

**Науковий журнал**

# **ITSYNERGY**

---

**2023**

**Випуск 1 (4)**

---

Kyiv, 2023

# SCIENTIFIC JOURNAL IT SYNERGY

*Published since 2021 year*

*Two time a year*

ISSN 2786-7226

Kyiv, 2023, Issue 1 (4)

**Establishers:** Academician Yuriy Bugay International Scientific and Technical University

**CrossRef:** <http://doi.org/10.53920/ITS>

The journal publishes the results of scientific research in the following specialties:  
121 – Software engineering; 122 – Computer science;  
172 – Telecommunications and radio engineering.

**Editors:** **Artem Moskalenko**, Candidate of Technical Sciences,  
Associate Professor (Kyiv, Ukraine)

## **Editor board:**

**Anatolii Makarenko**, octor of Sciences (Technical), Professor (Kyiv, Ukraine)

**Volodymyr Nakonechnyi**, Doctor of Sciences (Technical), Professor (Kyiv, Ukraine)

**Olha Tkachenko**, Doctor of Sciences (Technical), Professor (Kyiv, Ukraine)

**Oleksandr Makoveichuk**, Doctor of Sciences (Technical), Assoc. Professor (Kyiv, Ukraine)

**Oleksandr Samkov**, Doctor of Sciences (Technical), Senior Research Officer (Kyiv, Ukraine)

**Valerii Koval**, Doctor of Sciences (Technical), Professor (Kyiv, Ukraine)

**Ihor Butko**, Doctor of Sciences (Technical), Assoc. Professor (Kyiv, Ukraine)

**Oleg Odarushchenko**, Doctor of Sciences (Technical), Professor (Kyiv, Ukraine)

**Jüri Vain**, Doctor of Science (Computer), Professor (Tallinn, Estonia)

**Michael Alexander Radin**, PhD, Assoc. Professor (New York, U.S.A.)

**Oleksandr Holubenko**, PhD (Technical), Assoc. Professor (Kyiv, Ukraine)

**Serhii Ivko**, PhD (Technical) (Poltava, Ukraine)

**Galina Sokol**, PhD (Technical), Assoc. Professor (Kharkiv, Ukraine)

**Artem Boyarchuk**, PhD (Technical), Assoc. Professor (Kyiv, Ukraine)

**Technical editor:** **Olha Brazhnikova**

Recommended for publication by the decision of the Academician Yuriy Bugay  
International Scientific and Technical University (Ukraine),  
protocol № 10/2223 from 29.06.2023

**Editorial board address:** Scientific journal «IT SYNERGY», Academician Yuriy Bugay  
International Scientific and Technical University,  
provulok Khersonskyi (Mahnitohorskyi), 3, Kyiv, 02094, Ukraine

☎: (095) 945-77-80    **e-mail:** [journal@istu.edu.ua](mailto:journal@istu.edu.ua)    **web:** <http://its.istu.edu.ua>

Registered by the Ministry of Justice of Ukraine Certificate of state registration  
of the print media Series KB № 24967-14907P dated 20.09.2021

© Academician Yuriy Bugay International  
Scientific and Technical University

# НАУКОВИЙ ЖУРНАЛ IT SYNERGY

Засновано у червні 2021 року

Виходить 2 рази на рік

ISSN 2786-7226

Київ, 2023, Випуск 1 (4)

**Засновник:** Заклад вищої освіти «Міжнародний науково-технічний університет імені академіка Юрія Бугая»

**CrossRef:** <http://doi.org/10.53920/ITS>

У журналі публікуються результати наукових пошуків зі спеціальностей:

121 – Інженерія програмного забезпечення; 122 – Комп'ютерні науки;

172 – Телекомунікації та радіотехніка.

