International Journal of Core Engineering & Management
Volume-7, Issue-12, 2024 ISSN No: 2348-9510

ADVANCEMENT IN CLOUD INFRASTRUCTURE AUTOMATION: A SURVEY OF CURRENT TECHNOLOGIES AND FUTURE DIRECTIONS

Srinivasa Rao Singireddy
Architect, The Clorox Services Company, Pleasanton, CA.
srinivasarao.singireddy@gmail.com

Abstract

Automation is another essential aspect since more and more/this field is steadily growing to consider cloud computing as the backbone of the contemporary world. Cloud infrastructure automation consequently assures fast adaption to sophisticated demands through efficient resource management and well-coordinated processes. This paper seeks to define and discuss the emerging trends in cloud infrastructure automation based on flexibility needed in a cloud setup. Automated solutions require less interaction with people through methods that provide provisions, configurations and management of service that help to ensure high availability and less operational expense. The paper focuses on serverless architectures and the use of the FaaS and IaaS hybrid approach, where the primary concern is the efficient communication between the two models. It talks about new challenges arising from the use of multiple cloud configurations and requires new forms of management including Kubernetes and Terraform. It brings hope of predictive and self-healing systems where AI and ML dominate automation, turning the management of the cloud into more automation than ever before. This paper offers a clear understanding of all forms of automating cloud infrastructure, includingIaC, configuration management, continuous integration/continuous deployment, and containerisation. Thus, by analysing these technologies' impact on efficiency and reliability, this study provides a comprehensive reference for companies interested in transitioning toward and managing cloud automation systems.

Keywords: Cloud computing, Infrastructure automation, Resource management, Scalability, IaaS.

I. INTRODUCTION

The automation of cloud infrastructure has evolved quickly in the past few years to meet the high investment and strong market need for the ability to scale and global space for comprehensive system solutions. The concept of automation in a cloud setting refers to the use of technologies to achieve high levels of self-service in the areas of provisioning, configuration and management[1]. This push toward automation has been propelled by the need to separate service availability from operational costs and risks, as well as to meet compliance requirements in the evolving landscape of the cloud environment[2][3].

The transition from homogeneous models of clouds to serverless, edge computing, and a combination of both has defined new forms of cloud automation. Specifically, the serverless architecture model has garnered a lot of attention.It allows organisations to run code while the underlying infrastructure is taken care of by someone else. This change of paradigm makes it easier to manage tasks that are brief and stateless, which in turn makes broader use of computing resources more manageable and cost-effective. Some have urged scholars to expand their knowledge of the organisation of workflows that combine FaaS and IaaS on various platforms[4].

The current studies also focus on the increased complexity of multi-cloud environments in organisations. This makes it even more challenging to balance distributed computing across the cloud or even have the ability to automate Hadoop and other cross-cloud activities. Tools like Kubernetes and Terraform are seen as important tooling, providing the management and infrastructure for resources across multiple cloud providers. Thirdly, automation improves cloud security by adding constant check fixtures and auto-pilot security response systems that counter threats on a real-time basis[4].

In the future, artificial intelligence and machine learning in cloud automation are prospects for anticipative as well as for self-sustainable clouds. These intelligent automation systems can predict problems before they materialise and solve them without human intervention, thus reducing disturbances. This kind of development is set to transform the management of cloud since such systems will be able to run on their own with limited interference by people[5].

A. Purpose of the Study

This study seeks to identify and discuss available technologies to automate cloud infrastructure management today. It is to establish what has been accomplished, to evaluate automation benefits, the extent to which automation reduces occurrence of human interferences, and the extent to which it strengthens cloud reliability besides eclipsing challenges and future strategies in automation of clouds. The main key points of the paper are listed below:

- The research comes up with a general overview of data migration to enable organisations to determine the challenges that come with the process and different types of data migration.
- It compares the leading cloud service providers and their migration tools and demystifies the choices for efficient cloud migrations.
- The study identifies key challenges and risks associated with cloud data migration, allowing organisations to proactively tackle potential issues.
- By delineating the stages of migration and essential drivers, the study serves as a practical guide for organisations embarking on cloud data migration projects.

B. Structure of the Study

The paper is structured as follows: Section II provides an overview of cloud computing technologies. Section III discusses cloud infrastructure automation. Section IV explores related technologies and their impact. Section V examines the previous studies of cloud computing. Finally, Section VI offers conclusions and future work.

