AI-Optimized DevOps for Streamlined Cloud CI/CD

Naveen Vemuri[1], Naresh Thaneeru[2], Venkata Manoj Tatikonda[3]
[1]Masters in Computer Science, Silicon Valley University, Bentonville, AR, USA
[2]Masters in Computer Applications, Kakatiya University, Bentonville, AR, USA
[3]Masters in Computer Science, Silicon Valley University, Bentonville, AR, USA

Received: 14 Feb 2024 Revised: 14 Feb 2024 Accepted: 16 Feb 2024 Published: 17 Feb 2024

Abstract:- This research pays attention to the merging of Development Operations with Artificial Intelligence (AI). It starts by realizing that AI will be an important factor in many aspects of work and that it will be automating some job functions. Consequently, AI will be presented as a tool that will enhance knowledge acquisition, provide job performance and professional development. The story stresses the opportunity cost attributed to the shift from software licensing to Software as a Service (SaaS) and underscores the benefits gained through early and regular software release by the organizations which have adopted the practice. DevOps, as a revolutionary approach, seeks to eliminate the gaps in the two central processes namely development and operations. The technology emerging, which includes big data, cloud computing, and mobile internet, calls for quick software deployment and consequently, the DevOps approach is what you get. However, DevOps is a unified approach. In the abstract, it will talk about continuous integration (CI) and continuous delivery (CD) spotting the cost-effectiveness and role of automation in the production process. Through AI and DevOps described, air is evident which is the AI role in automation and troubleshooting development in the software and hardware field. In the paragraph the author puts forth AI-optimized DevOps as a proposal that is not only efficient in development and distribution process but also fast in pacing. The overall wrap-up summarizes the AIOps and MLOps applications in conjunction with DevOps workflow to eliminate disconnection between machine learning model development and operational deployment. The big picture actually is condensed at the end. It outlines exactly how the AI DevOps approach works in modern software development, with particular focus on the cloud CI/ CD platform.

Keywords:- *AI-Optimized DevOps, Continuous Integration, Continuous Delivery, Automation in Software Deployment, Streamlined Cloud CI/CD, AIOps and MLOps Integration*

I. INTRODUCTION

Analysts and observers believe that a wide range of professional duties characterized by a predictable and repetitive nature will be automated. Intelligent Automation, often known as AI, refers to the use of artificial intelligence in processes that can acquire knowledge, adjust, and enhance their performance over time

The cost of software release has been reduced in recent years, mostly because of the transition from shrink-wrapped software to software as a service. Organizations that possess the ability to deploy software at an early stage and at a frequent rate have a greater capacity to effectively compete in the market. DevOps, a novel methodology, offers software firms the ability to achieve these objectives. DevOps is an organizational strategy focused on fostering empathy and promoting collaboration across different functions. DevOps has also been referred to as a "catalyst for increased collaboration within multinational organizations".

The objective of DevOps is to minimize the duration between software creation and operation while maintaining high-quality standards. IBM has defined Collaborative DevOps as the act of creating mechanisms to synchronize software development teams with IT operations teams. DevOps is also defined as the application of the same principles that control development to infrastructure management.

Now, the development of big data, cloud computing, and mobile Internet has resulted in a growth in the number of requests for quick implementation and frequent use of business software [1]. Contemporary businesses, particularly those in the mobile Internet sector, need swift value transmission to end customers, get user input, and then iterate and enhance their goods [2]. During this process, the development and operation teams must abandon their conventional separation and instead adopt the holistic approach of DevOps, which treats them as a unified entity [3]. There has been an increase in the number of demands for the rapid deployment and frequent use of business software because of the development of big data, cloud computing, and mobile Internet. This has led to an increase in the level of demand for these services. [4].

DevOps is revolutionizing the conventional software development paradigm. Due to the growth and standardization of the software development industry, software development is no more a work that can be accomplished by a single software engineer [5]. Product managers are required to participate in contact with customers and form ties with them across the whole of the software development, operation, and maintenance life cycle. When it comes to writing program codes, development teams, which are composed of several software engineers, are required to work together. In addition to this, testing teams are accountable for analyzing the codes and software products that are only half finished [6]. Following the successful completion of the final inspection and the audit conducted by the client, they will also be required to perform the responsibilities of software execution, procedure, and maintenance while it is being used [7]. The whole process relies on the collaboration of many individuals and teams inside the IT department, and even across other organizations [8].