**Головний редактор:** **Артем Олексійович Москаленко**, кандидат технічних наук, доцент, ЗВО «МНТУ» (Київ, Україна)

## Редакційна колегія:

**Анатолій Олександрович Макаренко**, доктор технічних наук, професор (Київ, Україна)

**Володимир Сергійович Наконечний**, доктор технічних наук, професор (Київ, Україна)

**Ольга Миколаївна Ткаченко**, доктор технічних наук, професор (Київ, Україна)

**Олександр Миколайович Маковейчук**, доктор технічних наук, доцент (Київ, Україна)

**Олександр Всеволодович Самков**, доктор технічних наук, старший науковий співробітник (Київ, Україна)

**Валерій Вікторович Коваль**, доктор технічних наук, професор (Київ, Україна)

**Ігор Миколайович Бутко**, доктор технічних наук, доцент (Київ, Україна)

**Олег Миколайович Одарущенко**, доктор технічних наук, професор (Київ, Україна)

**Jüri Vain**, доктор технічних наук, професор (Таллінн, Естонія)

**Michael Alexander Radin**, доктор філософії, доцент (Нью-Йорк, США)

**Олександр Іванович Голубенко**, кандидат технічних наук, доцент (Київ, Україна)

**Сергій Олександрович Івко**, кандидат технічних наук (Полтава, Україна)

**Галина Вікторівна Сокол**, кандидат технічних наук, доцент (Харків, Україна)

**Артем Володимирович Боярчук**, кандидат технічних наук, доцент (Київ, Україна)

**Технічний редактор:** **Ольга Ігорівна Бражнікова**

Рекомендовано до друку рішенням Вченої ради ЗВО «МНТУ»,  
протокол № 10/2223 from 29.06.2023

**Адреса редакції:** Науковий журнал «IT SYNERGY», ЗВО «МНТУ»,  
провулок Херсонський (Магнітогорський), 3, м. Київ, 02094, Україна

**телефон:** (095) 945-77-80 **e-mail:** [journal@istu.edu.ua](mailto:journal@istu.edu.ua) **web:** <http://its.istu.edu.ua>

Зареєстровано Міністерством юстиції України  
Свідоцтво про державну реєстрацію друкованого засобу масової інформації  
Серія KB № 24967-14907P від 20 вересня 2021 року

© ЗВО «МНТУ»

## ЗМІСТ

Валерій Вікторович ЗАВГОРОДНІЙ, Ганна Анатоліївна ЗАВГОРОДНЯ,  
Валентин Олександрович ГОЛОВАЧУК

**ПРОЄКТУВАННЯ СИСТЕМИ РАНЖУВАННЯ  
ПОТЕНЦІЙНИХ КЛІЄНТІВ ТЕЛЕКОМУНІКАЦІЙНОЇ КОМПАНІЇ  
НА ОСНОВІ МАШИННОГО НАВЧАННЯ.....6**

Марина Анатоліївна ДЕРЖЕВЕЦЬКА, Тетяна Володимирівна БАУЛІНА,  
Тамара Василівна КУХТІК, Тетяна Юріївна СОЛОМКО

**ПРОЄКТ ПРОГРАМНОГО КОМПЛЕКСУ ДЛЯ ОБРОБКИ  
МЕДИЧНИХ ДАНИХ З ВИКОРИСТАННЯМ МЕТОДІВ  
ДИСПЕРСІЙНОГО АНАЛІЗУ .....20**

Андрій Вікторович ЛЕМЕШКО, Артем Васильович АНТОНЕНКО,  
Дмитро Ігорович КОСТЕЦЬКИЙ, Максим Миколайович ШРАМ,  
Антон Сергійович ЗАКРЕНИЧНИЙ

**МОДЕЛЮВАННЯ БЕЗПРОВОДОВИХ МЕРЕЖ  
У СЕРЕДОВИЩІ ОМNET++ З ВИКОРИСТАННЯМ INET FRAMEWORK..... 37**

Олександр Андрійович ТКАЧЕНКО, Ольга Іванівна ТКАЧЕНКО,  
Олександр Олександрович ДЕЛАНТ

**СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ  
ЩОДО ВИБОРУ ВЕЛОСИПЕДУ .....60**

Валерій Вікторович ЗАВГОРОДНІЙ, Ганна Анатоліївна ЗАВГОРОДНЯ,  
Ігор Андрійович ЯКИМЕНКО, Максим Юрійович САВЧУК

