Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

DEVOPS TECHNOLOGIES IN THE SOFTWARE DEVELOPMENT LIFE CYCLE

Ulugbek Urunov
Software Engineer, USA

Abstract

This article examines DevOps technologies as a modern approach to organizing the software development lifecycle. It analyzes the origins of the DevOps concept, its stages of formation and development, as well as the key principles and tools used in the development, testing, deployment, and maintenance of software products. Particular attention is paid to process automation, continuous integration, continuous delivery, and collaboration between development and operations teams. The article discusses the benefits of using DevOps to improve software quality, reduce time to market, and enhance the reliability of information systems. It concludes that the implementation of DevOps technologies is a key factor in the successful digital transformation of modern organizations.

Scientific Novelty. The scientific novelty of this study lies in its comprehensive analysis of the application of DevOps technologies across all stages of the software development lifecycle, taking into account current trends in digital transformation. The study systematizes the key principles and tools of DevOps, revealing their role in automating the development, testing, deployment, and maintenance of software products. The study substantiates the impact of DevOps technology integration on improving the efficiency of software lifecycle management, reducing the time it takes to release new versions, improving the quality of software products, and ensuring the continuous improvement of development processes in modern information systems.

Purpose of the Study. The purpose of this study is to analyze DevOps technologies and assess their role in improving the software development lifecycle, as well as to identify the specific features of using modern automation tools to improve the quality of software products, reduce development time, and

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

enhance the efficiency of information system development, testing, deployment, and maintenance processes.

Keywords: DevOps, software life cycle, software development, automation, continuous integration, continuous delivery, CI, CD, containerization, cloud technologies.

Introduction

Almost all modern information systems, digital services, and technology platforms rely on software. Requirements for software are constantly growing: high development speed, application reliability, and service quality are essential. Therefore, software lifecycle management methods have had to be refined. Companies want to reduce the time from product development to launch. At the same time, it is crucial not to compromise stability and security. In this situation, DevOps technologies are particularly in demand. They integrate development, testing, deployment, and maintenance into a single, continuous process [1].

The history of software development methods dates back to the middle of the last century. In the 1960s, programming was primarily performed by small teams of specialists who simultaneously designed, developed, and maintained the programs they were creating. As software products grew in complexity, responsibilities were divided between developers and operations specialists. This division allowed for greater specialization among employees, but it also led to organizational barriers, a slower release rate, and an increased rate of errors during the transition of software products to production.

In the 1970s and 1980s, waterfall software development lifecycle models became widespread. Each stage was completed sequentially, beginning with requirements analysis and ending with maintenance of the completed system. Despite the highly structured process, this approach lacked flexibility and was poorly adapted to rapidly changing customer requirements.

In the 1990s, agile development management methods began to actively develop. The emergence of the Agile concept marked a significant step in the evolution of software engineering. The focus shifted to continuous customer interaction, short development cycles, and regular releases of new software

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

versions. However, even the adoption of Agile did not completely eliminate the problem of collaboration between development and operations teams, as implementation and maintenance processes remained isolated.

The preconditions for the emergence of DevOps were finally established in the early 2000s. The spread of virtualization, cloud computing, automated testing, and configuration management systems had a significant impact. In 2009, Belgian specialist Patrick Debois organized the first DevOpsDays conference, which is considered the starting point for the official development of the DevOps concept. It was after this that the term DevOps gained widespread acceptance in the professional community and began to be actively adopted by large international companies.

The modern DevOps concept combines not only tools but also a distinctive culture. It focuses on continuous collaboration between developers, administrators, testers, and security engineers. This approach significantly shortens the development lifecycle, improves product quality, and ensures continuous system improvement [2].

The rapid development of containerization, cloud platforms, microservice architecture, and artificial intelligence technologies is further expanding the capabilities of DevOps. Automating most software lifecycle operations allows organizations to more quickly adapt to market changes, utilize computing resources more efficiently, and ensure high availability of digital services.

