1: "Very limited visual development; minimal drag-and-drop or pre-built components."
- 2: "Basic visual development with some pre-built components and limited customizability."
- 3: "Moderate visual development with decent drag-and-drop tools, templates, and customizability."
- 4: "Advanced visual development with a rich set of pre-built components, templates, and strong customization options for responsive design."
- 5: "Excellent visual development with comprehensive drag-and-drop features, fully customizable components, vast template libraries, and seamless support for modern, responsive UI design."

Workflow Automation

Evaluates the platform's ability to automate repetitive tasks, processes, and workflows either through built-in automation tools or integration with third-party workflow automation products. The platform should empower users, including citizen developers, to improve efficiency, reduce errors, and enable strategic use of resources.

- 0: "No workflow automation. The platform does not offer any built-in automation features, and no integration with third-party workflow tools is possible."
- 1: "Very limited workflow automation. The platform supports very basic automation for specific tasks, but lacks the ability to automate end-to-end processes. Integrations with external workflow tools are limited or non-functional."
- 2: "Basic workflow automation. The platform can automate specific workflows but lacks flexibility for more complex processes. There is some support for integrating third-party workflow automation tools, but manual intervention is often required for complex tasks."
- 3: "Moderate automation capabilities. The platform provides built-in automation for common workflows and supports integrations with third-party automation tools like Zapier, Microsoft Power Automate, or others. Automations for standard tasks are easy to configure, but complex automations require some manual setup."
- 4: "Strong workflow automation. The platform automates multiple processes natively and offers smooth integrations with third-party workflow automation tools, enabling users to design complex workflows with minimal technical intervention. Pre-built templates and drag-and-drop interfaces make it easy for citizen developers to automate more complex processes."

- 5: "Fully automated workflows with seamless integration. The platform provides comprehensive, fully automated workflows with minimal manual intervention required. It supports advanced native automation tools and easily integrates with external workflow automation products, offering end-to-end process automation with customizable, user-friendly interfaces for citizen developers."

Security Features

The security features and capabilities of this low-code/no-code platform, enabling citizen developers to easily implement security measures and protect their data, applications, and infrastructure. It includes support for regulatory compliance (SOC II, HIPAA, GDPR) and security best practices (OWASP), reducing security risks and ensuring data protection (in transit and at rest) without requiring deep technical expertise.

0: "No security features or compliance support; citizen developers must handle all security aspects manually, leaving applications highly vulnerable. No support for data encryption, OWASP, or compliance frameworks like SOC II, HIPAA, or GDPR."

- 1: "Minimal security features with little compliance support. Basic security measures may be available, but they are difficult to implement, leaving major vulnerabilities. Limited or no support for data encryption (in transit or at rest) and minimal adherence to OWASP standards."
- 2: "Basic security features are available, but compliance support (SOC II, HIPAA, GDPR) is incomplete or difficult to implement for citizen developers. The platform includes some basic encryption, but data security in transit and at rest may still be vulnerable. Some OWASP best practices are addressed, but the implementation is complex for non-technical users."
- 3: "Moderate security features; common security threats are addressed, and the platform provides guidance for data encryption (in transit and at rest) and support

for some compliance frameworks (SOC II, HIPAA, GDPR). OWASP basic security practices are incorporated, but some manual configuration may be needed. Citizen developers can implement these with moderate effort."

- 4: "Strong security features with support for major compliance frameworks (SOC II, HIPAA, GDPR) and built-in, easy-to-use data encryption (in transit and at rest). The platform includes built-in tools for OWASP best practices, making it easier for citizen developers to protect applications with minimal intervention."
- 5: "Excellent security features; full compliance with SOC II, HIPAA, and GDPR. The platform provides pre-configured, automated encryption for data in transit and at rest. OWASP security practices are seamlessly integrated, and security configurations are handled automatically. Citizen developers require little to no manual intervention, and the platform offers comprehensive, proactive security with full regulatory compliance."

Deployment Control

Evaluates the platform's control over deployment pipelines (CI/CD), environmental configurations (development, staging, production), and governance policies. It measures the ability of users, including citizen developers, to manage, customize, and monitor deployments effectively while maintaining compliance with governance standards.