**ПРОЄКТУВАННЯ ВІРТУАЛЬНИХ СЕРВЕРІВ  
НА ОСНОВІ ТЕХНОЛОГІЇ КОНТЕЙНЕРИЗАЦІЇ ..... 74**

Андрій Вікторович ЛЕМЕШКО, Артем Васильович АНТОНЕНКО,  
Віталій Олександрович СЛЮСАР, Дмитро Вадимович БАХУРИНСЬКИЙ,  
Микита Олександрович КУЦЕНКО

**ТЕХНОЛОГІЇ ХМАРНИХ ОБЧИСЛЕНЬ  
ТА ДИНАМІКА ЇХ РОЗВИТКУ .....89**

Олександр Анатолійович КОСТІКОВ, Тетяна Юріївна СОЛОМКО

**ОЦІНКА ЯКОСТІ ТЕСТОВИХ ЗАВДАНЬ МЕТОДАМИ  
СУЧАСНОЇ ТЕОРІЇ ТЕСТУВАННЯ ..... 109**

## CONTENTS

*Valerii ZAVGORODNII, Anna ZAVGORODNYA, Valentin HOLOVACHUK*

**DESIGNING OF THE SYSTEM FOR RANKING  
POTENTIAL CLIENTS OF A TELECOMMUNICATION COMPANY  
ON THE BASE OF MACHINE LEARNING.....6**

*Maryna DERZHEVETSKA, Tetyana BAULINA, Tamara KUKHTYK, Tetiana SOLOMKO*

**PROJECT OF A SOFTWARE COMPLEX FOR PROCESSING MEDICAL  
DATA USING METHODS OF DISPERSION ANALYSIS .....20**

*Andriy LEMESHKO, Artem ANTONENKO, Dmytro KOSTETSKYI,  
Maksym SHRAM, Anton ZAKRENYCHNYI*

**MODELING OF WIRELESS NETWORKS  
IN OMNET ++ ENVIRONMENT INVOLVING INET FRAMEWORK ..... 37**

*Olexandr TKACHENKO, Olha TKACHENKO, Oleksandr DELANT*

**DECISION SUPPORT SYSTEM REGARDING  
THE CHOICE OF A BICYCLE .....60**

*Valerii ZAVGORODNII, Anna ZAVGORODNYA, Ihor YAKYMENKO,  
Maksym SAVCHUK*

**DESIGN OF VIRTUAL SERVERS BASED  
ON CONTAINERIZATION TECHNOLOGY ..... 74**

*Andriy LEMESHKO, Artem ANTONENKO, Vitalii SLIUSAR,  
Dmytro BAKHURYNSKYI, Mykyta KUTSENKO*

**CLOUD COMPUTING TECHNOLOGIES  
AND DYNAMICS OF THEIR DEVELOPMENT .....89**

*Alexander KOSTIKOV, Tatyana SOLOMKO*

**ASSESMENT OF THE TEST ITEM QUALITY BY THE METHODS  
OF ITEM RESPONSE THEORY ..... 109**

УДК 004.855

DOI: <https://doi.org/10.53920/ITS-2023-1-1>

**Валерій Вікторович ЗАВГОРОДНІЙ,**

доктор технічних наук, професор,  
завідувач кафедри інформаційних технологій,  
Державного університету інфраструктури та технологій  
ORCID ID: 0000-0002-8347-7183

**Ганна Анатоліївна ЗАВГОРОДНЯ,**

кандидат технічних наук, доцент,  
доцент кафедри інформаційних технологій,  
Державний університет інфраструктури та технологій  
ORCID ID: 0000-0001-8523-1761

**Валентин Олександрович ГОЛОВАЧУК,**

магістр кафедри інформаційних технологій,  
Державний університет інфраструктури та технологій  
ORCID ID: 0009-0004-9001-6512

