ADVANCES IN CYBER-PHYSICAL SYSTEMS

Vol. 7, Num. 2, 2022

COMPARATIVE ANALYSIS OF SERVER AND SERVERLESS CLOUD COMPUTING PLATFORMS

Vladyslav Kotyk, Yevhenii Vavruk

Lviv Polytechnic National University, 12, Bandera Str, Lviv, 79013, Ukraine.
Authors' e-mail*: vladyslav.kotyk.mkisp.2021@lpnu.ua*

https://doi.org/10.23939/acps2022.__.___

Submitted on 05.10.2022

***Abstract* – Cloud computing is emerging as a powerful computing paradigm for the efficient use of resources. However, decisions to move to cloud computing always remain risky from the customer's point of view, considering the benefits they get from it.**

Existing research on cloud computing is more focused on technical aspects such as security, quality, efficiency, etc. However, research on the implementation of cloud computing is at an early stage. Thus, in this article, an attempt is made to create a model for cost analysis and advantages for deciding on the application of cloud computing. It takes into account various organizational parameters, designing server and serverless architectures using Microsoft Azure Portal cloud platform services and policies of this organization.

Also, it makes a comparative characterization of these services according to power and price criteria. A comparative description of these services according to capacity and price criteria is also given. It shows the structure of the test tool for assessment. Evaluation parameters and metrics are defined.

In addition, this article contains information about approaches to evaluating cloud platforms according to various criteria that are most important for a developer.

***Index Terms*: Cloud technologies, Serverless, Azure Portal, Cloud computing, Function App.**

I. INTRODUCTION

The emerging cloud computing is a model for enabling ubiquitous, convenient, and on-demand access to a shared pool of customizable computing resources (e.g. servers, storage, networks, applications, and services) [1] that can be provisioned with minimal service provider interactions and management efforts.

With the development of technology, cloud computing is gaining more and more popularity. They enable users to use the software from anywhere in the world and with the required characteristics [2].

The development of software architecture requires a detailed analysis of tools, components, and technologies for its implementation. Usually, a significant part of the allocated resources is not used in full and has relatively low efficiency. To optimize the allocation of computing resources, one should be able to effectively determine the complexity of the system and the required resources [3]. For this, the load and system downtime should be taken into account. The use of serverless computing makes it possible to solve some of the questions and achieve a more optimal solution for the selection and use of computer resources [4].

The concept of cloud computing is based on the idea that computing resources will be located somewhere other than the computer room, and that users will connect to them using the resources when needed. This pushes the infrastructure into the network so that the overall cost of managing hardware/software resources is reduced. There is insufficient research on a model that can demonstrate the benefits of cloud computing [5] adoption and suggest the ideal time to move to cloud computing. This study attempts to develop a cost-benefit analysis model that can present a clear picture to IT managers when it comes to migrating from legacy systems to cloud computing.

II. FORMULATION OF THE PROBLEMS

Cloud computing is an easily adaptable and efficient infrastructure for running applications and web applications. In connection with the constant growth of the capacities of electronic computing machines, the complexity of their implementation also increases. Local machines have already lost their adaptability and require more and more frequent modernization of hardware and software parts. The architecture of programs becomes more complex over time and loses flexibility. To solve this problem, there are cloud platforms that allow to optimize the used resources, ensure scalability and flexibility, and most importantly, effectively allocate costs to the speed of the system [3].

Even though cloud platforms provide elasticity and scalability of the software infrastructure, they still require detailed study to optimize production processes, reduce the cost of the system, and use computing power effectively. The correct choice of the necessary architecture for this system and the choice of the platform for implementation play a significant role in this. However, even the use of cloud platform services does not always ensure efficiency and achievement of the necessary goal of the software tool. Among cloud computing services, you can use such concepts as server and serverless architectures, which play a big role in the efficiency of using computing resources. Among the available studies, there is no clear picture of the use of serverless computing compared to server computing.

Performance and scalability testing and measurement of cloud software services are essential for the future optimization and growth of cloud computing [3].

III. PURPOSE AND FORMULATION OF THE TASK

Determining the objectives of the task is one of the prerequisites for ensuring an effective result for this assessment. This article focuses on the use of server and serverless computing using the Microsoft Azure Portal cloud platform.

Saving money and productivity are some of the main features of the software. The more productive the system, the more it costs, and vice versa. To solve this problem, it is necessary to determine the balance between the cost of the system and its performance.