II. CLOUD COMPUTING OVERVIEW WITH TECHNOLOGIES

The term "cloud computing" refers to a way that computing resources such as servers, networks, storage, applications, and services can be shared and made available quickly with little to no management work or involvement from service providers. Any individual or entity that uses Cloud computing to complete a job, take part in a transaction or both is considered an actor. Here are the

International Journal of Core Engineering & Management
Volume-7, Issue-12, 2024 ISSN No: 2348-9510

five key players: Users, Cloud Service Providers, Cloud Brokers, Cloud Carriers, and Cloud Auditors[6].The Figure 1 represents the cloud computing-based models.

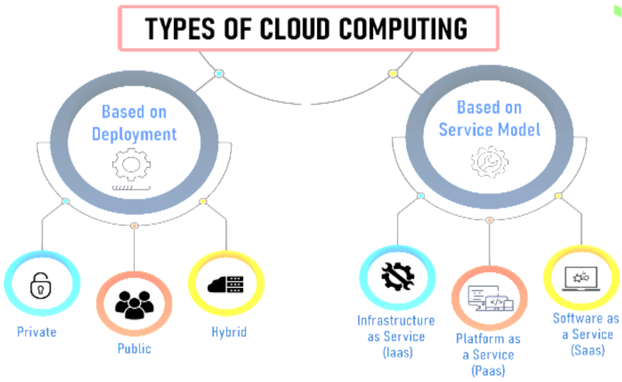

Figure 1: Cloud computing-based Models

The practicality and accessibility of distributed computing are made possible by several types of administrations and models that operate behind the scenes. Figure 1 also demonstrates that there are primarily two types of distributed computing models:

- **Deployment models:** The deployment model defines the access type to the cloud environment, or how is the cloud located? The four different kinds of cloud access are "public," "private," "hybrid," and collective.
- **Service models:** A cloud's service capabilities allow it to communicate with clients or applications in several ways [3]. Platform as a Service (PaaS), Infrastructure as a Service (IaaS), and Software as a Service (SaaS) are the service paradigms upon which cloud computing is built. There are now three main categories of services or service models available on the web.
- **Infrastructure as a service (IaaS):** Computers, both real and virtual, as well as devices to manage extra capacity, are all part of what cloud computing companies have to offer. A network of operational, emotionally supporting nodes controls the hypervisors, which in turn manage the virtual computers. It is the responsibility of the cloud client to install application programming and operating system images on the virtual machines. Infrastructure as a service makes it feasible for cloud providers to discover infrastructure freely via the Internet. IaaS allows users to rent resources such as storage, bandwidth, monitoring services, IP addresses, firewalls, VMs, and more. Time allotment is a key factor in how much a customer will pay for a resource. Examples: Windows Azure, Amazon EC2, Rackspace, Google Compute Engine.AFigure 2 shows the Cloud Infrastructure as a Service (IaaS).

International Journal of Core Engineering & Management
Volume-7, Issue-12, 2024 ISSN No: 2348-9510

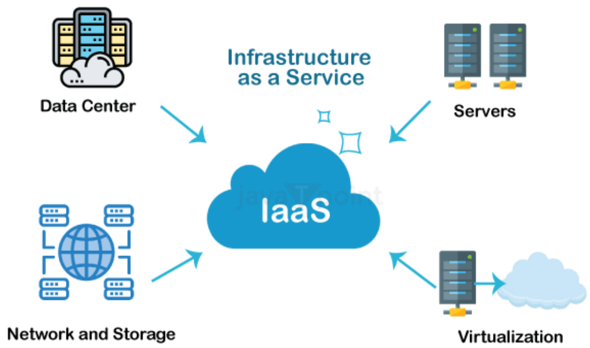

Figure 2: Cloud Infrastructure as a service (IaaS)

A. Cloud Computing Technologies

Behind the scenes of the cloud computing platform are a number of advancements that contribute to its dependability, adaptability, and usability. These developments include [7]:

1) **Virtualization:** The primary goal of this technology is to provide cloud customers with standardised versions of apps. An example would be a cloud provider's responsibility to provide its customers with the most recent version of an application whenever an update is made. As an example, virtualised IT frameworks may be requested from VMware and Xen. Innovations in virtual systems, such as Virtual Private Networks (VPNs), allow customers with altered network settings to access resources in the Cloud.
2) **SOA:** SOA is an application architecture that breaks down common commercial apps into smaller, more manageable pieces called services. Organisations may access cloud-based registration agreements with customisable features thanks to this aspect of cloud technology, which can be tweaked as needed to meet evolving business demands.
3) **Grid Computing:** The various computing resources are located in different parts of the world. Grid Computing is a solution that divides a big issue into smaller ones. Consequently, systems located inside the grid get these smaller components. The purpose of a grid system is to facilitate resource sharing via distributed and massively parallel computing.
4) **Utility Computing:** Cloud computing pricing is sometimes more complicated than it seems. The initial investment may be reduced with the aid of utility computing. There are no hidden fees; instead, the pricing adapts to the changing computing needs of individuals or businesses. Billing will decrease in proportion to the decrease in consumption.

B. Cloud Automation

Even with partial or mixed automation, this fundamental concept of automation will work. For instance, in the present day, Uber uses automated high-level decision and action selection to calculate the routes and waypoints for each car, but it still depends on human drivers to ensure a safe journey between each waypoint. It is reasonable to assume that fully autonomous Uber cars would be able to safely execute all actions necessary to travel the chosen route without any human intervention [8].

- **Data acquisition:** Automated processes record information about the context, operating state, and other aspects. For instance, an autonomous car gathers GPS information in addition to other sensor inputs, such as direction, wheel rotation speed, other cars' whereabouts, and so forth.

- **Information analysis:** The data is analysed by automated processes to profvide information on the operating environment, the vehicle's position and velocity, and other current condition details.
- **Decision and action selection:** Choosing the proper course of action. To prevent accidents with other cars, people, and objects, an autonomous vehicle must determine the best route to take from its present position to the intended destination.
- **Action implementation:** Executing the selected action. For instance, an autonomous car has to steer, brake, and accelerate in order to go safely from one waypoint to the next.

III. OVERVIEW OF CLOUD INFRASTRUCTURE AUTOMATION

Cloud infrastructure automation refers to the process of managing and provisioning cloud resources automatically using tools and scripts, reducing the need for manual intervention. It encompasses tasks like server provisioning, scaling, monitoring, configuration management[9], and networking in cloud environments such as AWS, Microsoft Azure, or Google Cloud. Infrastructure as Code (IaC) frameworks such as Terraform, AWS CloudFormation, or Ansible enable engineers to declaratively describe infrastructure, hence facilitating automation. In cloud infrastructure automation, important ideas and resources include:

A. Infrastructure as Code (IaC)

The process of automatically configuring system dependencies and provisioning local and remote instances is known as infrastructure as code, or IaC. IaC is seen by practitioners as a key component of DevOps processes, enabling them to quickly deliver software and services to end customers. IaC has been embraced by information technology (IT) companies including GitHub, Mozilla, Facebook, Google, and Netflix. Researchers may uncover possible IaC-related research fields by conducting a thorough mapping analysis on the body of current IaC research, for instance. Potential security vulnerabilities and defects in IaC scripts[10][11].

B. Software Configuration Management

Constant change is a defining feature of the software evolution process. Working towards a similar objective, a group of individuals often creates, modifies, and trades both individual and shared software components. Introducing the principles together named "software configuration management" (SCM) is essential to maintaining control over all multi-version, multi-person operations. There is a great deal of overuse of terminology since SCM has been studied in various software development fields. To help with this, we have provided the following definitions [12]:

- **Configuration item:** Any uniquely identifiable piece of software, such as documents, modules, and libraries, is referred to as a configuration item.
- **Version:** A configuration item has one unique instance called a version. For the administration of configuration elements, it offers a reliable reference system. Every version save the ones that are now functioning is frozen, meaning that they cannot be modified without generating a new version.
- **Revision:** The development of a configuration item may be traced back to revisions, which are versions that replace older ones and may include arbitrary changes as compared to

International Journal of Core Engineering & Management
Volume-7, Issue-12, 2024 ISSN No: 2348-9510

their predecessor. It is usually made to support long-term modifications or to address issues.

- **Variant:** Config items may have variants, which are different versions of the same thing. Created to accommodate for variations in the environment, they are considered legitimate at a certain point in time. What makes up the versionset is a set of different configuration pieces that have been selected from certain revision levels.
- **Release:** A customer-ready version set that has passed certain established quality assurance tests is called a release [13].