**ПРОЄКТУВАННЯ СИСТЕМИ РАНЖУВАННЯ  
ПОТЕНЦІЙНИХ КЛІЄНТІВ ТЕЛЕКОМУНІКАЦІЙНОЇ КОМПАНІЇ  
НА ОСНОВІ МАШИННОГО НАВЧАННЯ**

*Робота присвячена проєктуванню системи ранжування потенційних клієнтів телекомунікаційної компанії з огляду на їхню ймовірність придбання продукту на основі машинного навчання, що дозволить оптимізувати бізнес-процес роботи з «холодними» клієнтами.*

*У даній роботі розглядається call-центр телекомунікаційної компанії, спрямований на здійснення вихідних дзвінків з основною метою продажу послуг компанії. Незважаючи на те, що в різних галузях можуть бути власні особливості в діяльності call-центру, загальна модель організації залишається приблизно однаковою, що робить дане дослідження актуальним для будь-якої галузі.*

*Пропонується використовувати історичні дані для навчання алгоритму машинного навчання, який зможе емулювати діяльність супервайзера щодо формування завдань для call-центру.*

*Для кращого розуміння сучасного стану роботи call-центру в роботі наведена діаграма бізнес-процесу формування завдань на дзвінки у нотації BPMN. Також в роботі наведена діаграма, що де-*

**монструє вплив впровадження системи на основі машинного навчання на бізнес-процес формування завдань для обдзвону.**

**Дана робота досліджує задачу ранжування, яка може бути перетворена на задачу бінарної класифікації. В рамках класифікації необхідно визначити ймовірність належності потенційних клієнтів до одного з двох класів, що дозволяє вирішити задачу бінарної класифікації. Перший клас представляє клієнтів, зацікавлених у послугах компанії, тоді як другий клас охоплює клієнтів, які не проявляють інтересу до послуг компанії. У даному контексті найважливішою є ймовірність того, що потенційний клієнт належить до першого класу. Після отримання ймовірностей належності до першого класу, відбувається сортування всіх потенційних клієнтів у порядку спадання ймовірності, вирішуючи таким чином задачу ранжування.**

**Ключові слова:** call-центр, машинне навчання, навчання за прецедентами, система ранжування, аналіз бізнес-процесів, алгоритм класифікації.

**Valerii ZAVGORODNII**

Doctor of technical sciences, Professor,  
Head of the Department of Information Technologies  
State University of Infrastructure and Technologies

**Anna ZAVGORODNYA**

Candidate of technical sciences, Associate Professor,  
Associate Professor of the Department  
of Information Technologies  
State University of Infrastructure and Technologies

**Valentin HOLOVACHUK**

Master of the Department of Information Technologies,  
State University of Infrastructure and Technologies

## **DESIGNING OF THE SYSTEM FOR RANKING POTENTIAL CLIENTS OF A TELECOMMUNICATION COMPANY ON THE BASE OF MACHINE LEARNING**

**The work is devoted to the design of a system for ranking potential customers of a telecommunications company based on their probability of purchasing a product based on machine learning, which will allow optimizing the business process of working with «cold» customers.**

***This work examines the call center of a telecommunications company, aimed at making outgoing calls with the main purpose of selling the company's services. Despite the fact that different industries may have their own peculiarities in the activity of a call center, the general model of the organization remains approximately the same, which makes this study relevant for any industry.***

***It is proposed to use historical data to train a machine learning algorithm that will be able to emulate the activity of a supervisor in the formation of tasks for a call center.***

***For a better understanding of the current state of call center operation, the work presents a diagram of the business process of creating call tasks in BPMN notation. The paper also presents a diagram demonstrating the impact of the implementation of a system based on machine learning on the business process of creating tasks for callback.***

***This work investigates the ranking problem, which can be transformed into a binary classification problem. As part of the classification, it is necessary to determine the probability of potential customers belonging to one of two classes, which allows solving the problem of binary classification. The first class represents customers who are interested in the company's services, while the second class covers customers who do not show interest in the company's services. In this context, the most important thing is the probability that the potential customer belongs to the first class. After obtaining the probabilities of belonging to the first class, all potential customers are sorted in descending order of probability, thus solving the ranking problem.***