This article aims to evaluate the architecture using a server-based or serverless system and compare its results. The main attention should be focused on the following characteristics - increasing system productivity due to shorter waiting time and response time, and reducing the cost of used resources.

For this, it is necessary to determine the main types of architectures for server and serverless computing, analyze the features and capabilities of these architectures, and provide basic structural diagrams. To evaluate the characteristics of the computing power, a software model of the application for the implementation of the task should be provided. An important factor is also the definition of approaches and methods for evaluating the software tool and conducting their analysis. Evaluating server and serverless architecture requires defining the main metrics and comparing the prices of cloud services.

IV. SYSTEM COMPONENTS

A. CHOICE OF SOFTWARE ARCHITECTURE

The architecture of the software tool plays a significant role in the implementation of the project infrastructure. It depends on many factors that determine whether the developed software will effectively use the provided resources and provide optimal price options for solving its own goals.

Determining the most effective characteristics includes parameters such as latency, availability, system speed, response time, CPU and GPU power, memory usage, etc.

The main purpose of this article is to determine the features of server and serverless architecture and to conduct a comparative analysis between these systems. The implementation is based on the use of the services of the Microsoft Azure Portal cloud platform [1].

Azure Portal is a Microsoft product released in 2010. This platform offers a wide selection of various support tools, programming languages, and frameworks. It runs on Microsoft Windows and Linux operating systems. Currently, the platform has about 60 services and data centers in more than 38 locations around the world.

To implement the task, we will determine the server and serverless architecture of the application.

Server architecture, another name the client-server architecture, — is a computing model in which the server hosts, delivers, and manages most of the resources and services requested by the client. It is also known as the networking computing model or client-server network as all requests and services are delivered over a network. The client-server architecture or model has other systems connected over a network where resources are shared among the different computers.

To implement our server architecture, we will use a virtual machine, named Azure Virtual Machine, that provides the necessary computing power, dynamically expandable memory, and can easily manage. The structural model of this architecture is shown in Fig. 1.

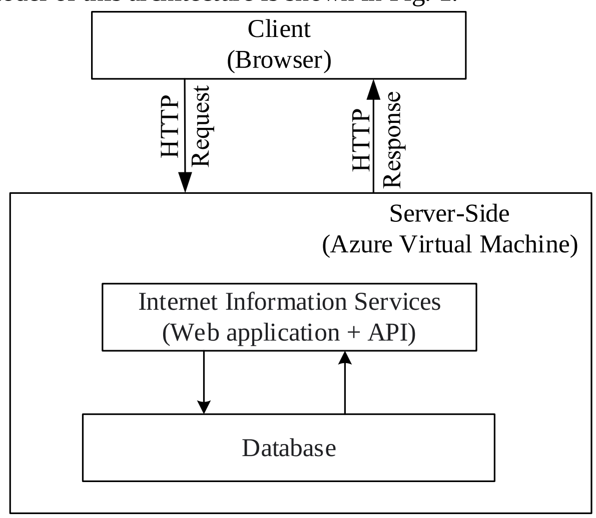

Fig. 1. Structure of server architecture using Azure VM

Server architecture using Azure VM predicted client and server sides. With this approach to the implementation of the architecture, the client and server parts of the software will be placed on a pre-configured IIS server. This server is located on a virtual machine of the selected cloud platform.

The Server part of the architecture includes next components:

- <u>*Internet Information Services (IIS)*</u> — the main service that will be started with a web application with API endpoints.
- <u>*Microsoft SQL Server*</u> — management system for deploying SQL Database.

Server architecture provides such features as:

- Full Access – can install or alter any software, quickly fix issues, and have absolute control over backups.

- Security – all of your critical data is safely stored on the premises and no third-party service has access to it, the server computing model is known for high security.

Serverless architecture – is an approach to software development that allows developers to create and run services without having to manage the underlying infrastructure. Developers can write and deploy code, and the cloud provider provides servers to run their applications, databases, and storage systems at any scale.

The general structure of this type of architecture is shown in Fig.2.