- 0: "No control over deployment. The platform fully manages all deployments with no input from users. No CI/CD integration, environmental management, or governance oversight is possible."
- 1: "Minimal control over deployment options. Basic deployment settings are available, but users have little influence over the CI/CD process. Limited support for environmental configurations, with minimal governance features."

- 2: "Limited deployment control. Users have access to some deployment settings (e.g., basic environment selection like development or production), but CI/CD pipelines are predefined and not customizable. Basic governance features may be available but lack flexibility."
- 3: "Moderate deployment control. The platform allows customization of deployment settings and supports basic CI/CD integration (e.g., automated builds and tests). Environmental controls are available for multiple environments (e.g., dev, staging, production), with moderate governance features such as role-based access and logging."
- 4: "High degree of deployment control. The platform supports full CI/CD pipeline integration with customizable workflows (e.g., automated tests, rollbacks, continuous monitoring). Environmental configurations can be fully managed across different stages, and strong governance features (e.g., compliance checks, audit trails) are built in."
- 5: "Complete control over deployment environments, CI/CD pipelines, and governance. Users can fully customize the deployment pipeline (e.g., integration with external CI/CD tools, detailed automation workflows). Comprehensive environmental management (development, staging, production) with advanced governance policies (e.g., policy-driven deployments, compliance automation, and audit logging) is provided."

Extended Criteria – Rating Scale

Ecosystem Compatibility

The compatibility of this platform with other cloud services, tools, and technologies.

- 0: "No compatibility with external ecosystems."
- 1: "Very limited compatibility with select ecosystems."

- 2: "Basic compatibility with certain ecosystems."
- 3: "Moderate compatibility across multiple ecosystems."
- 4: "High compatibility with most common ecosystems."
- 5: "Fully compatible with a wide range of ecosystems and platforms."

hpaPaaS Vendor Neutrality

The ability of this platform to support multiple hpaPaaS vendors, enabling users to avoid vendor lock-in.

- 0: "No vendor neutrality; fully tied to one vendor."
- 1: "Very low vendor neutrality; mostly dependent on one vendor."
- 2: "Low vendor neutrality; moderate re-engineering required to switch."
- 3: "Moderate vendor neutrality; multi-cloud supported with some adjustments."
- 4: "High vendor neutrality; minimal changes required to switch vendors."
- 5: "Full vendor neutrality; seamless switching across multiple providers."

Development Speed

The speed at which developers can create, test, and deploy applications on this platform.

- 0: "Extremely slow development process."
- 1: "Very slow; significant delays in development."
- 2: "Moderately slow with some acceleration in specific areas."
- 3: "Average speed; development process is reasonable."
- 4: "Fast development; quick turnaround times."
- 5: "Extremely fast development; rapid iteration and prototyping."

No-code Database Support

The range and depth of database features supported by the platform, including capabilities such as relational databases, scalability, security, data integrity, and advanced querying, all without the need for code.

- 0: "No database support; entirely external solutions required."
- 1: "Very limited database support with basic CRUD operations."
- 2: "Basic relational database support with limited query and data integrity features."
- 3: "Moderate database support with standard relational features (foreign keys, indexes) and moderate scalability."
- 4: "Advanced database support with comprehensive relational, security, and integrity features, including built-in scalability and backups."
- 5: "Enterprise-grade database support with advanced querying (SQL, NoSQL), full ACID compliance, security, encryption, and high scalability."

Component Library Support

The platform's ability to support internal and external component libraries, provide ready-made templates, allow the creation of custom components, and offer a marketplace for component exchange and reuse.

- 0: "No support for component libraries, templates, or custom components."
- 1: "Very limited component library with basic internal templates and no external components."
- 2: "Basic support for internal component libraries and templates with limited external integration."
- 3: "Moderate support for internal and external component libraries and templates with some ability to build custom components."
- 4: "Advanced support for internal/external component libraries and templates, with strong custom component creation features."
- 5: "Full support for internal/external component libraries, templates, a rich marketplace, and advanced custom component development within the platform."

AI Performance and Usability

Measures the overall performance of the AI in terms of output quality (accuracy, coherence, creativity), response time, scalability, and resource efficiency, ensuring practical usability in a high-productivity environment.