***Keywords:*** call center, machine learning, learning by precedents, ranking system, business process analysis, classification algorithm

**Постановка проблеми.** На сьогоднішній день галузь аналізу даних та штучного інтелекту надзвичайно динамічно розвивається, що доводиться щорічним зростанням кількості публікацій з цього напрямку [1]. Помітною є тенденція до зацікавлення штучним інтелектом не лише передових дослідницьких центрів та університетів, але й провідних технологічних компаній, які все частіше створюють власні лабораторії з штучного інтелекту. Цей рівень зацікавленості пояснюється тим, що компанії усвідомлюють великий потенціал методів машинного навчання та активно впроваджують їх у свої бізнес-процеси з метою досягнення значно покращеної ефективності.



Великі компанії проявляють традиційно великий інтерес до методів штучного інтелекту. Вони мають значний обсяг даних для навчання і доступ до ресурсів для дослідження та впровадження таких проєктів. Згідно з аналітичною компанією Gartner, за наступні 2 – 5 років методи машинного навчання та прогнозованої аналітики досягнуть плато продуктивності. Суспільство буде сприймати ці технології та їх впровадження в бізнес-процеси компаній як щось очевидне. Водночас будуть об'єктивно оцінюватися їх можливості, переваги та обмеження [2, 3].

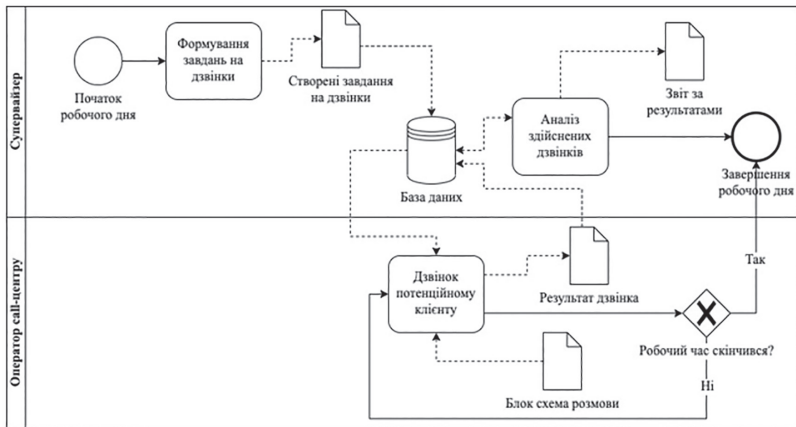
**Аналіз останніх досліджень і публікацій.** Методи машинного навчання вже сьогодні широко використовуються в галузі телекомунікаційних технологій. Телекомунікаційні компанії успішно вирішують багато складних завдань завдяки використанню цих методів. Особливу роль відіграє прогнозна аналітика, яка дозволяє отримати цінну інформацію з великих обсягів даних, які щодня генеруються в цій галузі.

Стаття, опублікована компанією Microsoft [4, 5], засвідчує успішне застосування методів машинного навчання в діяльності call-центрів. В ній приведені реальні приклади завдань, в яких методи машинного навчання здатні повністю або частково замінити людей. Особливістю цієї статті є те, що для кожного описаного завдання наведені ключові показники ефективності (Key Performance Indicators, KPI) та бізнес-цілі, які впливають на ці завдання, а також оцінка ступеня їх впливу. Крім цього, у статті міститься таблиця з описом рекомендованих алгоритмів та обсягів даних, необхідних для вирішення кожного з наведених завдань.

**Мета статті** – проєктування системи ранжування потенційних клієнтів телекомунікаційної компанії з огляду на їхню ймовірність придбання продукту на основі машинного навчання, що дозволить оптимізувати бізнес-процес роботи з «холодними» клієнтами.