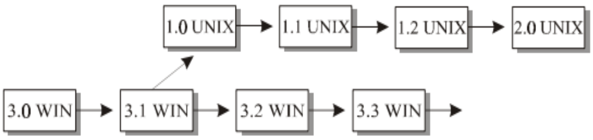

Figure 3: Configuration items, versions, variants, revisions, releases

The boxes in Figure 3. depict the individual configuration elements in a typical version graph. A piece of software is now available in Windows 3.0 (WIN). The versions are shown by the diagonal arrows (1.0 UNIX) and the revisions by the horizontal arrows (3.1 WIN - 3.2 WIN - 3.3 WIN - 3.4 WIN). The current version, 2.0 UNIX, is the result of a series of updates from 1.1 UNIX to 1.2 UNIX[14].

C. CI/CD Integration

The term "Continuous Integration" (CI) refers to a method of integrating code changes more often rather than saving them for the final stage of development. Continuous Delivery refers to a software architecture that allows for software distribution at any moment to the market. The issue of finding and fixing the defect quickly may be greatly mitigated with continuous integration and continuous delivery. As the time it takes to identify and repair defects decreases, an organisation may make more releases that stick to the stated timeframe.

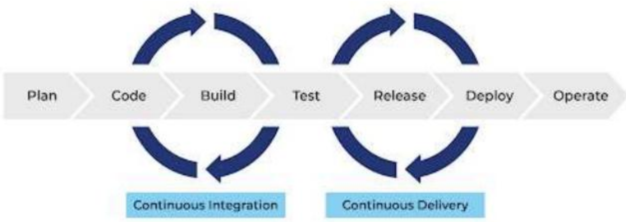

Figure 4: Steps in CI/CD

Continuous Integration/Deployment (CI/CD) is a procedure that is repeated indefinitely and involves constantly integrating changes, creating, testing, and deploying. The information is shown in Figure 4 [15].

D. Container Orchestration

Because of its cheap overhead and quick deployment, containers provide an excellent way to implement such microservices. Moreover, they are appropriate for effective horizontal scaling, which may be accomplished by putting in place several similar containers. As such, hundreds or even thousands of services with several (perhaps complicated) interdependencies may make up a contemporary complex application. The creation of a higher containerisation layer known as container orchestration was warranted given the current design and deployment trends, which suggest that using container solutions for big applications may prove challenging to embrace. The scheduling, booking, and accounting of resources as well as the coordination and communication across microservices are all automated by container orchestrator engines, also known as container orchestrators.Nowadays, there are a number of container orchestrators on the market, including Docker Swarm, Kubernetes, and Mesos (to name a few of the more popular ones), after ten years of research and development in container technology [16].

E. Auto-scaling and Load Balancing

Among the enhancements that support the sustenance of a customer's cloud subscription system's availability is auto-scaling. There is a problem that has to be taken into consideration when an auto-scaling mechanism appears in a cloud system with several established system mechanisms. Anytime a new component is introduced to a stable system, there are often no free disadvantages. Equally distributing the loads among the virtual computers in a data centre is the primary objective of a load-balancing system[17]. It makes sure that no virtual machine is using up significantly more resources than the others. Incidentally, a load balancer reduces latency in user request processing and optimizes cloud performance [18].

IV. TECHNOLOGIES IN CLOUD INFRASTRUCTURE AUTOMATION

The automation of cloud infrastructure is the process of automating the deployment, management, and optimisation of cloud resources via the use of different tools and technologies. In addition to boosting operational efficiency, these technologies make cloud systems more agile and scalable [19]. Some of the most important technologies for automating cloud infrastructure are listed below:

A. Infrastructure as Code (IaC)

The ability to provide and control cloud infrastructure using code is made possible by the foundational technology known as Infrastructure as Code (IaC). Automating the configuration and setup of cloud services may be achieved by specifying infrastructure in code. This allows organisations to ensure consistency and repeatability. Developers may streamline deployment procedures and reduce the likelihood of human mistakes with the use of tools like Terraform, AWS CloudFormation, and Ansible, which allow them to version control and test changes to infrastructure [20].

B. Containerization and Orchestration

Developers are able to bundle apps and their dependencies together thanks to containerisation technologies like Docker, which enable programs to operate in separate environments. Deploying and scaling are made easier using this method. Kubernetes and OpenShift are two examples of orchestration systems that automate the deployment, scaling, and administration of containerised applications. This allows for effective management of many containers, which in turn ensures high availability and optimum resource utilisation.