In today's digital economy, DevOps technologies are becoming one of the most sought-after areas of software engineering development. Their use allows for the unification of all stages of the software development lifecycle into a single, continuous process based on automation, shared responsibility among project participants, and the continuous improvement of software product quality.

It would be appropriate to devote the remainder of this article to the theoretical foundations of DevOps, principles of organizing the software development life cycle, modern automation tools, and an analysis of practical experience in implementing DevOps technologies in various organizations.

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

Theoretical Foundations of DevOps Technologies. DevOps technologies are a modern methodology for organizing software development, testing, deployment, and maintenance processes, based on close collaboration between all participants in the software product lifecycle. The main goal of DevOps is to create a unified environment in which development and operations processes are viewed as interconnected elements of a unified software management system.

The term DevOps is formed by combining the English words "development" and "operations," reflecting the integration of the activities of developers and information systems operations specialists. Unlike traditional development models, in which each team performs a strictly limited set of functions, DevOps assumes collective responsibility for the quality of the software product throughout its entire lifecycle [3].

As researchers note, "one of the key features of DevOps is the transition from sequential development stages to a continuous process of creating, testing, implementing, and maintaining software. By automating most operations, the human factor is significantly reduced, the speed of releasing new versions increases, and the number of production errors is reduced" [4].

The modern DevOps concept is based on several fundamental principles. First and foremost, process automation is crucial. Software builds, test execution, code quality checks, infrastructure management, application deployment, and information system performance monitoring are automated. Automation ensures repeatability of all operations and minimizes the likelihood of errors.

Developers continuously push their changes to a shared repository. The build and tests are then automatically run. This approach helps identify errors early, making them much cheaper to fix. This is the principle of continuous integration.

Continuous delivery is equally important. Once all tests are successfully completed, the new version of the program is automatically prepared for deployment to the production environment. If necessary, the release can be made with minimal human intervention [5, 6, 7].

Another key element of DevOps is continuous monitoring of software performance. Modern systems track application performance, computing

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

resource utilization, error rates, service availability, and many other parameters. The collected data helps quickly fix problems and improve the product [8].

A collaborative culture is particularly important. Within DevOps, specialists from different departments actively collaborate, make decisions together, and share responsibility for project outcomes. This approach promotes effective communication and accelerates the completion of production tasks.

The spread of cloud computing has played a significant role in the development of DevOps. Cloud infrastructure enables the rapid creation of new computing resources, scaling applications based on workload, and efficient use of hardware without the need to purchase expensive servers.

Along with the development of cloud technologies, containerization has become widespread. Containers allow applications to run in an isolated environment, regardless of operating system and hardware characteristics. This ensures consistent software functionality throughout development, testing, and production.

Another area of DevOps development is managing infrastructure as software code. All server, network equipment, and software environment parameters are described as configuration files. This approach significantly simplifies infrastructure replication, automates the deployment process, and ensures change control.

Modern DevOps technologies are closely linked to the use of microservices architecture. Instead of a single large software suite, a collection of independent services is created, each performing a distinct function. This allows for independent updates of various system components, accelerates the development of new features, and increases software resilience.

An important area of development for this concept is the integration of information security tools directly into the development process. This approach is known as DevSecOps. Security checks are performed automatically at all stages of the software lifecycle, from source code analysis to monitoring the security of running services. This enables the timely identification of potential vulnerabilities and significantly reduces the risk of information security incidents.

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

DevOps is a comprehensive technology for organizing the software development lifecycle, integrating modern methods of automation, infrastructure management, quality control, and teamwork. This concept helps improve development efficiency, reduce the time it takes to release new software versions, and increase the reliability of information systems.

Applying DevOps technologies to the software development lifecycle. The software development lifecycle is a sequence of interrelated stages, beginning with requirements generation and ending with the maintenance of the finished software product. In traditional development models, transitions between stages were often time-consuming, requiring numerous manual operations and a high risk of errors. Using DevOps technologies allows for the integration of all lifecycle stages into a single automated process, ensuring high development speed and stable software operation.