The emergence of the Internet age has led to increased demands for improved quality and efficiency in software development and operation, due to the quick changes in consumer demand and the external environment [9]. Given this context, DevOps emerged specifically to provide IT personnel with a streamlined working environment and streamlined workflow.

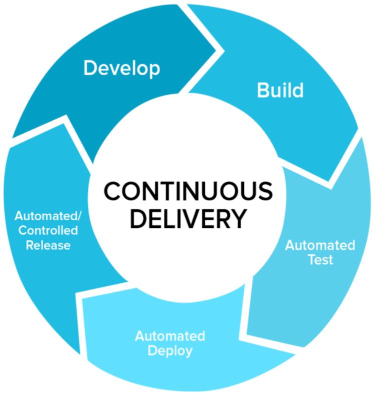

Fig 1 Continuous Delivery process

The figure above illustrates that the Continuous Delivery process encompasses the many stages of the Software Development Life Cycle (SDLC) with automation implemented at each step. This allows the user to efficiently deliver updates to clients in a dependable manner and accomplish seamless end-to-end automation, such as one-click deployment and rapid release. Therefore, there is no need for a fixed timetable to implement new items. The user can efficiently handle, troubleshoot, and implement microservices. The code must be resilient, ensuring it handles all edge test cases to maintain a standardized system process. This effectively resolves intricate issues encountered throughout the deployment phase, resulting in a smooth and streamlined operation. Users can increase the frequency of releases, therefore expediting the feedback cycle with consumers [10].

Continuous delivery streamlines the whole of the software delivery process via automation. Each commit will initiate the automatic process of building, testing, and deploying to the corresponding environments. The deployment to production may be performed manually to verify the intended configuration before releasing it to the production environment. The main objective is to streamline the application lifecycle and expedite value delivery by implementing an efficient production strategy in a short timeframe.

The primary objectives of Continuous Integration (CI) are to expedite issue detection and resolution, enhance software excellence, and minimize the duration required for validating and deploying new software upgrades. Continuous integration emphasizes the integration of smaller pushes and code modifications. A developer regularly makes code contributions, with a minimum frequency of once per day. The code is retrieved from the code repository by the developer to ensure that the code on the local host is integrated before it is sent to the build server. The build server will conduct a large number of tests during this step, after which it will either accept or reject the code change. The primary issues associated with establishing Continuous Integration (CI) include increased frequency of code changes to the shared codebase, ensuring the management of a unified source code repository, streamlining the process of building software and automating the testing phase. Further issues include conducting tests in settings that closely resemble the production environment, ensuring the team has clear insight into the testing process, and facilitating developers' seamless access to any version of the application.

Due to the frequent occurrence of integration, there is a substantial reduction in the need to backtrack and identify errors, allowing the user to allocate more time towards developing features.

Continuous Integration is cost-effective. Failure to integrate continually incurs high costs. Failure to adhere to a consistent strategy will result in longer intervals between integrations. This greatly increases the level of difficulty in identifying and resolving issues. Integration issues may easily disrupt project timelines or lead to complete project failure [11].

Automation is the central activity of DevOps, with a particular emphasis on setting up, configuring, deploying, and supporting infrastructure and applications. Automation facilitates the quick establishment of surroundings in a consistent and automated manner. In the past, server setup and application deployment were mostly carried out manually, which had a significant risk of mistakes, lack of stability, and inability to support agile business practices. To tackle these challenges, organizations were using highly proficient personnel to carry out manual setup. However, this did not resolve the issue, which might still affect crucial and valuable tasks such as software deployment, machine setup, operating system updates, problem-solving, or bug resolution inside a company. Automation serves as a savior by automating the bulk of vital company processes.

The DevOps environment offers a comprehensive automated approach that eliminates the strain on enterprises caused by human intervention or access to production settings. DevOps uses technologies to automate significant portions of the whole software development and deployment process [12].