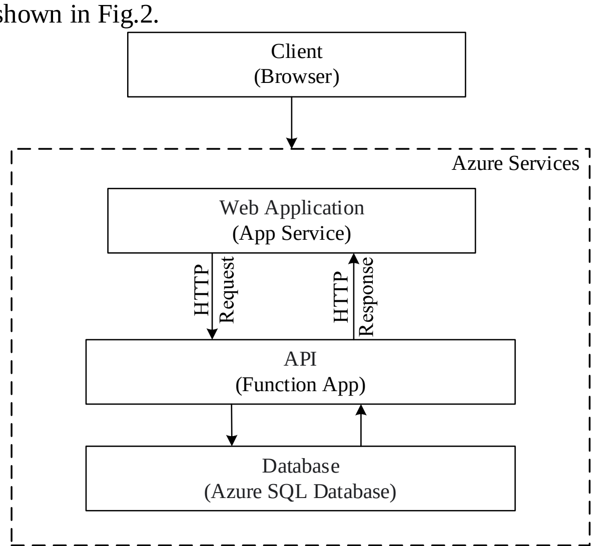

Fig. 2. Structure of serverless architecture using Azure services

The following services and components were used to implement the software:

- <u>*App Service*</u> – is a fully managed platform for building web applications. It is needed for deploying static web pages of the developed application.
- <u>*Function App*</u> – is the service of serverless computing that is used for creating API endpoints that communicate with the database using an HTTP connection. It provides execution of all processes and computing related to data.
- <u>*Azure SQL Database*</u> – is a fully managed platform as a service (PaaS) database engine that handles most of the database management functions such as upgrading, patching, backups, and monitoring without user involvement.

Serverless architecture provides next features:

- Flexibility – serverless models scale without your intervention.
- No need to run a server 24/7
- Less effort for maintaining project infrastructure
- Quick deployment
- Pay only for what you use.

B. DETERMINATION OF APPROACHES TO ASSESSMENT

There are several types of approaches to evaluating the computing power of a system.

Price-performance. The performance of the system plays a huge role in the operation of the program. The value of this characteristic depends on how the program will perform the set tasks and goals, cope with a large load, and ensure the integrity of the system. The computing resources of cloud platforms can be increased depending on the needs, but at the same time, the price of using these resources also increases. The correct combination of the specified characteristics determines the creation of effective software infrastructure at an optimal price policy.

The price-performance approach involves determining the most efficient system in terms of price and performance. This means that the system will meet all requirements, be fast and efficient in terms of performance and use of computing power, but at the same time will have the most favorable price for their creation, implementation, and maintenance.

Elasticity and scalability. The elasticity refers to the ability of a cloud to automatically expand or compress the infrastructural resources on a sudden up and down in the requirement so that the workload can be managed efficiently. This elasticity helps to minimize infrastructural costs. This does not apply to all kinds of environments, it is helpful to address only those scenarios where the resources requirements fluctuate up and down suddenly for a specific time interval. It is not quite practical to use where persistent resource infrastructure is required to handle the heavy workload. Cloud scalability is used to handle the growing workload where good performance is also needed to work efficiently with software or applications. Scalability is commonly used where the persistent deployment of resources is required to handle the workload statically.

Safety and security. Cloud computing brings several attributes that require special attention when it comes to trusting the system. The trust of the entire system depends on the data protection and prevention techniques used in it. The major issues in cloud computing include resource security, resource management, and resource monitoring.

There are three major potential threats in cloud computing, namely, security, privacy, and trust. Security plays a critical role in the current era of the dream vision of computing as a utility. It can be divided into four subcategories: safety mechanisms, cloud server monitoring or tracing, data confidentiality, and avoiding malicious insiders' illegal operations and service hijacking.

C. DEVELOPMENT OF THE BLOCK DIAGRAM OF THE APPLICATION

Evaluation of computing power characteristics requires means for data processing and obtaining the results of their evaluation according to the selected approach and selected services.

It is assumed that the web application developed for the evaluation process will implement simple operations with the database, the so-called CRUD operation.

CRUD is the acronym for CREATE, READ, UPDATE and DELETE. These terms describe the four essential operations for creating and managing persistent data elements, mainly in relational and NoSQL databases.

In CRUD operations:

- C is an acronym for creating, which means to add or insert data into the SQL table.
- R is an acronym for reading, which means retrieving or fetching the data from the SQL table.
- U is an acronym for the update, which means making updates to the records present in the SQL tables.
- D is an acronym for delete, which means removing or deleting the records from the SQL tables.

These operations are necessary to determine the performance of a given system, namely to estimate the access time to the database.

Fig. 3 shows the general block diagram of the application.