C. Configuration Management Tools
System and application settings may be easily and efficiently managed with the help of configuration management solutions. These tools, such as Chef, Puppet, and Ansible, ensure that all servers and applications are consistently configured according to predefined policies. This helps maintain compliance, reduces configuration drift, and allows for rapid recovery in case of system failures[21].

D. Continuous Integration and Continuous Deployment (CI/CD)
CI/CD procedures streamline the software development lifecycle, allowing for faster application development, testing, and deployment. Tools like Jenkins, GitLab CI/CD, and CircleCI facilitate automated testing and deployment pipelines, ensuring that changes to applications can be released reliably and frequently without manual intervention[22].

E. Monitoring and Logging Solutions
The performance and health of cloud infrastructure depend on technology for logging and monitoring. Real-time insights into the behaviour and performance of cloud resources may be obtained using tools such as Splunk, Grafana, and Prometheus. Automated alerts and dashboards help teams quickly identify and address issues, optimising resource allocation and enhancing operational resilience[23].

F. Service Mesh
An infrastructure layer called a service mesh helps cloud-native applications' microservices communicate with one another. Technologies like Istio and Linked provide automated routing, load balancing, service discovery, and observability. This helps in managing complex microservice architectures, ensuring that services can communicate securely and reliably without extensive manual configuration.

G. Serverless Computing
Using serverless computing, programmers can launch apps without worrying about the underlying infrastructure. Websites such as Google Cloud Functions, AWS Lambda, and Azure Functions automatically handle the necessary resource scaling and management according to demand. Developers are free to concentrate on coding rather than server administration with this method's reduced operational overhead [20].

V. LITERATURE REVIEW
This paper, Alshareef (2023),offers an extensive overview of cloud computing. A broad overview of cloud computing is provided, followed by a discussion of its benefits, present state, obstacles, and potential future directions. After that, they cover cloud computing in-depth, including its designs, models, fault tolerance, strategies for selecting services, industry acceptance, and scheduling of cloud-based resources[24].
This paper, Banz, (2019) fills this need by creating an innovative approach to requirement engineering tailored to cloud and automation applications. The best requirements elicitation methodologies were iteratively blended after an assessment. The iterative nature of the process and the combination of elicitation methods (such as interviews, workshops, surveys, and reuse)

allow it to continually verify that the requirements are current, comprehensive, and viable. The intended product or service may then be developed in a methodical manner[25].

This paper,Garg and Garg (2019) gives a high-level summary of cloud computing and how it is changing the face of industrial automation in the future, with the goals of increasing efficiency and optimising costs. Automated resources may be efficiently managed, provided quickly, and released with little administrative work; this study also discusses a method for successful deployment and maintenance[26].

This paper, Prassanna, Pawar and Neelanarayanan, (2017) descriptions of the main obstacles, such as moving the apps to the cloud. Time and effort are wasted due to downtime during the configuration and management of the more complicated infrastructure after application migration to the cloud. Therefore, it is necessary to automate this environment. Some automation technologies in the cloud are necessary to design distributed systems in a way that supports security, repeatability, stability, and scalability. Tools for automating infrastructure, such Terraform and cloud formation, and applications, like Docker and Habitat, are summarised in this article[27].

This paper, Luthra et al., (2021)offers one such approach that makes use of the storage facilities offered by three prominent public cloud providers. This initiative is intended to fill the need for an open-source solution that works with several cloud platforms, since none currently exist. To access the platform's APIs, you may use the native SDK libraries that come with each cloud service. The purpose of these API calls is to check whether the storage services' current configuration has changed, or at least is in use. The fix allows for the integration of all libraries into one integrated driver application. Automation, regular security and audit checks, and the ability to manage storage accounts across various hybrid cloud services are all within the program's purview[28].

This paper, Ghule and Gopal, (2020), focuses on studying storage service components and interfaces that are applicable to cloud computing, cloud-for-storage, and storage-for-cloud. Then, it makes a generic list of interfaces that can be used for architecting automation, which addresses the need to make a generic choice when it comes to cloud engineering process automation and reusable cloud solutions[29].