Continuous Integration (CI) involves the regular integration and merging of development work, such as code, by team members frequently, sometimes several times per day. Continuous Integration (CI) makes it possible to establish a software release cycle that is both shorter and more frequent, which ultimately results in improved software quality and increased productivity [10]. Utilizing automated processes for the creation and assessment of software is a necessary step in this process. After the successful completion of automated tests and quality assessments, the purpose of continuous delivery (CD) is to ensure that an application constantly maintains a state of readiness for production. This is the ideal outcome of a CD. Continuous integration (CI) and deployment automation are two methods that are used by continuous delivery (CD) to automatically release software in an environment that is extremely similar to production. Force is the most important aspect to consider when it comes to CDs. According to the findings of the research [13], this technique provides a multitude of advantages, including a decreased risk of implementation, decreased expenses, and more prompt access to user input.

II. LITERATURE REVIEW

A. Continuous Delivery

Continuous delivery often consists of many essential stages. Developers decompose their tasks into incremental updates or modifications, which are then implemented on the trunk or mainline in the version control system. Automated tests provide developers with prompt feedback within minutes whenever they make a change, which is another crucial component. Automated testing enables prompt feedback and learning, giving developers the chance to promptly address any issues that arise when the automated tests fail. Collectively, these procedures are referred to as continuous integration. Following the successful completion of the build and tests, more processes are carried out, which include doing thorough automated acceptance testing, human exploratory testing, and performance testing. Automatic program deployment to an environment generated automatically using version control-maintained system and application configuration data is essential for these to occur.

One of the most common development strategies used in the software development industry is known as Continuous Integration (CI). Members of the team must integrate and merge their development work consistently, which may require integrating and merging code many times each day. Through Continuous Integration (CI), software companies can accomplish shorter and more frequent release cycles, improve the quality of their software, and increase the productivity of their staff. The use of automated software development and testing is included in this strategy.

The objective of Continuous Delivery is to identify methods for efficiently, rapidly, and dependably delivering software that is of high quality and value. Continuous Delivery focuses on the velocity of the market and the ability to swiftly go from conceptualization to implementation, surpassing competitors in terms of speed. The high pace of this industry results in a reduced feedback cycle and a quicker period to achieve value. A shorter feedback loop enables quicker failure, faster resolution, more rapid adjustments, and ultimately, faster success.

The agility shown by firms like Amazon and Netflix gives them a clear competitive edge in the business world. This is seen in their ability to dominate their sectors, expand into related industries, and establish themselves as market leaders. These organizations possess the ability to swiftly adjust to the evolving marketplace and outpace their competition in delivering novel products and services. The deployment pipeline automates the build, deploy, test, and release process for an application; it is the central tenet of continuous delivery. If an application undergoes any kind of alteration, the deployment pipeline is instantly activated.

B. Continuous Integration

A common development approach in the software development industry is Continuous Integration (CI). At least a few times a day, team members must merge and integrate their development efforts, including code. Software companies may improve software quality, increase team efficiency, and shorten release cycles by using Continuous Integration (CI). Automated software development and testing is a part of this strategy. As part of the Continuous Integration process, developers often commit all of their changes to a central repository, after which they do automated builds and tests. Building or integrating the program is often referred to as CI in the software release process. An automated part, like continuous integration (CI) or build service, and a cultural part, like the habit of frequent integration, are both necessary.

C. DevOps

A strategy that mixes artificial intelligence and device analytics, DevOps is a business-oriented approach to software development. When DevOps teams make use of software that is driven by artificial intelligence, they are able to make use of a large number of data points, which enables them to simplify the testing, development, release, and monitoring of products with increased accuracy and efficiency. It also speeds up the whole procedure, which is another benefit. [5,6]. Multiple automated processes need identification and resolution of challenges, as well as collaboration among team members to enhance efficiency in development and operation.

The use of artificial intelligence results in a considerable improvement in the efficiency of DevOps [6]. Artificial intelligence (AI) provides a variety of technologies that provide users in the business world the ability to rapidly map and integrate data, therefore enhancing crucial business choices and fostering a favorable experience for customers. Machines may optimize the data management process in DevOps systems to enhance value creation.