**Виклад основного матеріалу дослідження.** У даній роботі розглядається call-центр телекомунікаційної компанії, спрямований на здійснення вихідних дзвінків з основною метою продажу послуг компанії. Незважаючи на те, що в різних галузях можуть бути власні особливості в діяльності call-центру, загальна модель організації залишається приблизно однаковою, що робить дане дослідження актуальним для будь-якої галузі.

Одним з важливих елементів будь-якого call-центру, спрямованого на вихідні дзвінки, є супервайзери, чия основна відповідальність полягає у формуванні завдань для операторів. При формуванні цих завдань супервайзери визначають, кому і в якому обсязі слід здійснювати дзвінки від імені call-центру. Це має стратегічне значення, оскільки ефективність роботи всього call-центру залежить від успішного вирішення цього завдання. Для кращого розуміння сучасного стану роботи call-центру була розроблена діаграма бізнес-процесу формування завдань на дзвінки у нотації BPMN [6 – 8]. Зазначена діаграма зображена на рисунку 1.



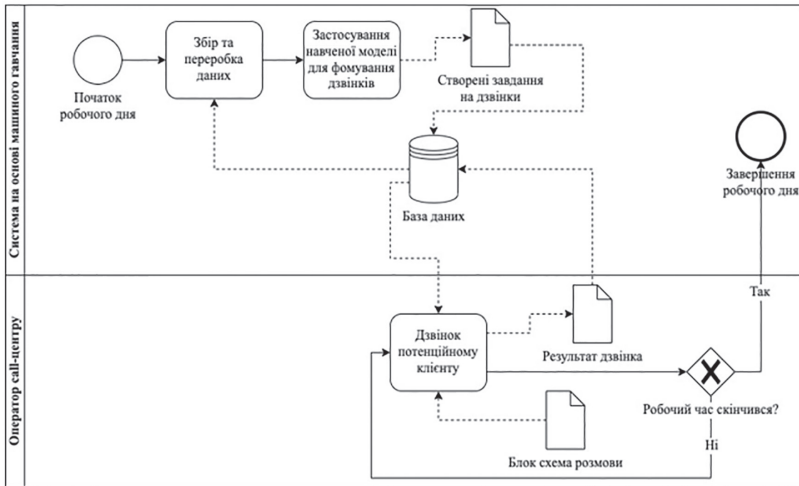
**Рис. 1. Формування завдань на дзвінок у нотації BPMN у стані AS IS**

У бізнес-процесі можна виділити кілька складних етапів. По-перше, щоденно супервайзеру доводиться створювати завдання та аналізувати їх результати після виконання. Засновуючись на аналізі здійснених дзвінків, супервайзер приймає рішення про те, з якими потенційними клієнтами краще зв'язатися з метою збільшення конверсії. Однак виникає проблема, оскільки ретельний та оперативний аналіз результатів є вельми складним завданням через вплив великої кількості факторів. Часто супервайзери мають покладатися на свій досвід та інтуїцію, що не завжди є найоптимальнішим для підвищення ефективності роботи call-центру. Багато часу витрачається на дзвінки до абонентів, які

не зацікавлені у підключенні, що призводить до неефективного використання робочого часу операторів і може шкодити репутації компанії.

Із рисунка 1 видно, що вся інформація про дзвінки, які були створені та оброблені, зберігається у базі даних. Завдяки цьому, використовуючи історичні дані, можна навчити алгоритм машинного навчання, який зможе емулювати діяльність супервайзера щодо формування завдань для call-центру.

Для демонстрації впливу впровадження системи на основі машинного навчання на бізнес-процес формування завдань для обдзвону було підготовлено діаграму у форматі BPMN, яка зображена на рисунку 2.



**Рис. 2. Формування завдань на обдзвін у нотації BPMN у стані TO BE**

Впровадження цього рішення повністю автоматизує процес формування завдань, що виключає необхідність участі супервайзерів. Це означає, що супервайзерам більше не доведеться витрачати час на рутинне формування завдань для call-центру, і вони зможуть приділити більше часу на інші пріоритетні завдання.

