UDC 004.78:004.432 DOI https://doi.org/10.35546/kntu2078-4481.2025.1.2.4

YE. A. BOROVSKOVA
Master's Degree, Software Engineer
AppsFlyer Ltd
ORCID: 0009-0008-5481-8090

INFLUENCE OF SERVERLESS INFRASTRUCTURE USAGE ON APP MAINTENANCE COSTS IN SEVER-SIDE APPLICATIONS

The relevance of the study is determined by the increasing demand for efficient IT infrastructure in the context of rapid digital technology development. Traditional models of server resource management reveal limitations in scalability, financial efficiency, and flexibility. Serverless infrastructure, based on a pay-as-you-go model, offers an innovative approach to cost optimization and performance improvement. However, challenges in its implementation require thorough analysis and clear recommendations.

The aim of the study is to evaluate the impact of serverless infrastructure on application maintenance costs, identify its advantages and risks, and develop a methodology for assessing its economic feasibility in both short and long-term perspectives. To achieve this, methods of systems analysis, comparative analysis, forecasting, and cost modeling were applied. As a result, the study compared the features of serverless and traditional models and provided practical recommendations for optimizing serverless infrastructure usage.

The research demonstrates that serverless infrastructure significantly reduces costs in low-traffic scenarios, simplifies resource management, and decreases the need for highly skilled DevOps professionals. At the same time, hidden costs, integration challenges, data security risks, and vendor dependency were identified as key issues. The proposed methodology for assessing the economic feasibility of transitioning to serverless architecture enables detailed analysis of costs and benefits, accounting for project-specific features and facilitating informed decision-making.

The findings emphasize that serverless architecture is a promising tool for optimizing digital systems, yet its implementation requires careful planning and risk assessment. Future research should focus on the long-term impact of serverless technologies on IT infrastructure efficiency, the development of adaptive models for high-performance systems, and innovative solutions for ensuring data security in cloud environments. This will contribute to improving infrastructure management and enhancing the efficiency of digital systems in the future.

Key words: *serverless infrastructure, cost optimization, resource management, economic feasibility, cloud services, implementation risks.*

Є. А. БОРОВСКОВА
магістр, інженер програмного забезпечення
AppsFlyer Ltd
ORCID: 0009-0008-5481-8090

ВПЛИВ ВИКОРИСТАННЯ БЕЗСЕРВЕРНОЇ ІНФРАСТРУКТУРИ НА ВИТРАТИ НА ПІДТРИМКУ ЗАСТОСУНКІВ У СЕРВЕРНІЙ ЧАСТИНІ

Актуальність дослідження зумовлена зростаючими вимогами до ефективності IT-інфраструктури в умовах динамічного розвитку цифрових технологій. Традиційні моделі управління серверними ресурсами демонструють обмеження щодо масштабованості, фінансової ефективності та гнучкості. Увагу привертає безсерверна інфраструктура, яка базується на моделі оплати за фактичне використання ресурсів і пропонує інноваційний підхід до оптимізації витрат та підвищення продуктивності. Водночас спостерігаються виклики у процесі впровадження, які потребують ґрунтовного аналізу та чітких рекомендацій.

Метою дослідження є оцінка впливу безсерверної інфраструктури на витрати на підтримку серверних застосунків, визначення переваг і ризиків, а також створення методики оцінки економічної доцільності її впровадження у коротко- та довгостроковій перспективі. Для виконання поставлених завдань застосовано методи системного аналізу, компаративного аналізу, прогнозування та моделювання витрат. У результаті виконано порівняння особливостей безсерверної моделі та традиційних рішень, а також розроблено практичні рекомендації для оптимізації її використання.

Дослідження демонструє, що безсерверна інфраструктура сприяє суттєвому зниженню витрат у сценаріях із низьким рівнем трафіку, спрощенню управління ресурсами та зменшенню потреби у висококваліфікованих фахівцях DevOps. Водночас ідентифікуються приховані витрати, складнощі інтеграції з існуючими системами, ризики для безпеки даних та залежність від постачальників хмарних послуг. Запропонована методика оцінки економічної доцільності переходу на безсерверну модель забезпечує можливість аналізу витрат і вигод з урахуванням специфіки проєкту, що дозволяє приймати обґрунтовані рішення.