- 0: "Extremely poor performance; outputs are irrelevant, incoherent, or slow."
- 1: "Low performance; outputs are inaccurate or delayed, with high resource usage."
- 2: "Moderate performance; outputs are somewhat accurate, but slow or inefficient."
- 3: "Good performance; outputs are mostly accurate, coherent, and timely."
- 4: "High performance; outputs are highly accurate, creative, and fast."
- 5: "Excellent performance; outputs are perfectly accurate, coherent, creative, and near real-time with minimal resource usage."

AI Customization and Ethics

Evaluates the platform's AI customization capabilities, ease of use for citizen developers, fine-tuning, ethical adherence, bias management, security features, and compliance with regulations (e.g., GDPR, HIPAA). Ensures the AI is adaptable, ethically responsible, and secure while being user-friendly for diverse business applications.

- 0: "No AI customization or ethical compliance. The platform offers no controls for citizen developers to adjust AI behavior or mitigate bias. No security measures for data privacy (in transit or at rest), and the AI exhibits clear bias. No support for GDPR, HIPAA, or other regulatory frameworks."
- 1: "Minimal AI customization. The platform provides very limited customization options for AI, with significant ethical concerns, security gaps, and bias in outputs. Basic security features, such as encryption, are manual and difficult to

implement. Compliance with GDPR, HIPAA, or similar frameworks is largely unsupported."

- 2: "Basic AI customization and ethical control. The platform allows for some AI adjustments (e.g., input parameters), but citizen developers may find it hard to manage. Moderate bias and security issues may arise, though basic encryption and compliance with GDPR, HIPAA are partially supported. Implementing ethical practices requires technical oversight."
- 3: "Good AI customization with built-in ethical guidelines. The platform offers pre-configured customization options (e.g., model tuning, adjustable algorithms) and tools to address common ethical issues, such as bias detection. Moderate security features (e.g., data encryption in transit and at rest) are available. Support for GDPR, HIPAA compliance is provided, but may require some configuration by citizen developers."
- 4: "Highly customizable AI with strong ethical controls and automated security. The platform provides easy-to-use customization tools, with advanced bias detection and mitigation techniques built-in. Strong encryption, automated security configurations, and built-in compliance with GDPR, HIPAA, and similar standards are available. Citizen developers can easily fine-tune AI models with minimal effort."
- 5: "Fully customizable AI with seamless ethical integration and compliance. The platform offers intuitive, fully customizable AI solutions with automated bias detection, fairness optimization, and ethical safeguards. Security features such as encryption, threat detection, and automated GDPR, HIPAA compliance are built-in and require no manual intervention. Citizen developers can easily adjust AI

behavior, fine-tune models, and ensure regulatory compliance without technical expertise."

Platform Support by Agencies or ISVs

Evaluates the availability, quality, and depth of support provided by third-party agencies or ISVs in terms of technical expertise, platform customization, integrations, and ongoing services. This includes the ecosystem of partners who offer extensions, maintenance, and consulting services for the platform.

- 0: "No support from agencies or ISVs. The platform is entirely self-managed with no external ecosystem for support, customization, or consulting."
- 1: "Minimal support from a limited number of agencies or ISVs. Basic customization or consulting services are available, but the ecosystem is small and lacking expertise."
- 2: "Limited support from a small network of agencies or ISVs. Some customization and integration services are available, but options are limited in terms of specialization and geographic availability."
- 3: "Moderate support from a growing network of agencies or ISVs. Customization, integration, and consulting services are available, with a range of partners offering industry-specific expertise."
- 4: "Strong support from a well-established ecosystem of agencies and ISVs. Comprehensive customization, integrations, and consulting services are readily available, with a diverse set of partners offering deep technical expertise and ongoing maintenance options."
- 5: "Extensive support from a large, mature ecosystem of agencies and ISVs. The platform has full global coverage, with highly specialized partners offering end-to-end services, from initial customization to ongoing support, industry-specific

solutions, advanced integrations, and strategic consulting. ISVs provide a rich marketplace of extensions and add-ons."

Code AI Criteria – Rating Scale

Code Quality

Measures the overall quality of the generated code in terms of maintainability, readability, and adherence to best practices.