Technology enables individuals to effectively manage information silos, ensuring timely access to accurate messages and facilitating optimal decision-making [6]. Users may use an AI platform to efficiently map a substantial amount of fragmented data. Subsequently, this information may be consolidated into a unified database. Their artificial intelligence employs data processing techniques that enable them to make precise judgments devoid of any flaws or contradictions. Artificial intelligence data mapping software facilitates the precise and effortless mapping of vast quantities of data. An integrative approach may be used to link the separate data repositories inside the IT system, leading to precise consumer insights that may enhance data integrity and efficiency.

AI technology will facilitate users in efficiently and precisely exploring diverse data sources to accomplish activities with increased speed and accuracy [6]. Artificial intelligence has the capability to combine data from many sources to provide a coherent and effective data analysis procedure.

III. AI AND DEVOPS

AI and DevOps are interconnected and complementary since AI is used inside a framework to enhance the performance of information technology devices. DevOps solutions, in contrast, are a business-oriented strategy for software delivery [14]. The use of artificial intelligence (AI) enables DevOps teams to acquire the knowledge and skills necessary to expedite the process of software development and distribution. Moreover, AI has the capability to enhance automation, swiftly resolve issues, and effortlessly address any remaining challenges. AI and DevOps play crucial roles in the IT industry [15, 16].

These are business-oriented strategies that expedite application development by integrating software development and operations. Enhancing the precision of their findings might perhaps lead to greater success. The stage may have a more specific focus on evaluating the firm's ability to invest in an application by predicting the outcome. It is capable of providing products on a timeline rather than a normal schedule because of its ability to respond more quickly than usual. Due to its ability to generate faster and more accurate operations, it may result in a higher rate of deployment [16]. DevOps facilitates the uninterrupted delivery of value to end-users by automating and assuring the seamless functioning of systems.

Artificial intelligence (AI) is becoming indispensable for the development and administration of applications [16]. Artificial Intelligence (AI) is user-friendly, but corporations often grasp the true potential of this technology when they overlook its use in enhancing organizational activities. Artificial intelligence plays a crucial role in DevOps by effectively resolving various challenges faced by the IT industry. They are essential for the progression of the digital transformation [15].

IV. AI-OPTIMIZED DEVOPS

Despite the fact that AIOps and MLOps are relatively new terms in the field of software development, their scopes and applications may be distinct from one another. The ideas of MLOps are used throughout the whole process in order to maximize the economic impact of the solutions that are generated by machine learning. According to the information presented in this article, MLOps is essentially a combination of the concepts of DevOps with machine learning, which ends up producing a variety of advantages for businesses. It has been a regular practice in the field of software development to make use of a wide variety of tools and methods in order to fulfill the needs of the applications that are now in use [17].

Machine learning software is not immune from this trend, as seen by the current excitement around machine learning and artificial intelligence. DevOps is more complicated in terms of producing applications that are both quick and effective for modern clients. It is important to take into consideration the ideas of Machine Learning Operations (MLOps) and Artificial Intelligence for IT operations when contemplating the inclusion of machine learning or artificial intelligence into DevOps (AIOps). Although there are a few significant differences between the two statements, the importance of each of them is similar [17].

It is possible to describe MLOps as the use of DevOps principles, techniques, and technologies within the context of machine learning initiatives. The concepts of DevOps, which include collaboration, source control, testing, and continuous integration and automation (CI/CD), are used by machine learning projects in order to effectively assist the whole process, beginning with the design phase and continuing through training and deployment [17].

For example, MLOps makes it possible to integrate data science and IT operations teams in a seamless manner, which in turn makes it possible to deliver machine learning projects more efficiently and dependably. It is vital to connect a technology sector with well-defined processes that are focused on business in order to make the most of the capabilities that it has. Otherwise, machine learning faces the risk of being reduced to a basic experiment and liability for a firm, with few or no economic gains [17]. This would be a significant loss for the profession. To optimize the commercial benefits of machine learning, MLOps is used throughout the iterative process of designing, constructing models, and carrying out operations.