The main part of the application focused on the operations of reading, creating, updating, and deleting data. Two different modes of operation are defined for each operation

- One request – 1 operation. This mode of operation involves changing one record in the database after one request, regardless of the type of this operation
- One request – 1000 operations. With this approach, an array of data is changed at the same time in one query to the database, rather than each record individually.

After each type of operation, the access time to the database is read and this information is saved.

Another part of the program involves increasing the number of connections to the database to evaluate the system's concurrent query performance and test the system for robustness to multiple queries.

The implementation of CRUD operations is focused on determining the efficiency of the system when connecting to the database, using the IIS server architecture to create API functions and serverless using Function App to implement API requests.

D. SELECTION OF CHARACTERISTICS AND EVALUATION METRICS

Cloud metrics are logs of data generated by a cloud infrastructure or application. Using data, organizations can detect, monitor, and respond to various changes in the costs, security, and performance of their cloud environments.

By collecting, analyzing, and applying the right cloud metrics, we can:

- Optimize billing and cloud costs
- Ensure compliance and security management

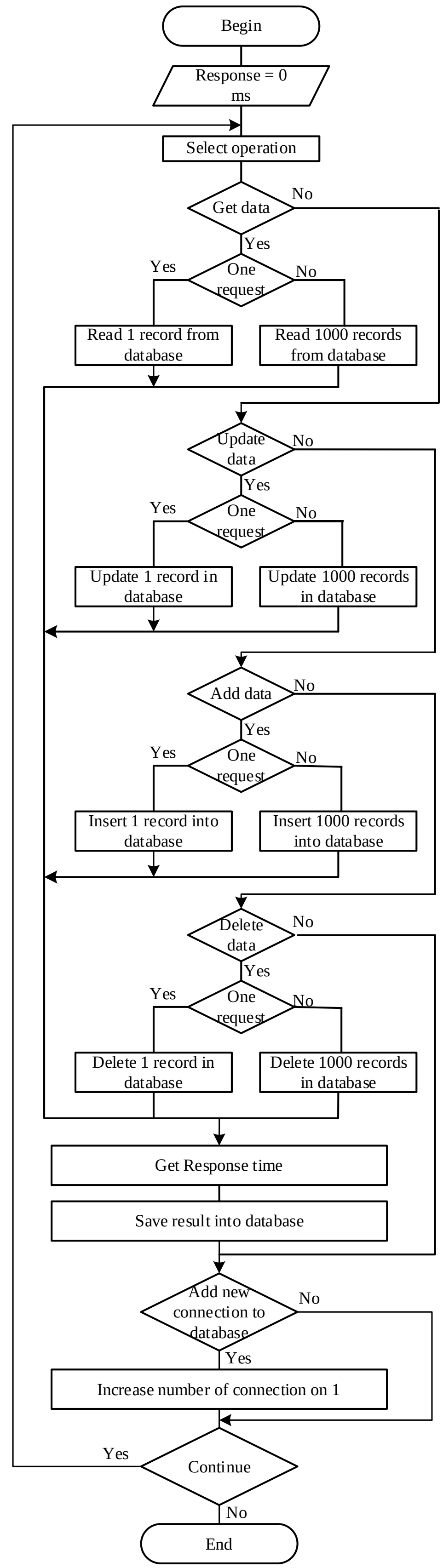

Fig. 3. Block diagram of an application

- Address issues as soon as they occur to prevent them from affecting your entire infrastructure
- Program performance management
- Properly allocate resources in the cloud

There are a large number of characteristics and metrics for evaluating cloud services. Among the main ones, we will choose the following indicators:

1) Response time – the period between a terminal operator’s completion of an inquiry and the receipt of a response. Response time includes the time taken to transmit the inquiry, process it by the computer, and transmit the response back to the terminal.

2) The number of connections – the total number of connections used by the system to access the database.

3) CPU Utilization – measures the percentage of computing units you are using. The trace will reveal whether the CPU is degrading performance due to under- or over-utilization.

4) Memory Usage – helps measure memory usage across public, private, and hybrid cloud environments.

5) Requests per minute – shows how many requests the cloud application receives every minute.

6) Disk Utilization – allows you to monitor the amount of disk space on a node to determine if it is sufficient for your workloads. Typical storage metrics include IOPS (input/output operations per second) and bandwidth.

7) Total Costs – the price of the developed infrastructure, calculated for each of the studied architectures, containing all the necessary services and settings, including databases.

Table 1.