Робота акцентує увагу на тому, що безсерверна архітектура є перспективним інструментом для оптимізації цифрових систем, проте її впровадження потребує ретельного планування та врахування потенційних ризиків. Подальші дослідження передбачають детальне вивчення довгострокового впливу безсерверних технологій на IT-інфраструктуру, розробку адаптивних моделей для високонавантажених систем, а також створення інноваційних рішень для забезпечення безпеки даних у хмарному середовищі. Це сприятиме вдосконаленню управління інфраструктурою та підвищенню ефективності цифрових систем у майбутньому.

Ключові слова: *безсерверна інфраструктура, оптимізація витрат, управління ресурсами, економічна доцільність, хмарні сервіси, ризики впровадження.*

Problem statement

In the current conditions of information technology development, there is a growing need to optimise the cost of supporting server applications, which are key components of many digital systems. Given the ever-increasing complexity of the infrastructure and the need to ensure its high performance, uninterrupted operation, and scalability, traditional approaches to managing server resources require significant improvement. Using a serverless infrastructure that automates the deployment, management, and scaling of computing resources is becoming one of the most promising solutions to this problem.

The serverless model is characterised by developers not needing to worry about maintaining servers, and resources are automatically allocated depending on the application's needs. This reduces technical support costs, improves computing power efficiency, and simplifies development processes. However, the implementation of such a model is also accompanied by several challenges related to analysing the feasibility of its application in specific cases, particularly in the long term, where there may be hidden costs associated with integration, data security and adaptation of existing software.

The problem's relevance is determined by the need to develop clear methodologies for assessing the cost-effectiveness of the transition to a serverless infrastructure, considering the specifics of server-based applications. The scientific task is to create conceptual models for analysing and forecasting costs, increasing digital system's conomic efficiency. The study's practical significance is to develop recommendations for optimising software support costs in the context of using innovative serverless infrastructure technologies.

Analysis of the latest research and publications

Using serverless infrastructure has become an important tool for optimising server support costs. This topic covers a number of technical, economic, and legal aspects, which are reflected in the works of various authors. O. Kostenko and V. Furashev study the legal aspects of introducing innovative technologies, such as serverless solutions, in the web space. They argue that such architectures contribute to the creation of adaptive legal platforms that can function effectively in the dynamic conditions of the meta-universe, optimising resources and reducing support costs [1]. In the field of information security, special attention is paid to automating data protection processes. K. Chyzhmar and his researchers demonstrate how serverless solutions provide resilience to cyber threats through automated security management. In addition, the introduction of such technologies can significantly reduce costs without compromising system functionality [2].

D. Zanon's research reveals the economic benefits of serverless architectures. His analysis focuses on eliminating the need to administer physical servers, which is especially relevant for small and medium-sized enterprises. The study's practical examples confirm the effectiveness of these systems in optimising operating costs [3].

J. Scheuner conducted a comparative analysis of various serverless platforms. The paper highlights how resource optimisation ensures stable performance even in high-load scenarios. The researcher also emphasises the importance of adapting systems to the specifics of the tasks [4].

B. Zambrano considers architectural solutions aimed at increasing the scalability of serverless systems. His recommendations are focused on corporate environments, where the introduction of such technologies can significantly reduce the cost of technical support and administration [5].

The peculiarities of process automation in serverless infrastructure are highlighted in the works of M. Roberts and J. Chapin. The authors emphasise that automated resource management helps to increase system reliability and reduce costs by minimising the impact of the human factor [6].

The role of serverless solutions in scientific computing is analysed by J. Añel and colleagues. The researchers' conclusions show that automation of resource management can reduce infrastructure costs and make complex computing tasks more accessible [7].

The flexibility in scaling systems provided by serverless solutions is analysed by C. Safer. His research focuses on the effectiveness of using cloud platforms for dynamic projects with heavy loads. According to the author, such approaches can significantly reduce operating costs [8].

An analysis of the impact of autonomous services and microfrontends on resource savings was conducted by J. Gilbert and E. Price. Their research demonstrates how increasing the modularity of systems contributes to the economic feasibility of implementing these technologies [9].