During the design phase of a machine learning project, the major purpose is to define the different business and data requirements that are necessary for the project. During this phase, a business model and a solution based on machine learning are built in order to successfully solve the problem that the user is experiencing. The process of identifying the potential users of the application is that which is being carried out here. In addition, the software's potential applications are improved, and an evaluation is carried out to establish whether or not the essential data that is necessary for the training of an artificial intelligence model is available. The data that is gathered from each of these processes is subsequently used in the architectural design of the application, which is driven by machine learning [17].

The product that was created as a consequence of the design stage that came before it is put through testing during the experimentation and development phase. The purpose of this testing is to validate the practical use of the proposed machine-learning solution in real-world settings. After that, the machine learning approach is repeated until a stable model is achieved, at which point it may be put into production for use.

DevOps strategies that have been proven successful, such as MLOps in Operations, are used in order to successfully apply the model in a production setting. This application phase is driven by machine learning, and it makes use of testing, monitoring, versioning, automation, and continuous delivery, in addition to other DevOps methods [17, 18]. It is required to apply each of the aforementioned stages as a following step to the one that came before it in order to get a machine learning project ready for the customers.

V. THE PRINCIPLES OF MLOPS

The Principles of MLOps MLOps, similar to DevOps, is governed by principles in both its implementation and execution. MLOps is the machine learning counterpart of DevOps. Although there could exist more concepts, the following ones are considered the most crucial in MLOps.

A. Testing Phase

As is the case with DevOps, testing is an essential component of both the process of developing software and the process of machine learning. A data pipeline, an ML model pipeline, and an application pipeline are the three components that makeup MLOps, which is a method of testing machine learning systems that makes use of a systematic methodology [18]. A detailed evaluation is carried out on each component in order to evaluate the accuracy of the machine learning model. This is done to guarantee that the model is both practical and able to be integrated without any problems.

B. Tracking

In order to ensure that a machine learning model that has been deployed remains accurate, it is essential to do regular monitoring of the model. The dependencies of the model, the data version, the usage of the model, and any changes to the model are all carefully monitored to ensure that it functions as intended. It is required to predefine the behaviors that are supposed to be shown by the model and then utilize those behaviors as a benchmark for comparison in order to validate the performance of the model. If the model does not live up to expectations or displays spikes that are not consistent, the proper actions should be taken to address the problem [18].

C. Revision

It is essential to make use of a version control system to properly manage the machine learning model. This system is responsible for monitoring and documenting any alterations that are made to the data sets and code. As a consequence of this, the version control tool makes it possible to keep track of and save many versions of the machine learning model if events cause the data to be altered or produce anomalies [18]. When a problem occurs, it is simple to roll back to an earlier version or exactly pinpoint the area where a software issue was fixed. Both of these options are available.

D. Persistence

The continuous workflow of DevOps has been included in the machine learning pipeline by MLOps, which is one of the greatest advantages of MLOps. The lifetime of a machine learning model is finite, and it is sensitive to alteration as fresh data is constantly being added to the system. The abbreviation MLOps, which stands for Machine Learning Operations, was developed to ease the process of implementing machine learning engineering strategies. These strategies include continuous integration (CI), continuous delivery (CD), constant testing (CT), and continual monitoring (CM) [18].

E. Implementation of Automatic Processes

In order to successfully include MLOps in any machine learning project, it is essential to place a considerable amount of emphasis on the automation process. It is possible to determine the level of maturity of the machine-learning process by examining the degree of automation present in your machine-learning model. This, in turn, increases the pace at which machine-learning models can be trained, developed, and deployed. This theory makes use of critical events as triggers for each activity in the workflow, which makes it possible to have a machine-learning workflow that is automated and does not need any intervention from humans. Three main stages of adoption correspond to the automation of MLOps.

DevOps's continuous integration and continuous delivery pipeline automation are used during the third phase to develop, test, and autonomously deploy machine learning models. In order to build, test, and autonomously deploy machine learning models, the third phase of the DevOps process involves the use of the CI/CD pipeline.