This paper, Kamath et al. (2023) proves it is possible to build a solid framework that accomplishes the provisioning, testing, and continuous integration of cloud resources. Using the idea of a build pipeline, the framework may automate a number of processes, including code analysis, packaging, deployment, notifications, infrastructure provisioning, and test execution. This sparks the idea of creating a straightforward software solution that integrates with the CI/CD pipeline, utilises GitHub Actions, has a simple frontend, and uses a dashboard to monitor the usage of provisioned resources. The goal is to speed up the process of cloud infrastructure provisioning and ensure that it happens quickly [30].

VI. CONCLUSION AND FUTURE WORK

While automation in cloud infrastructure management introduces several challenges, the overarching benefits— enhanced efficiency, cost reduction, improved security, and better resource management— strongly advocate for its adoption. Automating tasks is becoming more and more necessary for businesses that want to remain competitive in the ever-changing digital market. Finally, an extensive examination of cloud data migration methodologies and their applicability in different situations is provided by this paper. It highlights the critical factors influencing successful

migration, including data integrity, security, and the choice of migration tools. The insights gained from this analysis can assist organisations in making informed decisions that enhance the efficiency and reliability of their cloud data migration processes.
Future research should focus on developing more advanced migration techniques that leverage AI and ML to automate and optimise the migration process further. Additionally, exploring the impact of emerging cloud technologies and frameworks on migration strategies can provide valuable insights into the evolving landscape of cloud computing.

REFERENCES

1. A. P. A. Singh, "STRATEGIC APPROACHES TO MATERIALS DATA COLLECTION AND INVENTORY MANAGEMENT," Int. J. Bus. Quant. Econ. Appl. Manag. Res., vol. 7, no. 5, 2022.
2. H. B. Hassan, S. A. Barakat, and Q. I. Sarhan, "Survey on serverless computing," 2021. doi: 10.1186/s13677-021-00253-7.
3. S. G. Ankur Kushwaha, Priya Pathak, "Review of optimize load balancing algorithms in cloud," Int. J. Distrib. Cloud Comput., vol. 4, no. 2, pp. 1–9, 2016.
4. N. A. Angel, D. Ravindran, P. M. D. R. Vincent, K. Srinivasan, and Y. C. Hu, "Recent advances in evolving computing paradigms: Cloud, edge, and fog technologies," 2022. doi: 10.3390/s22010196.
5. H. Shafiei, A. Khonsari, and P. Mousavi, "Serverless Computing: A Survey of Opportunities, Challenges, and Applications," ACM Comput. Surv., 2022, doi: 10.1145/3510611.
6. M. De Donno, K. Tange, and N. Dragoni, "Foundations and Evolution of Modern Computing Paradigms: Cloud, IoT, Edge, and Fog," IEEE Access, 2019, doi: 10.1109/ACCESS.2019.2947652.
7. M. I. Malik, "CLOUD COMPUTING-TECHNOLOGIES," Int. J. Adv. Res. Comput. Sci., vol. 9, no. 2, pp. 379–384, Apr. 2018, doi: 10.26483/ijarcs.v9i2.5760.
8. F. Bauer, "Cloud Automation and Economic Efficiency," IEEE Cloud Comput., 2018, doi: 10.1109/MCC.2018.022171664.
9. H. Mullangi et al., "AI-Augmented Decision-Making in Management Using Quantum Networks," Asian Bus. Rev., vol. 13, no. 2, pp. 73–86, 2023, doi: 10.18034/abr.v13i2.718.
10. A. Rahman, R. Mahdavi-Hezaveh, and L. Williams, "A systematic mapping study of infrastructure as code research," Inf. Softw. Technol., 2019, doi: 10.1016/j.infsof.2018.12.004.
11. V. K. Yarlagadda and R. Pydipalli, "Secure Programming with SAS: Mitigating Risks and Protecting Data Integrity," Eng. Int., vol. 6, no. 2, pp. 211–222, Dec. 2018, doi: 10.18034/ei.v6i2.709.
12. A. Bartusevics and L. Novickis, "Models for implementation of software configuration management," in Procedia Computer Science, 2015. doi: 10.1016/j.procs.2014.12.002.
13. Magnusson, U. Asklund, and S. Minör, "Fine-grained revision control for collaborative software development," ACM SIGSOFT Softw. Eng. Notes, 1993, doi: 10.1145/167049.167061.
14. L. BENDIX, A. DATTOLO, and F. VITALI, "SOFTWARE CONFIGURATION MANAGEMENT IN SOFTWARE AND HYPERMEDIA ENGINEERING: A SURVEY," 2001. doi: 10.1142/9789812389718_0022.