Requirements analysis and project planning begin the lifecycle. At this stage, functional and non-functional requirements are defined. The architecture of the future system is formed, tasks are allocated, and technologies are selected. DevOps does not directly change the methods of requirements analysis. However, it does promote closer collaboration between analysts, developers, testers, and operations specialists. When architectural decisions are discussed together, it allows for the early consideration of requirements for scalability, performance, reliability, and software security [9].

The next stage is code development. To effectively organize collaboration, version control systems are used, providing centralized source code storage and enabling simultaneous work by a large number of developers. All changes are recorded in the repository, allowing for tracking development history, quickly identifying errors, and reverting to previous versions of the project if necessary. Continuous integration processes are automatically launched after changes are made to the software code. Each update undergoes automated builds, dependency checks, and tests. If errors are detected, developers are notified almost immediately after the changes are made, significantly reducing the time it takes to find and fix defects.

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

Automated testing is one of the most important elements of DevOps. Modern automation tools enable unit testing, integration testing, functional testing, load testing, and security testing without human intervention. This significantly improves software quality and reduces the likelihood of critical errors after the release of a new product version.

Once testing is successfully completed, the continuous delivery phase begins. The completed software version is automatically transferred to the test or production environment. All deployment operations are performed according to predefined scripts, ensuring repeatability and eliminating most errors associated with manual server configuration.

Infrastructure management plays a vital role in the development lifecycle. In modern projects, infrastructure is described as software code containing server parameters, network connections, operating systems, and necessary software components. This approach allows for the rapid creation of identical computing environments, automated scaling, and significantly simplified information system maintenance.

Application containerization holds a special place in DevOps. Using containers ensures a consistent software runtime environment, regardless of the hardware platform or operating system. This significantly reduces the number of errors that occur when migrating an application between development, testing, and production environments.

After application deployment, the operation and monitoring phase begins. Modern monitoring systems continuously collect information on server status, computing resource usage, application response times, the number of user requests, and any errors. Analyzing this data allows for the prompt detection of software deviations and the prevention of serious failures.

Information obtained during operation is used to further improve the software product. Developers receive information about system performance, the most common errors, and user usage patterns. This allows them to promptly adjust architectural decisions, optimize the code, and improve the quality of future software versions.

In today's environment, significant attention is paid to information security issues. Automated verification tools enable source code analysis, detection of

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

known vulnerabilities in used libraries, monitoring of server configurations, and assessment of the security level of the information infrastructure. Integrating such mechanisms into the overall development process ensures continuous monitoring of software security throughout the entire lifecycle.

The use of DevOps technologies has a significant impact on key performance indicators in software development. It reduces the duration of individual lifecycle stages, reduces the number of errors during the implementation of new versions, improves the stability of information systems, and increases the speed of recovery after failures. At the same time, the quality of interaction between all project participants improves, as development, testing, operations, and maintenance processes become part of a single organizational system.

Thus, the use of DevOps technologies enables comprehensive automation of the software development lifecycle. The integration of development, testing, deployment, operations, and monitoring processes enables organizations to significantly improve the quality of software products, accelerate their time to market, and ensure the sustainable operation of digital services in the face of constantly changing user requirements and advances in information technology.

Table 1 - Impact of DevOps technologies on the stages of the software development life cycle

Life cycle stage	**The Result of DevOps Application**
Analysis and planning	Improving the efficiency of interaction between project participants and reducing the time required to agree on requirements
Software development	Automate version control and speed up collaborative development
Testing	Automated test execution and early error detection
Deployment	Fast and stable implementation of new software versions
Operation and maintenance	Continuous monitoring, prompt fault detection and rapid restoration of functionality

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

The presented data shows that DevOps technologies provide automation of all key stages of the software development life cycle, helping to reduce process execution time and improve the quality of the final software product.