The educational aspect of using serverless systems is the subject of research by J. Katzer. He emphasises that training specialists in this area is a key element for the successful implementation of technologies and increasing the overall performance of systems [10].

DevOps practices for serverless systems are studied by S. Bangera. In his work, he emphasises that coordinated work between development and system administrator teams helps to optimise deployment processes and reduce infrastructure maintenance costs [11].

The research presented here demonstrates the significant potential of serverless architecture to reduce the cost of supporting systems on the server side. Process automation, scalability, and resource savings allow this technology to be used in both business and research projects while maintaining a balance between efficiency and economic feasibility. Despite advances in serverless infrastructure research, important aspects remain unresolved, which limits its effective implementation. In particular, the conceptual features of this model compared to traditional approaches are not well understood, which makes it difficult to adapt it to a wide range of applications. The cost-effectiveness of serverless technologies is mostly assessed for individual scenarios, but there is a lack of systematic approaches to cost-benefit analysis in different conditions. Integration of serverless infrastructure with existing systems remains a challenge due to the lack of models for assessing risks and hidden costs. Data security aspects in the cloud environment require additional research, especially given the increased requirements for data protection. There is also a lack of practical guidance for adapting this architecture to different types of applications, which limits its versatility. The proposed research is aimed at developing a methodology for a comprehensive assessment of economic feasibility and practical recommendations that take into account the specifics of applications and long-term prospects. This is expected to fill existing gaps, broaden scientific understanding of the problem, and facilitate more efficient implementation of serverless infrastructure in various industries.

Formulation of the research objective

The article's purpose is to analyse the impact of using serverless infrastructure on the costs of supporting server applications, determine its effectiveness in the context of maintenance optimisation, and develop recommendations for its implementation in modern digital systems.

Research objectives:

1. To investigate the conceptual features of serverless infrastructure, its advantages and differences from traditional server resource management models, and to assess the economic efficiency of its use based on the analysis of server application support costs.
2. Identify the key challenges and risks of implementing serverless infrastructure, including hidden costs, aspects of integration with existing systems, and data security issues in the cloud environment.
3. To develop practical recommendations for optimising the use of serverless infrastructure, considering the needs of different types of applications, and to propose a methodology for assessing the economic feasibility of switching to a serverless model in the short and long term.

Summary of the main material

Serverless infrastructure has become an innovative approach to managing server resources that radically changes traditional approaches to hosting and supporting applications. It is based on a model where computing resources are provided on demand, and developers focus exclusively on software functionality, leaving infrastructure management to the service provider [4]. Traditional models involve the continuous operation of servers with predefined characteristics, which requires significant financial and technical resources to ensure their performance and scalability. In this context, serverless infrastructure provides automation of resource management, cost reduction, and performance improvement (Table 1).

Table 1

Comparison of traditional hosting and serverless infrastructure in server-based relationships

Comparison parameter	Traditional hosting	Serverless infrastructure
Basic principle of operation	Constant operation of servers with fixed characteristics	Use resources only when needed
Costs	Fixed costs, regardless of the level of usage	Payment for actual use
Complexity of management	Requires the involvement of DevOps specialists	Can be maintained by a small team or even a single developer
Scalability	Manual configuration or use of predefined parameters	Automatic scalability
Examples of technologies	EC2-based servers, physical servers	AWS/Lambda, Google Cloud/Cloud Functions, Microsoft Azure/Azure Functions
Cost-effectiveness	Depends on the level of resource utilisation	Significantly reduced for low traffic applications

Source: compiled by the author based on [4, 5, 7, 9].

The practical implementation of serverless infrastructure demonstrates significant advantages in today's environment. For example, AWS Lambda allows server functions to run only on demand, which is especially beneficial for low-traffic applications. At the same time, costs can be reduced to a few dollars per month, compared to tens or even hundreds of dollars for using traditional servers that run continuously. Using NestJS in combination with AWS Lambda and API Gateway ensures compatibility with microservice architecture, modularity, and efficient dependency injection, which further simplifies development and maintenance [6]. Reducing the complexity of the infrastructure also significantly reduces the need to engage DevOps specialists as support tasks become less technically complex [10]. This approach provides the flexibility and scalability required for modern digital systems.