Comparative characteristics of cloud services

Type of service	AZURE			
	Characteristics			Price
Virtual Machine	Core	RAM	Storage	
	1	1 GiB	4 GiB	$11.68/month
	1	2 GiB	4 GiB	$20.22/month
	2	4 GiB	8 GiB	$40.88/month
	2	8 GiB	16 GiB	$75.92/month
	4	16 GiB	32 GiB	$151.84/month
Cloud Functions	1M requests 400000 GB-sec			Free
	$0.20/1M+$0.000016/GB-s			-
Database	DTU[1]	Storage	Max Storage	
	10	250 Gb	250 Gb	$14.72/month
	20	250 Gb	250 Gb	$29.43/month
	50	250 Gb	250 Gb	$73.61/month
	100	250 Gb	1 Tb	$147.18/month
	200	250 Gb	1 Tb	$294.37/month

Table 1 shows the comparative characteristics of cloud services used for each type of architecture, taking into account the price and the provided capacities.

[1]Database transaction unit – represents a blended measure of CPU, memory, reads, and writes.

V. TEST EVALUATION OF THE MODEL

To evaluate performance, a basic study was conducted based on CRUD operations to compare the metric - time response for server and serverless architecture. For each type of operation, 10 requests were made to read, write, update, and delete data. After that, the response time for each request was measured. Fig. 4-7 shows a linear graph for the specified architectures from the response time during operations of reading, writing, editing, and deleting information from the database.

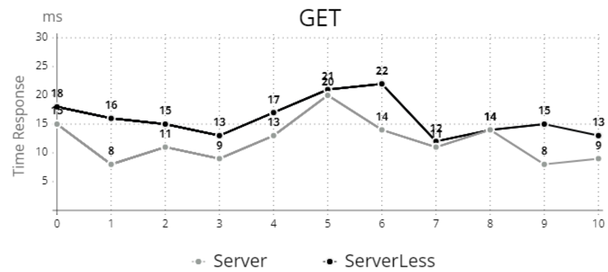

Fig. 4. Evaluation GET operation

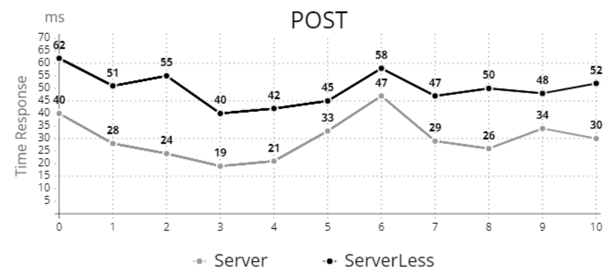

Fig. 5. Evaluation POST operation

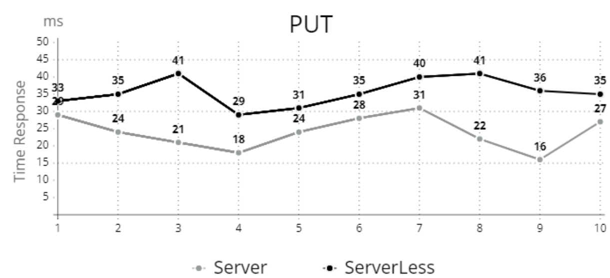

Fig. 6. Evaluation PUT operation

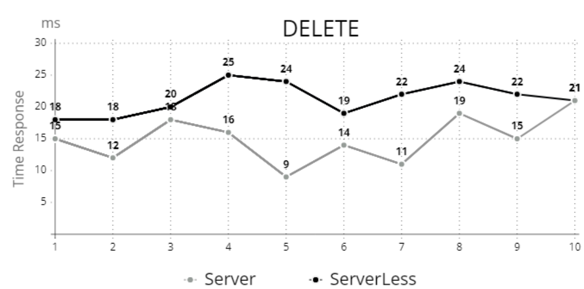

Fig. 7. Evaluation DELETE operation

The average measurement results are presented in Table 2

Table 2.

Research results

Type of operation	Time response (average)		Percentage
	Serverless	Server	
GET	17.6 ms	13.3 ms	24.4 %
POST	50 ms	30.1 ms	39.8 %
PUT	35.6 ms	24 ms	32.6 %
DELETE	21.3 ms	15.1 ms	29.1 %
In Total	**31.1 ms**	**20.7 ms**	**33.4 %**

An instance Azure SQL Database with a total performance of 10 DTU and storage of 250 GB was used for the database at $14.72/month.