Modern DevOps tools and technologies. Practical implementation of the DevOps concept is impossible without specialized software tools that automate all stages of the software development lifecycle. The modern DevOps ecosystem includes a wide range of tools designed for source code management, build automation, testing, deployment, monitoring, and maintenance of software products. The comprehensive use of such solutions enables a continuous development process in which most operations are performed automatically.

In DevOps, version control systems play a key role. They store source code centrally, control changes, and allow multiple developers to work simultaneously. Every change is recorded in the repository. This preserves development history, simplifies error detection, and allows for the restoration of previous versions. This makes team development much more efficient and reduces the risk of data loss [10].

The next important element is automated continuous integration and continuous delivery processes. Such platforms automatically receive changes to the software code, build it, run tests, analyze software quality, and, if all checks are successful, prepare the new version for deployment. This automation significantly reduces the time between making changes and releasing the final product.

Containerization plays a special role in modern DevOps practices. Container technologies allow software to be packaged along with all necessary libraries, settings, and dependencies into a single software container. This approach ensures consistent application performance regardless of the computing environment. Containerization significantly simplifies software migration between different servers and cloud platforms, and facilitates the scalability of information systems.

Specialized orchestration platforms are used to manage large numbers of containers. They automatically distribute the load between computing nodes, monitor service availability, scale applications, and ensure recovery from

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

failures. Automatic container management improves the reliability of the information infrastructure and significantly reduces the need for specialists to perform routine operations.

An integral part of DevOps is managing infrastructure as software code. All parameters of the computing environment are described using special configuration files containing information about servers, network settings, software, and the rules for interaction between system components. This approach allows for the rapid creation of identical development, testing, and production environments, and significantly simplifies the maintenance of the information infrastructure.

Automated testing tools play a key role. They enable a large number of checks to be performed without direct developer intervention. Unit, integration, functional, regression, and load testing can be performed automatically. Regular testing after each code change allows for the timely detection of defects and their prevention from entering production.

Monitoring and logging tools are essential. They continuously collect information on application performance, server status, CPU usage, RAM capacity, response time, and the number of errors occurring. Analyzing this data allows for early detection of performance degradation, prediction of potential failures, and intervention to address them before they become serious.

Cloud technologies are widely used in modern information systems. Cloud infrastructure allows organizations to quickly create new computing resources, automatically scale applications as workloads increase, and efficiently utilize hardware capacity without purchasing expensive equipment. Cloud platforms significantly expand the scope for DevOps implementation and enable the rapid adaptation of infrastructure to changing business requirements.

Another promising area is the application of artificial intelligence and machine learning technologies in DevOps processes. Modern intelligent systems can analyze event logs, detect anomalies, predict the likelihood of failures, automatically identify the causes of malfunctions, and generate recommendations for optimizing information systems. The use of such solutions further improves the efficiency of software operations.

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

Particular attention is paid to information security. Modern DevOps platforms include automated code analysis, dependency checking, server configuration monitoring, and detection of known vulnerabilities. Automated security processes significantly reduce the likelihood of critical errors and ensure software compliance with modern information security standards.

The integrated use of these tools creates a unified technological environment in which all stages of the software lifecycle are interconnected. Any change to the software code automatically initiates a process chain that includes build, testing, quality analysis, deployment, and subsequent monitoring. This approach ensures high development speed while maintaining the required level of reliability and security.

An analysis of the advantages, limitations, and development directions of DevOps technologies. The implementation of DevOps technologies has a significant impact on the efficiency of software development processes and the overall performance of organizations. The integration of development, testing, deployment, and operations processes significantly improves the quality of software products, reduces project implementation timelines, and ensures the stable operation of information systems. However, the transition to a new development organization model is accompanied by a number of organizational and technical challenges that require a comprehensive approach to address them.