The cost-effectiveness of serverless technology is one of the key aspects that makes this model attractive for modern server-based applications. Thanks to the pay-as-you-go principle, serverless infrastructure can significantly reduce costs, especially for low-traffic applications. Unlike traditional hosting, where servers are running 24/7, the serverless approach involves allocating resources only when tasks are running. This provides savings, reduces operational complexity, and allows developers to focus on the functional aspects of applications (Table 2). The figures in Table 2 are based on official AWS tariffs available in public documentation and represent a calculation for a typical usage scenario – an application with 50 requests per day. This scenario simulates real-world conditions for small or test projects that are often encountered in practice.

Table 2

Cost-effectiveness of traditional hosting and serverless infrastructure in server applications

Metric	Traditional server (EC2)	Serverless model (AWS Lambda)
Type of resource	EC2 t3. medium (2 vCPU, 4 GB RAM)	AWS Lambda (on-demand functions)
Cost per usage unit	$0.0416 (per hour)’	$0.00000001564 (per request assuming that its execution time = 5 seconds)’
Monthly costs (50 requests/day)	$29.95 (running 24/7)’	$0.0073 (assuming 5 seconds execution time per request)’
Annual costs	$359.40 (running 24/7)	$0.003 (assuming 5 seconds execution time per request)
Involvement of DevOps resources	Requires highly qualified specialists	Can be supported by a single developer

Source: compiled by the author on the basis of [12–16].

The practical implementation of serverless technologies demonstrates significant savings for low-traffic applications. For example, in a scenario with 50 requests per day – 5 seconds duration time per each one, the annual cost of a traditional t3.medium server is approximately $359, while using AWS Lambda costs only $0.003. This analysis confirms that a serverless infrastructure is extremely efficient when servers are used infrequently and provides flexibility in scaling. Combined with services such as DynamoDB, S3, and NestJS, it can reduce operating costs and ensure easy integration even for small development teams.

Implementing a serverless infrastructure is a promising way to optimise costs and increase the efficiency of server resource management. However, it is accompanied by several challenges and risks that require detailed analysis. One of the main risks is hidden costs that may arise from uneven or unpredictable resource usage [8]. Although the pay-as-you-go model seems cost-effective, it can become financially burdensome for applications with high call frequency or significant data processing, which increases the overall cost of service.

Integrating serverless infrastructure with existing systems is also a challenge. Implementing solutions such as AWS Lambda or API Gateway requires adapting the application architecture, which can require additional development, testing, and configuration costs [3]. In the case of complex systems with many dependencies, integrating a serverless infrastructure can lead to unexpected technical problems, such as performance degradation due to network delays or coordination issues between different components.

Another challenge is ensuring data security, as serverless infrastructure involves storing and processing information in a cloud environment. This increases the risks associated with data confidentiality and availability, which requires additional security measures [11]. This is compounded by dependence on the service provider, which can create limitations in the choice of solutions and affect the long-term strategy of IT infrastructure development.

Another important aspect is the need for developers with appropriate qualifications to work with serverless technologies. Despite simplifying some operational tasks, developing and maintaining serverless applications requires a deep understanding of new technologies and their specifics. Failure to consider these challenges at the implementation stage can lead to significant delays in project implementation and increased costs for team training [4].

Thus, while serverless infrastructure offers significant advantages in flexibility and cost-effectiveness, successful implementation requires careful planning, a sound risk assessment, and the development of mitigation strategies. This will ensure stable application performance and optimise overall infrastructure support costs.

Optimising the use of serverless infrastructure requires a comprehensive approach that considers the specifics of applications and their functional requirements. For applications with low traffic or irregular workloads, it is recommended to use a pay-as-you-go model to minimise costs. In such cases, using AWS Lambda with the API Gateway for managing requests provides cost-effectiveness and scalability [5]. It is important to design the application architecture in such a way as to avoid unnecessary function calls that can lead to increased costs.

For systems with high-performance requirements, you should pay attention to the possibility of optimising delays that may occur due to the peculiarities of processing requests in the cloud environment. This can be achieved through data caching and integration with services such as DynamoDB, which provide high-speed access to information. For applications with a large number of interdependent components, it is advisable to consider using a microservice architecture that simplifies integration with serverless solutions and reduces the complexity of infrastructure management.