Крім того, впровадження системи ранжування може сприяти збільшенню конверсії продажів. Алгоритми машинного навчання

ефективно опрацьовують велику кількість змінних та історичних даних, набагато краще, ніж людський фактор.

Отже, ми маємо задачу спроектувати систему, яка щоденно буде генерувати завдання для телефонних дзвінків у call-центрі з найвищою ймовірністю продажу.

#### *Постановка задачі машинного навчання*

Перед тим, як розпочати проектування системи, важливо чітко визначити, яку задачу машинного навчання ми намагаємося вирішити. Машинне навчання – це галузь штучного інтелекту, що займається вивченням алгоритмів, які можуть самостійно навчатися. Існує кілька типів навчання, але в даній роботі ми будемо використовувати найпоширеніший – навчання за прецедентами.

Цей тип навчання базується на виявленні загальних закономірностей на основі конкретних емпіричних даних. Загалом, задача навчання за прецедентами полягає в тому, що ми маємо навчальний набір даних – скінченну множину прецедентів та їх описи. Наше завдання – виявити залежності, які властиві всім прецедентам, навіть тим, які не входять до навчального набору. Опис кожного прецедента представляється числовим вектором, що складається з ознак, які характеризують всі прецеденти навчального набору. Цей вектор визначає параметри алгоритму. Алгоритм машинного навчання знаходить такі значення параметрів моделі, при яких функціонал якості на навчальному наборі досягає оптимального значення.

Традиційно, існують кілька типів завдань навчання за прецедентами, які перераховані нижче [9, 10]:

1. Завдання класифікації: у цьому випадку необхідно призначити об'єктам одну з обмеженої множини можливих відповідей.

2. Завдання регресії: тут відповіддю є дійсне число, і метою є прогнозування числових значень.

3. Завдання ранжування: це полягає в сортуванні вхідних даних за певним критерієм. Важливо зазначити, що завдання ранжування також може бути сформульоване як завдання класифікації або регресії.

4. Завдання прогнозування: тут об'єктами є відрізки часових рядів, і метою є передбачення майбутніх значень на основі цих рядів.

Дана робота досліджує задачу ранжування, яка може бути перетворена на задачу бінарної класифікації. В рамках класифікації

ми маємо визначити ймовірність належності потенційних клієнтів до одного з двох класів, що дозволить вирішити задачу бінарної класифікації. Перший клас представляє клієнтів, зацікавлених у послугах компанії, тоді як другий клас охоплює клієнтів, які не проявляють інтересу до послуг компанії. У даному контексті нас головним чином цікавить ймовірність того, що потенційний клієнт належить до першого класу. Після отримання ймовірностей належності до першого класу, ми будемо сортувати всіх потенційних клієнтів у порядку спадання ймовірності, вирішуючи таким чином задачу ранжування.

#### *Методи розв'язання задачі класифікації*

Задача класифікації є однією з найбільш вивчених галузей машинного навчання, і вона має велику різноманітність алгоритмів з різними рівнями складності. Ці алгоритми починаються від простих статистичних методів і досягають ресурсоемних методів глибокого навчання.

На сьогоднішній день, серед найпопулярніших і ефективних алгоритмів у всіх галузях класифікації є алгоритми, засновані на деревах та нейронних мережах. Дерева є простими у розумінні інтерпретованими моделями, які добре працюють навіть на невеликих наборах даних. Вони використовують розбиття ознак на дерево рішень, що дозволяє легко визначити класифікаційні правила.

Нейронні мережі, зокрема багатoshарові перцептрони, здатні до складних нелінійних моделей та автоматичного вивчення високорівневих ознак. Вони здатні працювати з великими обсягами даних та виявляти складні залежності між вхідними і вихідними даними. Завдяки глибині та ширині нейронних мереж можна досягти вражаючих результатів у багатьох задачах класифікації.