One of the main advantages of DevOps is the reduction in development time and the release of new software versions. Automating build, testing, and deployment processes significantly reduces the time required for routine operations and accelerates the implementation of new functionality. As a result, organizations are able to respond more quickly to changing user requirements and quickly adapt software products to evolving market conditions.

A significant benefit is improved software quality. Automated testing allows for the detection of most errors early in the development lifecycle, when fixing them requires significantly less time and resources. Continuous code quality control contributes to the creation of more reliable and robust information systems.

Another positive result of DevOps implementation is the reduction in errors during software deployment. The use of automated scripts ensures that all operations are performed consistently, regardless of human error. This reduces

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

the likelihood of failures due to improper server configuration or disruptions in the sequence of operations.

A significant advantage is the increased efficiency of collaboration between project participants. Developers, testers, system administrators, and information security engineers work within a single process, use common tools, and are jointly responsible for the quality of the software product. This approach facilitates faster information exchange, reduces decision approval time, and increases team productivity.

Using DevOps technologies significantly improves the reliability of information systems. Continuous monitoring of application status, automatic fault detection, and rapid recovery ensure high availability of digital services. This is especially important for organizations whose activities involve the provision of electronic services around the clock.

Despite significant benefits, DevOps implementation comes with a number of challenges. One of the most common issues is the need to change the organizational culture of the enterprise. Many organizations have long relied on traditional management models, where development and operations departments operate independently. The transition to collaborative work requires changes to established management methods, the distribution of responsibilities, and the organization of internal communications.

Another challenge is the insufficient level of specialist qualifications. Effective DevOps implementation requires knowledge of programming, system administration, process automation, cloud technologies, information security, and infrastructure management. Training employees and enhancing their professional competence requires significant time and financial resources.

Certain challenges arise when integrating DevOps into organizations' existing information systems. In many cases, legacy software products are used that do not support modern automation methods and require significant modernization. Furthermore, the transition to new technologies may be accompanied by a temporary decrease in productivity due to the need to reconfigure the existing infrastructure.

Particular attention must be paid to information security issues. Automating deployment processes increases the speed of change propagation, but at the same

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

time increases the risk of errors or vulnerabilities spreading rapidly. For this reason, modern organizations are actively implementing DevSecOps principles, which involve integrating information security mechanisms into all stages of the software development lifecycle.

The future development of DevOps is closely linked to the advancement of artificial intelligence, machine learning, and intelligent automation. Modern software solutions are already capable of automatically analyzing event logs, predicting failures, identifying application anomalies, and generating performance optimization recommendations. The role of intelligent systems will continue to grow, significantly reducing human involvement in routine operations.

Another promising area is the expanding use of cloud computing and serverless technologies. Organizations are gradually moving toward fully managed infrastructure, in which most of the processes for creating, scaling, and maintaining computing resources are automated. This will further increase the flexibility of information systems and reduce their operating costs.

The rise of microservices architecture is significantly impacting the development of DevOps. The independent operation of individual components of an information system allows for their autonomous updating, testing, and scaling without disrupting the entire software system. This approach increases the resilience of software products and accelerates the implementation of new functionality.

Table 2 - Assessment of the impact of DevOps technologies implementation on key software development indicators

Indicator	**The Importance of DevOps Implementation**
New version release time	45 percent reduction
Number of deployment errors	60 percent reduction
Automatic testing frequency	Up to 100 launches per day
Mean time to recovery after failure	Reduction from 4 hours to 30 minutes
Development team productivity	A 35 percent increase

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

The figures provided are illustrative and demonstrate typical results achieved by organizations through the comprehensive implementation of DevOps technologies. Automating development and operations processes helps increase team productivity, reduce errors, and speed up time to market for software products.

Conclusion

The study demonstrated that DevOps technologies are an important development in modern software engineering and play a significant role in improving the software development lifecycle. Their use enables the integration of software development, testing, deployment, operation, and maintenance processes into a single automated system.