In the process of developing and maintaining serverless applications, attention should be paid to automating resource monitoring and management. Using tools such as CloudWatch allows you to receive data on the performance of functions and respond quickly to failures or increased load. Additional security measures, including encryption and access control,

should be implemented to ensure data security, as the use of the cloud environment involves an increased risk of information leakage.

Depending on the needs of the application, it is also important to consider the integration of serverless solutions with existing systems. When moving from a traditional infrastructure, it is necessary to audit the architecture and identify components that can be migrated to a serverless model without significant adaptation costs. At the same time, the risks of dependence on a cloud service provider should be taken into account, and the ability to migrate to other platforms should be ensured if necessary.

The practical implementation of such approaches will optimise costs, ensure application stability, and increase resource management flexibility. This will form the basis for creating effective digital solutions that meet modern business and technology requirements.

Evaluating the economic feasibility of switching to a serverless model is a key stage in planning the implementation of modern infrastructure solutions in server applications. In today's environment, when resource efficiency and cost optimisation are becoming a priority for most organisations, this methodology allows you to make informed decisions about the feasibility and scope of serverless architecture integration. It helps to avoid unnecessary costs and risks associated with insufficient consideration of project specifics or errors in forecasting resource requirements.

Unlike traditional estimation models that focus on analysing the capital and operating costs of supporting servers, the methodology for the serverless model takes into account the dynamics of the load, the specifics of using functional components, and integration with cloud services. It is based on analysing key indicators, such as traffic volume, number of requests, amount of stored data, performance level, and duration of use of functions. This makes the approach adaptable to the conditions of a particular project and allows us to consider both short-term and long-term benefits (Fig. 1).

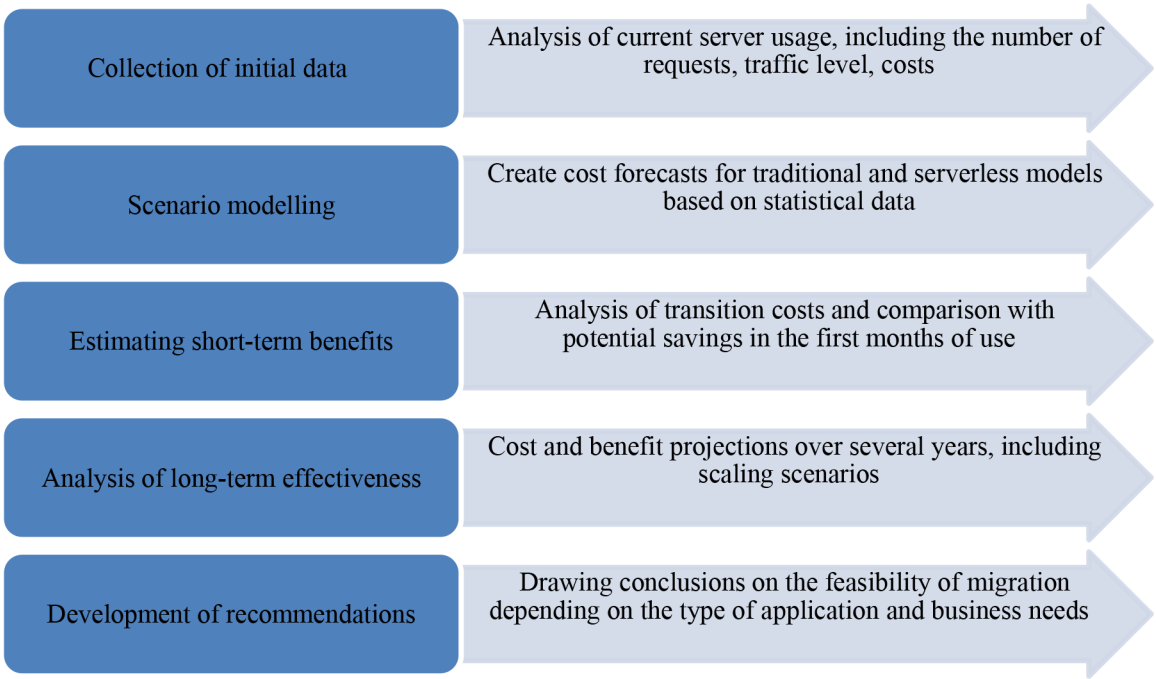

Fig. 1. Stages of the methodology for assessing the economic feasibility of switching to a serverless model