For the server architecture were used Virtual Machine image B1s, 1 vCPUs, 1 GiB RAM, 4 GiB storage – $11.68/month cost and Azure database – $14.72/month.

For the serverless architecture, only database resource usage was charged, as 1 million requests for Function App are initially free. The data in Table 2 indicate that the server architecture is more productive than the serverless one. Taking into account the obtained results, the average response time of the server system is 20.7 ms, which is 33.4 % more productive than the serverless system, whose average time is 33.1 ms. Instead, the cost of implementing a serverless system is $14.72 versus $26.40 for a server system, which reduces the price of implementation by 44 %.

From these studies, it can be concluded that a server system should be used to create a more productive system, but if the main focus is on pricing, then serverless computing is better.

VI. CONCLUSIONS

Cloud platforms are widely used tools for software infrastructure integration and development.

In this article, the types of architectures for evaluating the capacities involved were defined. Among the analyzed architectures, it can be highlighted that server architectures provide full access to resources and are more secure. While scalability, fast implementation, and pay only when you use are features of serverless architecture. Approaches to evaluating system performance were identified, namely price performance, flexibility and scalability, and security.

A software model of the application was given to evaluate the performance, particularly the response time and the number of established connections to the database. Evaluation metrics were defined, the main ones were response time, number of connections, uptime, CPU utilization, memory usage, total price, and others. The conducted study showed that server architecture was 33.4 % more productive, but serverless was 44 % cheaper than server. Using a server was faster, which saves time and reduces system response time. Also, the comparative characteristics of the price and resources provided by Microsoft Azure platform were given.

References

[1] "The cloud and Microsoft azure fundamentals" (2019). *Microsoft Azure Infrastructure Services for Architects*. Wiley, pp. 1–46. DOI: 10.1002/9781119596608.ch1.

[2] Sharma, Y. *et al.* (2016) "Reliability and energy efficiency in cloud computing systems: Survey and taxonomy," *Journal of network and computer applications*, 74, pp. 66–85. DOI: 10.1016/j.jnca.2016.08.010.

[3] Mahmoudi, N. and Khazaei, H. (2022). "Performance modeling of metric-based serverless computing platforms," *IEEE transactions on cloud computing*, pp. 1–1. DOI: 10.1109/tcc.2022.3169619.

[4] Pérez, A. *et al.* (2018). "Serverless computing for container-based architectures," *Future generations computer systems: FGCS*, 83, pp. 50–59. DOI: 10.1016/j.future.2018.01.022.

[5] Pavych, N. and Pavych, T. (2019). "Method for time minimization of API requests service from cyber-physical system to cloud database management system," *Advances in Cyber-Physical Systems*, 4(2), pp. 125–131. DOI: 10.23939/acps2019.02.125.

Vladyslav Kotyk was born in Sarny City, Ukraine, in 2000. From 2006 to 2017, he studied at the Sarnensky Educational Complex "School-collegium" named after T. G. Shevchenko. In 2021 he received a Bachelor's degree by a specialization computer engineering at Lviv Polytechnic National University. Since 2021 he has been receiving a Master's degree at Lviv Polytechnic National University.

His research interests include image processing and segmentation, and object identification using neural. The main area includes cloud platforms such as Azure Portal and AWS and cloud computing.

Yevhenii Ya. Vavruk is an associate professor of the Computer Engineering Department at Lviv Polytechnic National University. He graduated from Lviv Polytechnic Institute in 1975 with a degree in Electrical Engineering, Faculty of Automation, Department of Electronic Computers. During 1975-2000 he worked at Lviv Research Radio Engineering Institute, and held the position of head of the group for the development of processors and signal processing systems. In this position he was engaged in the design, debugging, and testing of hardware exchange channels of information and measuring systems, processors for controlling moving objects, high-performance systems, and special computers for radar information processing.

In 2001 he successfully defended his dissertation on "Functionally-oriented processors for on-board control and information processing systems" at Lviv Polytechnic National University. From 2001 to 2005 he held the position of senior lecturer at the Department of Electronic Computers. Since 2005 he has been an associate professor of the Department of Electronic Computers.

His research interests are digital signal and image processing (design of algorithms and hardware for digital signal and image processing), and the design of parallel computational algorithms and structures.

He is the author of 98 copyright certificates for inventions in the field of computer engineering (32 implemented in the development), 35 articles in professional journals, 28 conference abstracts, and 27 educational and methodical developments (including 7 textbooks).