- 0: "Poor quality; code is unreadable and hard to maintain."
- 1: "Low quality; code lacks structure and clarity."
- 2: "Moderate quality; some structure but hard to maintain."
- 3: "Good quality; readable and maintainable."
- 4: "High quality; follows best practices and is well-structured."
- 5: "Excellent quality; highly maintainable and clean."

Code Performance

Evaluates the performance of the generated code in terms of execution speed, memory usage, and efficiency. Code Performance focuses on speed and execution time in real-time or high-throughput environments.

- 0: "Very poor performance; high memory usage and slow execution."
- 1: "Low performance; inefficient and slow."
- 2: "Moderate performance; some inefficiencies."
- 3: "Good performance; performs well in most cases."
- 4: "High performance; optimized for speed and memory."
- 5: "Excellent performance; highly efficient and fast."

Generated Code Customizability

Assesses how easily the generated code can be customized or extended by users for specific needs.

- 0: "No customizability; code is rigid and unmodifiable."
- 1: "Very limited customizability."
- 2: "Moderate customizability; some flexibility for adjustments."
- 3: "Good customizability; allows for common modifications."
- 4: "Highly customizable; supports extensive modifications."
- 5: "Fully customizable; adaptable for any use case."

AI-Assisted Debugging

Evaluates the platform's ability to assist with debugging, identifying, and fixing issues in the generated code.

- 0: "No AI-assisted debugging available."
- 1: "Minimal AI debugging features with low accuracy."
- 2: "Moderate AI debugging; helps in identifying basic issues."
- 3: "Good AI debugging; identifies and suggests fixes for common issues."
- 4: "Highly effective AI debugging with robust issue detection."
- 5: "Fully automated debugging; AI fixes most issues autonomously."

Code Generation Accuracy

Measures how accurately the code generated by the AI aligns with the user's intent based on the provided prompt.

- 0: "Completely inaccurate; generated code does not meet any requirements."
- 1: "Low accuracy; generated code meets minimal requirements with significant errors."
- 2: "Moderate accuracy; some aspects of the code align with the prompt, but it requires fixes."
- 3: "Good accuracy; most of the generated code matches the prompt with minor adjustments needed."

- 4: "High accuracy; generated code closely matches the prompt with few deviations."
- 5: "Perfect accuracy; generated code fully meets the intent of the prompt."

Code Efficiency

Evaluates how efficient the generated code is in terms of performance, resource usage, and optimization. Code Efficiency focuses on resource optimization (memory, CPU, etc.).

- 0: "Extremely inefficient; code is resource-heavy and performs poorly."
- 1: "Low efficiency; code uses excessive resources with slow performance."
- 2: "Moderate efficiency; code performs reasonably but can be optimized."
- 3: "Good efficiency; code performs well with minimal resource usage."
- 4: "High efficiency; code is optimized for performance and resource usage."
- 5: "Perfect efficiency; code is extremely fast and resource-efficient."

Code Readability

Measures the readability and maintainability of the code generated from the prompt, ensuring it follows best coding practices.

- 0: "Unreadable code; poorly structured with no commenting."
- 1: "Low readability; hard to follow with minimal structure."
- 2: "Moderate readability; some structure but difficult to maintain."
- 3: "Good readability; follows standard practices but needs more clarity."
- 4: "High readability; well-structured with clear documentation."
- 5: "Perfect readability; clean code that is easy to understand and maintain."

Error Rate in Generated Code

Tracks the frequency of errors or bugs in the generated code based on the prompt. Error Rate focuses on the quality and correctness of the code (i.e., minimizing bugs and errors).

- 0: "High error rate; most of the generated code contains bugs or issues."
- 1: "Frequent errors; generated code requires significant fixes."
- 2: "Moderate errors; some bugs, but mostly functional."
- 3: "Low error rate; few bugs that are easy to fix."
- 4: "Very low error rate; minor issues in the generated code."
- 5: "No error rate; code is generated without any issues."

Code Security

Evaluates whether the generated code adheres to security best practices and avoids introducing vulnerabilities based on the prompt.

- 0: "Completely insecure; generated code contains major vulnerabilities."
- 1: "Low security; generated code introduces security risks."
- 2: "Moderate security; some secure practices but still vulnerable."
- 3: "Good security; follows secure practices with minor issues."
- 4: "High security; generated code adheres to security best practices."
- 5: "Perfect security; fully secure code with no vulnerabilities."