Source: author's own development

The methodology allows you to assess whether it is reasonable to switch to a serverless model for a particular application under different load scenarios. It will help to predict which costs will be most significant in the initial stages of integration and which will be more significant in the long term, taking into account the possible increase in the volume of requests or changes in business requirements. In addition, thanks to detailed modelling, companies are expected to be able to better adapt their resource management strategy, reducing the risks of mismanagement and increasing the efficiency of their IT solutions.

This will be the basis for ensuring application stability, reducing financial burden and maintaining competitive advantage.

Conclusions

Serverless infrastructure opens up new opportunities for optimising the cost of supporting server applications, increasing productivity, and simplifying resource management. The analysis showed that the main advantages of this model are flexibility in scaling, a significant reduction in financial costs in low-traffic scenarios, and a reduction in the need to attract highly qualified DevOps specialists.

It is found that the implementation of serverless infrastructure is accompanied by key problems such as hidden costs, difficulties in integrating with existing systems, increased data security risks, and dependence on cloud service providers. Practical recommendations for optimising this model to meet the needs of different types of applications are developed, including selecting appropriate tools, resource monitoring and automation of infrastructure management. A methodology for assessing the economic feasibility of switching to a serverless model based on the analysis of costs and benefits in the short and long term is proposed.

The results of the study are useful for making informed decisions on the implementation of serverless architecture in modern digital systems. Prospects for further research include a detailed study of the long-term impact of serverless technologies on the cost-effectiveness of IT infrastructure, the development of models for adapting these solutions for systems with high-performance requirements, and the exploration of new ways to ensure data security in the cloud environment. This will help improve existing methods and increase the efficiency of digital systems in the future.

Bibliography

1. Kostenko O., Furashev V. Genesis of Legal Regulation Web and the Model of the Electronic Jurisdiction of the Metaverse. *Bratislava Law Review*. 2022. Vol. 6, No. 2. P. 21–36. DOI: https://doi.org/10.46282/blr.2022.6.2.316 (date of access: 13.12.2024).

2. Chyzhmar K., Dniprov O., Korotiuk O., Shapoval R., Sydorenko O. State Information Security as a Challenge of Information and Computer Technology Development. *Journal of Security and Sustainability Issues*. 2020. Vol. 9(3). P. 819–828 URL: https://www.researchgate.net/publication/340545387_STATE_INFORMATION_SECURITY_AS_A_CHALLENGE_OF_INFORMATION_AND_COMPUTER_TECHNOLOGY_DEVELOPMENT (date of access: 11.12.2024).

3. Zanon D. Building Serverless Web Applications. *Packt Publishing Ltd*. 2017. URL: https://books.google.com.ua/books?hl=uk&lr=&id=5eZDDwAAQBAJ&oi=fnd&pg=PP3&dq=INFLUENCE+OF+SERVERLESS+INFRASTRUCTURE+USAGE+ON+APP+MAINTENANCE+COSTS+IN+SERVER-SIDE+APPLICATIONS+books&ots=XMqMJEjD6m&sig=FesWJ9UC2Ic2D2d-Yh6vaG1CIik&redir_esc=y#v=onepage&q&f=false (date of access: 13.12.2024).

4. Scheuner J. *Performance Evaluation of Serverless Applications and Infrastructures*. Chalmers Tekniska Högskola (Sweden). 2022. URL: https://www.proquest.com/openview/a8df44b8f2f29937b4aac8241292364b/1?pq-origsite=gscholar&cbl=2026366&diss=y (date of access: 13.12.2024).