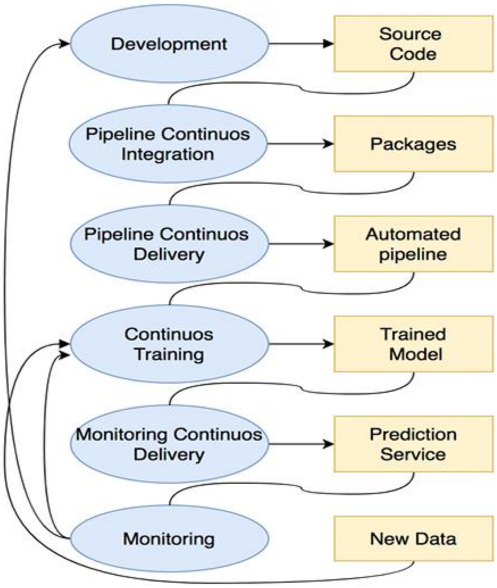

Fig 2 MLOPS Automated CI/CD Pipeline [18]

VI. CONCLUSION

The conclusion of AI-enhanced DevOps envisioning a future where Streamlined Cloud Continuous Integration and Continuous Deployment is the norm is remarkable, as it shows that software development and deployment are changing greatly. AI/DevOps amalgamation signals a major overhaul, handling existing hurdles and smoothening processes leading to improved productivity and economic stability.

The irreversible development of AI, on the other hand, can be explained by the now apparent fact that many professional tasks are very predictable and have repetitive natures. The narrative pinpoints AI's function as a dynamic force with the capacity for learning, improving, and maintaining performance throughout time. In software release, the economic situation went through the transformation from a traditional inner packing model, to the Software as a Service (SaaS) model. This gave an opportunity not only to those organizations that could release software earlier and frequently, but also get a positive strategic marketing advantage.

DevOps is emerging as a novel approach that seeks to infuse empathy and collaboration into a unified organizational strategy spanning technology teams, process owners and stakeholders alike. It might be called the (catalyst for collaboration) among the multinationals and reduces the time between the idea and the operation, and, definitely, it maintains the (standards of quality). Besides revealing the meaning behind the IBM's Collaborative DevOps, the author underlines the coordination of software development and IT operations, which corroborates with DevOps principles for the purpose of managing the infrastructure.

While established multinational corporations maintained control of the market through the use of big data, cloud computing and mobile internet, the need for speed in launching and iterating software has risen sharply. Today, the contact with DevOps, a single entity who offers collaboration between development and operation teams, is something that is essential for today’s industries especially in the mobile internet sector. Comprehensive DevOps platform is pointed out as the solution tackling the challenges. Those can be the software delivery whereby the platform is used, the operation costs cuts, and development risk decreases.

DevOps, the two key phrases in the process of automation are Continuous Integration and Continuous Delivery (CI/CD) systems, where every step of the Software Development Life Cycle (SDLC) is taken care of automatically. The streamlined and automated process as shown in the Figure-1 depicts the overall efficiency gained which is due to automation of end-to-end release, configuration and test automation, and ease of one click deployment and release (Figure- 1) as a result. Apart from removing time-constraints, users can, with ability and ease through micro-services, be able to fix, correct, and apply service features.

Continuous Integration serves its purpose by fastening the bug-out process, boosting the overall quality of the software and decreasing as time goes by, the D-Day moments we all so dread. As the code changes are incorporated over the timely basis, the entrepreneurs of the backtracking and even more the activity of feature development emerges. That, in turn, leads to the low integration cost.

Automation is still the driving force of DevOps practices, allowing us to take care of anthropogenic components, such as manual installation and implementation. Traditionally, manual server configuration, software deployment and update process brings about many risks while stability is low and agility is limited. Automated processes like creating servers or installing and updating software, mitigate these risks, enhance stability and support agile business activities. Being efficient as a DevOps is a solution that means the speeding-up of vital operation processes and also allows to create infrastructure well-taking into account all consistency and automation demands.

AI and DevOps trend is essential, in line with AI’s tools that increase the automation and fast problem solving, and in eliminating the Disruptive factor of the IT industry. AI-based DevOps as an innovative model enables organizations to concentrate on capitalizing and exercising appreciated economic significance to enrich efficiency in offering software.

Moreover the abstract talks in brevity about AIOps and MLOps featuring their association with DevOps methods to enhance machine learning solutions. The primary parts of MLOps which are testing, tracking, revision, persistence, and automation control depict how a strict ordering for an ML to become a part of the development and operation pipelines is important.