15. A. S.K, A. Amrathesh, and D. G. Raju M, "A review on Continuous Integration, Delivery and Deployment using Jenkins," J. Univ. Shanghai Sci. Technol., vol. 23, no. 06, pp. 919–922, Jun. 2021, doi: 10.51201/JUSST/21/05376.
16. H. M. Al Jawarneh et al., "Container Orchestration Engines: A Thorough Functional and Performance Comparison," in IEEE International Conference on Communications, 2019. doi: 10.1109/ICC.2019.8762053.
17. K. Patel and I. Researcher, "Quality Assurance In The Age Of Data Analytics : Innovations And Challenges," vol. 9, no. 12, 2021.
18. N. Hong Son, "Load Balancing in Auto Scaling Enabled Cloud Environments," Int. J. Cloud Comput. Serv. Archit., 2017, doi: 10.5121/ijccsa.2017.7502.
19. H. Thomas, "Enhancing Supply Chain Resilience Through Cloud-Based SCM and Advanced Machine Learning: A Case Study of Logistics," J. Emerg. Technol. Innov. Res., vol. 8, no. 9, 2021.
20. X. L. Wang, L. Wang, Z. Bi, Y. Y. Li, and Y. Xu, "Cloud computing in human resource management (HRM) system for small and medium enterprises (SMEs)," Int. J. Adv. Manuf. Technol., 2016, doi: 10.1007/s00170-016-8493-8.
21. M. Raquibul Hasan, M. Sifuddin Ansary, and N. Biz, "Cloud Infrastructure Automation Through IaC (Infrastructure as Code)," Int. J. Comput., 2023.
22. K. Patel, "An Analysis of Quality Assurance Practices Based on Software Development Life Cycle (SDLC) Methodologies," J. Emerg. Technol. Innov. Res., vol. 9, no. 12, pp. g587–g592, 2022.
23. N. Shetty and D. K. H. K., "Cloud Infrastructure Automation Tools: A Review," J. Univ. Shanghai Sci. Technol., 2021, doi: 10.51201/jusst/21/05397.
24. H. N. Alshareef, "Current Development, Challenges and Future Trends in Cloud Computing: A Survey," Int. J. Adv. Comput. Sci. Appl., 2023, doi: 10.14569/IJACSA.2023.0140337.
25. A. Banz, "Requirements engineering method for infrastructure automation and cloud projects," in Proceedings of the IEEE International Conference on Requirements Engineering, 2019. doi: 10.1109/RE.2019.00037.
26. S. Garg and S. Garg, "Automated Cloud Infrastructure, Continuous Integration and Continuous Delivery using Docker with Robust Container Security," in Proceedings - 2nd International Conference on Multimedia Information Processing and Retrieval, MIPR 2019, 2019. doi: 10.1109/MIPR.2019.00094.
27. J. Prassanna, A. R. Pawar, and V. Neelanarayanan, "A review of existing cloud automation tools," Asian J. Pharm. Clin. Res., vol. 10, no. July, pp. 471–473, 2017, doi: 10.22159/ajpcr.2017.v10s1.20519.
28. J. Luthra, A. Choudhary, A. Sharma, and D. N. K. Jayakody, "ARUSH:Automated Runtime Universal Scanner for Hybrid Cloud computing," in 2021 IEEE Globecom Workshops, GC Wkshps 2021 - Proceedings, 2021. doi: 10.1109/GCWkshps52748.2021.9682093.
29. D. Ghule and A. Gopal, "Addressing the Need to generalize choice of Interfaces for Cloud Engineering Process Automation: Withr reference to IaaS - Storage," in International Conference on Electrical and Electronics Engineering, ICE3 2020, 2020. doi: 10.1109/ICE348803.2020.9122900.
30. S. Kamath, M. M. Manohara Pai, S. Vignesh, and G. Darshan, "Revolutionizing Cloud Infrastructure Management: Streamlined Provisioning and Monitoring with Automated

Tools and User-Friendly Frontend Interface," in 2023 3rd International Conference on Intelligent Technologies, CONIT 2023, 2023. doi: 10.1109/CONIT59222.2023.10205728.