5. Zambrano B. *Serverless Design Patterns and Best Practices: Build, Secure, and Deploy Enterprise-Ready Serverless Applications with AWS to Improve Developer Productivity*. Packt Publishing Ltd. 2018. URL: https://books.google.com.ua/books?hl=uk&lr=&id=lppVDwAAQBAJ&oi=fnd&pg=PP1&dq=INFLUENCE+OF+SERVERLESS+INFRASTRUCTURE+USAGE+ON+APP+MAINTENANCE+COSTS+IN+SERVER-SIDE+APPLICATIONS+books&ots=I_MMvvweHA&sig=4IKsyQEmxqNtTzF4sCaGa-41fbA&redir_esc=y#v=onepage&q&f=false (date of access: 13.12.2024).

6. Roberts M., Chapin J. *What is Serverless? O'Reilly Media, Incorporated*. 2017. URL: https://symphonia.io/what-is-serverless.pdf (date of access: 13.12.2024).

7. Cloud and Serverless Computing for Scientists. J.A. Añel, et al. Springer International Publishing. 2020. URL: https://link.springer.com/book/10.1007/978-3-030-41784-0 (date of access: 13.12.2024).

8. Safeer C.M. *Architecting Cloud-Native Serverless Solutions: Design, Build, and Operate Serverless Solutions on Cloud and Open Source Platforms*. Packt Publishing Ltd. 2023. URL: https://books.google.com.ua/books?hl=uk&lr=&id=4fLBEAAAQBAJ&oi=fnd&pg=PP1&dq=INFLUENCE+OF+SERVERLESS+INFRASTRUCTURE+USAGE+ON+APP+MAINTENANCE+COSTS+IN+SERVER-SIDE+APPLICATIONS+books&ots=kDEya_1XLf&sig=8w2uOkB_pjMaaZIrkmWnZFRSd6c&redir_esc=y#v=onepage&q&f=false (date of access: 13.12.2024).

9. Gilbert J., Price E. *Software Architecture Patterns for Serverless Systems: Architecting for Innovation with Events, Autonomous Services, and Micro Frontends*. Packt Publishing Ltd. 2021. URL: https://books.google.com.ua/books?hl=uk&lr=&id=gUs2EAAAQBAJ&oi=fnd&pg=PP1&dq=INFLUENCE+OF+SERVERLESS+INFRASTRUCTURE+USAGE+ON+APP+MAINTENANCE+COSTS+IN+SERVER-SIDE+APPLICATIONS+books&ots=5mmeBL0fq&sig=j9XaXBE2oQpkz4g6ky_DsN77VJU&redir_esc=y#v=onepage&q&f=false (date of access: 13.12.2024).

10. Katzer J. *Learning Serverless. O'Reilly Media, Inc*. 2020. URL: https://books.google.com.ua/books?hl=uk&lr=&id=1f8FEAAAQBAJ&oi=fnd&pg=PT9&dq=INFLUENCE+OF+SERVERLESS+INFRASTRUCTURE+USAGE+ON+APP+MAINTENANCE+COSTS+IN+SERVER-SIDE+APPLICATIONS+books&ots=BFL9FJYjkj&sig=y-wf-cckO70Kz0MdWtBym8XBL7M&redir_esc=y#v=onepage&q&f=false (date of access: 13.12.2024).

11. Bangera S. *DevOps for Serverless Applications: Design, Deploy, and Monitor Your Serverless Applications Using DevOps Practices.* Packt Publishing Ltd. 2018. URL: https://books.google.com.ua/books?hl=uk&lr=&id=5PxwDwAAQBAJ&oi=fnd&pg=PP1&dq=INFLUENCE+OF+SERVERLESS+INFRASTRUCTURE+USAGE+ON+APP+MAINTENANCE+COSTS+IN+SERVER-SIDE+APPLICATIONS+books&ots=PQc1kYapOY&sig=m4Qi3rT0ifNfqRM2zPDm8VE4hj0&redir_esc=y#v=onepage&q&f=false (date of access: 13.12.2024).

12. Amazon Web Services. *AWS Lambda Pricing*. 2024. URL: https://aws.amazon.com/lambda/pricing/ (date of access: 13.12.2024).

13. Amazon Web Services. *Amazon EC2 Pricing*. 2024. URL: https://aws.amazon.com/ec2/pricing/ (date of access: 13.12.2024).

14. Amazon Web Services. *Amazon API Gateway*. 2024. URL: https://aws.amazon.com/api-gateway/ (date of access: 13.12.2024).