In short, it is not the AI-fueled DevOps exploration which brings into life an AI/DevOps merger, but the latter is the agenda which urges technical integration. It represents a win-win situation for the software development and deployment process as it not only caters to the fast changing conditions on the technological front but also provides a robust and flexible platform for organizations to keep up with the competition which is relying more and more on innovative technologies.

CONFLICTS OF INTEREST

The authors declare no conflicts of interest regarding the publication of this paper.

REFERENCES

[1]. Chen Yajun, lily li, Xu Xiaokun, et al. Research on the application of cloud platform for safety monitoring of water conservancy and hydropower projects based on cloud computing microservice architecture and DevOps concept. Digital Technology and Application, vol. 38, no. 3, pp. 5, 2020.

[2]. Qiao Hongming, Liang Huan, Yao Wensheng, et al. Discussion on DevOps security management for 5G network. Mobile Communications, vol. 43, no. 10, pp. 5, 2019.

[3]. Liu Liyuan. Application of CMDB in DevOps automatic operation and maintenance. Information and Computer, vol. 32,no. 11, pp. 4, 2020.

[4]. Tong Xiangjie, Zheng Wu, Xie Fengling, et al. Hardware DevOps Practice in Digital Transformation of Enterprises. Value Engineering, vol. 39, no. 1, pp. 5, 2020.

[5]. Tang Songqiang, Cai Yongjian, Tang Haitao, et al. DevOps Construction Research and Practice. Computer Age, no. 4, pp. 5, 2021.

[6]. Li Yang. Application of DevOps in Team Work. Digital Design, vol. 10, no. 3, pp. 2, 2021.

[7]. Lu Gang, Chen Changyi, Huang Zelong, et al. Research on intelligent cloud Native Architecture and Key Technologies for Cloud Network Convergence. Telecommunications Science, vol. 36, no. 9, pp. 8, 2020.

[8]. Xue Long, Lu Gang, Zhou Qi, et al. Cloud-native-oriented intelligent operation and maintenance architecture and key technologies, no. 12, pp. 105-112, 2021.

[9]. Liu Weiguang. Application of financial grade cloud native distributed architecture. China Finance, no. 6, pp. 3, 2022.

[10]. A. Phillips, M. Sens, A. de Jonge, and M. van Holsteijn, The IT Manager's Guide to Continuous Delivery: Delivering business value in hours, not months: XebiaLabs, 2015.

[11]. Ishaq Azhar Mohammed. (2013). Intelligent authentication for identity and access management: a review paper. International Journal of Managment, IT and Engineering (IJMIE), 3(1), 696–705. Retrieved from https://www.ijmra.us/project doc/IJMIE_JANUARY2013/IJMIEJan13Ishaq.pdf

[12]. Sudhir Allam, "Significance of developing Multiple Big Data Analytics Platforms with Rapid Response", International Journal of Emerging Technologies and Innovative Research (www.jetir.org), ISSN:2349-5162, Vol.2, Issue 3, page no.777-783, March-2015, Available :http://www.jetir.org/papers/JETIR1701523.pdf

[13]. R. Conejo, M. Urretavizcaya and J. Prez-de-la-Cruz, Current topics in artificial intelligence. Berlin: Springer, 2004.\

[14]. V. Sugumaran, Distributed artificial intelligence, agent technology and collaborative applications. Hershey, PA: Information Science Reference, 2009.

[15]. H. Salzman, "Engineering perspectives and technology design in the United States", AI & Society, vol. 5, no. 4, pp. 339-356, 1991.

[16]. H. Izadkhah, "Transforming Source Code to Mathematical Relations for Performance Evaluation", Annales Universitatis Mariae Curie-Sklodowska, sectio AI – Informatica, vol. 15, no. 2, p. 7, 2015.

[17]. T. Bradley and T. Bradley, "Why DevOps means the end of the world as we know it", TechSpective, 2016. [Online]. Available: https://techspective.net/2015/08/16/why-devops means-the-end-of-the-world-as-we-know-it/

[18]. D. Linthicum, "What is DevOps? DevOps Explained | Microsoft Azure", Azure.microsoft.com, 2016. [Online]. Available: https://azure.microsoft.com/en-us/overview/what-is devops/