DevOps has been shown to reduce development timelines, improve software quality, reduce errors during new releases, and enhance the reliability of information systems. Automating key lifecycle processes allows organizations to more quickly adapt to changing user requirements and efficiently utilize existing computing resources.

The study determined that successful DevOps implementation requires not only the use of modern software tools but also the development of a new organizational culture based on collaboration, shared responsibility, and continuous improvement of development processes. Of particular importance is the integration of information security mechanisms to ensure the protection of software products at all stages of their lifecycle.

The prospects for the development of DevOps technologies are linked to the further spread of cloud computing, containerization, microservice architecture, intelligent automation, and artificial intelligence. The integrated application of these technologies will significantly improve the efficiency of software development, ensure the sustainable operation of digital services, and create the conditions for the further digital transformation of organizations across various industries.

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

References

1. Krutyakov D. V. PRACTICE OF IMPLEMENTING DEVOPS AND CI/CD IN TEAM SOFTWARE DEVELOPMENT // Science Bulletin. No. 2 (95). URL: https://cyberleninka.ru/article/n/praktika-vnedrenia-devops-i-ci-cd-v-komandnoy-razrabotke-programmnogo-obespecheniya
2. Lotov K. A., Andieva E. Yu. EVOLUTION OF DEVOPS: THE ROLE OF CI/CD IN ACCELERATION OF THE SOFTWARE DEVELOPMENT CYCLE // Bulletin of Science. No. 4 (85). URL: https://cyberleninka.ru/article/n/evolyutsiya-devops-rol-ci-cd-v-uskorenii-tsikla-razrabotki-programmnogo-obespecheniya
3. Tyumentsev D. V. DEVOPS IN THE ERA OF CLOUD TECHNOLOGIES: MODERN PRACTICES AND DEVELOPMENT PROSPECTS // Science Bulletin. 2023. No. 8 (65). URL: https://cyberleninka.ru/article/n/devops-v-epohu-oblacnyh-tehnologiy-sovremennye-praktiki-i-perspektivy-razvitiya
4. Bezpyaty M. V. AUTOMATION AND OPTIMIZATION OF DEVELOPMENT AND DEPLOYMENT PROCESSES IN DEVOPS: APPLICATION OF MODERN METHODS AND TOOLS // Innovations and Investments. 2023. No. 7. URL: https://cyberleninka.ru/article/n/avtomatizatsiya-i-optimizatsiya-processov-razrabotki-i-razvertyvaniya-v-devops-primenenie-sovremennyh-metodov-i-instrumentov
5. Bell S., Brant R. DevOps and machine learning: how to deploy a machine learning model. IT Pro, 2020. 22(3), pp. 44-48.
6. Kim J., Humble J., DeBois P., Willis J. The DevOps Handbook: How to Deliver World-Class Agility, Reliability, and Security in Technology Organizations.
Portland, OR: IT Revolution Press. 2016. pp. 34-67.
7. Bird S., Menzies T., Zimmerman T. The Art and Science of Software Data Analysis. 1st ed. San Francisco, CA: Morgan Kaufmann. 2015. pp. 201–225.
8. Forsgren, N., and Humble, J. (2018). Accelerate: The Science of Lean Software and DevOps: Building and Scaling High-Performance Technology Organizations. Portland, OR: IT Revolution Press. 2018. pp. 45–70.

Modern American Journal of Business, Economics, and Entrepreneurship
ISSN (E): 3067-7203
Volume 01, Issue 01, April, 2025
Website: usajournals.org

9. De Grandis D. Making Work Visible: Uncovering Time Theft to Optimize Work and Document Flow. Portland, OR: IT Revolution Press. 2017. pp. 87-112.
10. Smeds K., Nybom K. DevOps: A Software Architect's Perspective. 1st ed. Boston, MA: Addison-Wesley Professional. 2019. pp. 130–155.