15. Amazon Web Services. *Amazon DynamoDB*. 2024. URL: https://aws.amazon.com/dynamodb/ (date of access: 13.12.2024).

16. Amazon Web Services. Amazon S3. 2024. URL: https://aws.amazon.com/s3/ (date of access: 13.12.2024).

References

1. Kostenko, O., & Furashev, V. (2022). Genesis of legal regulation web and the model of the electronic jurisdiction of the metaverse. *Bratislava Law Review*, 6 (2), 21–36. DOI: https://doi.org/10.46282/blr.2022.6.2.316

2. Chyzhmar, K., Dniprov, O., Korotiuk, O., Shapoval, R., & Sydorenko, O. (2020). State information security as a challenge of information and computer technology development. *Journal of Security and Sustainability Issues, 9*(3), 819–828. https://www.researchgate.net/publication/340545387_STATE_INFORMATION_SECURITY_AS_A_CHALLENGE_OF_INFORMATION_AND_COMPUTER_TECHNOLOGY_DEVELOPMENT

3. Zanon,D.(2017).*Buildingserverlesswebapplications*.PacktPublishingLtd.Retrivedfromhttps://books.google.com.ua/books?hl=uk&lr=&id=5eZDDwAAQBAJ&oi=fnd&pg=PP3

4. Scheuner, J. (2022). *Performance evaluation of serverless applications and infrastructures*. Chalmers Tekniska Högskola (Sweden). Retrived from https://www.proquest.com/openview/a8df44b8f2f29937b4aac8241292364b/1

5. Zambrano, B. (2018). *Serverless design patterns and best practices: Build, secure, and deploy enterprise-ready serverless applications with AWS to improve developer productivity*. Packt Publishing Ltd. Retrived from https://books.google.com.ua/books?hl=uk&lr=&id=lppVDwAAQBAJ&oi=fnd&pg=PP1

6. Roberts, M., & Chapin, J. (2017). *What is serverless?* O'Reilly Media, Incorporated. Retrived from https://symphonia.io/what-is-serverless.pdf

7. Añel, J.A., et al. (2020). *Cloud and serverless computing for scientists*. Springer International Publishing. https://link.springer.com/book/10.1007/978-3-030-41784-0

8. Safeer, C.M. (2023). *Architecting cloud-native serverless solutions: Design, build, and operate serverless solutions on cloud and open-source platforms*. Packt Publishing Ltd. Retrived from https://books.google.com.ua/books?hl=uk&lr=&id=4fLBEAAAQBAJ&oi=fnd&pg=PP1

9. Gilbert, J., & Price, E. (2021). *Software architecture patterns for serverless systems: Architecting for innovation with events, autonomous services, and micro frontends.* Packt Publishing Ltd. Retrived from https://books.google.com.ua/books?hl=uk&lr=&id=gUs2EAAAQBAJ&oi=fnd&pg=PP1

10. Katzer, J. (2020). *Learning serverless*. O'Reilly Media, Inc. Retrived from https://books.google.com.ua/books?hl=uk&lr=&id=1f8FEAAAQBAJ&oi=fnd&pg=PT9

11. Bangera, S. (2018). *DevOps for serverless applications: Design, deploy, and monitor your serverless applications using DevOps practices*. Packt Publishing Ltd. Retrived from https://books.google.com.ua/books?hl=uk&lr=&id=5PxwDwAAQBAJ&oi=fnd&pg=PP1

12. Amazon Web Services. (2024). AWS Lambda Pricing. https://aws.amazon.com/lambda/pricing/ (date of access: 13.12.2024).

13. Amazon Web Services. (2024). Amazon EC2 Pricing. https://aws.amazon.com/ec2/pricing/ (date of access: 13.12.2024).

14. Amazon Web Services. (2024). Amazon API Gateway. https://aws.amazon.com/api-gateway/ (date of access: 13.12.2024).

15. Amazon Web Services. (2024). Amazon DynamoDB. https://aws.amazon.com/dynamodb/ (date of access: 13.12.2024).

16. Amazon Web Services. (2024). Amazon S3. https://aws.amazon.com/s3/ (date of access: 13.12.2024).