Проте, необхідно враховувати, що існують ситуації, коли прості алгоритми можуть показувати кращі результати в окремих задачах. Це може статися з причини обмеженості даних або недостатньої кількості навчальних прикладів для ефективного навчання складних моделей. Тому варто розглядати широкий спектр алгоритмів і не виключати простих методів, наприклад, таких як наївний Баєс, логістична регресія або метод опорних векторів. Ці алгоритми можуть бути ефективними в ситуаціях, коли дані мають чітку структуру або коли кількість ознак обмежена.

Крім того, прості алгоритми зазвичай мають менші вимоги до обчислювальних ресурсів і можуть працювати швидше ніж склад-

ні моделі. Це особливо важливо, якщо є необхідність обробляти великі обсяги даних в реальному часі або на пристроях з обмеженими обчислювальними можливостями.

Отже, вибір алгоритму для задачі класифікації залежить від конкретного контексту, доступних даних, обчислювальних ресурсів та конкретних потреб. Варто провести експерименти з різними алгоритмами і здійснити порівняльний аналіз результатів, щоб знайти найбільш ефективне рішення для конкретної задачі класифікації.

#### *Вимоги до системи ранжування*

Під час проектування системи ранжування необхідно визначити вимоги, які повинні бути задоволені. Основні вимоги до системи включають наступне:

1. Збір ознакового опису потенційних клієнтів з корпоративної бази даних.

2. Використання навченої моделі машинного навчання для отримання ймовірності продажу.

3. Формування вибірок потенційних абонентів з найвищою ймовірністю продажу (розмір вибірок заздалегідь визначений).

4. Взаємодія з корпоративною системою формування завдань для call-центру.

5. Автоматичне оновлення моделі машинного навчання.

6. Формування звіту про оновлення моделі та його розсилка через корпоративну пошту.

7. Формування звіту про результати роботи моделі та його розсилка через корпоративну пошту.

#### *Загальна архітектура системи ранжування потенційних клієнтів*

На рисунку 3 зображена загальна структура системи у вигляді компонентної діаграми. Розроблювана система взаємодіє з корпоративною базою даних та системою формування завдань для call-центру, що є внутрішньою системою компанії. Остання забезпечує інтерфейс для взаємодії з розроблюваною системою.

Система складається з п'яти компонентів, кожен з яких виконує певні функціональні завдання, що описані нижче:

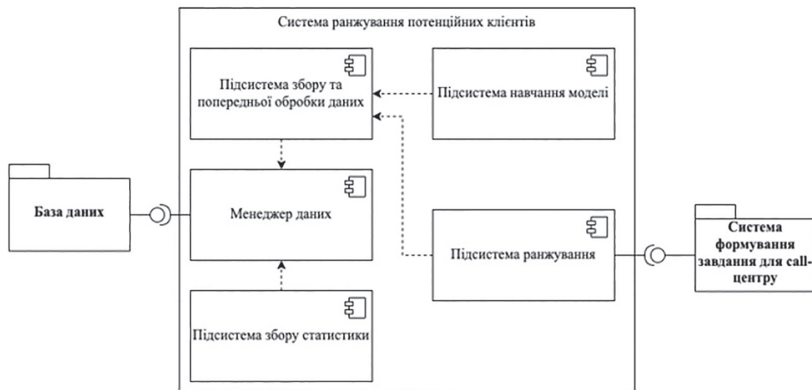
1. Менеджер даних: цей компонент відповідає за взаємодію з корпоративною базою даних і забезпечує основний функціонал обробки даних.

2. Підсистема збору та попередньої обробки даних: ця підсистема відповідає за формування навчальної вибірки та підготовку даних для щоденного ранжування.

3. Підсистема збору статистики: цей компонент виконує збір статистичних показників роботи системи з метою постійного моніторингу якості.

4. Підсистема навчання моделі: ця підсистема відповідає за підтримку навченої моделі машинного навчання в актуальному стані.

5. Підсистема ранжування: цей компонент взаємодіє із зовнішньою системою формування завдань для call-центру і відповідає за щоденне формування вибірок потенційних абонентів з найбільшою ймовірністю продажу.



**Рис. 3. Діаграма компонентів**

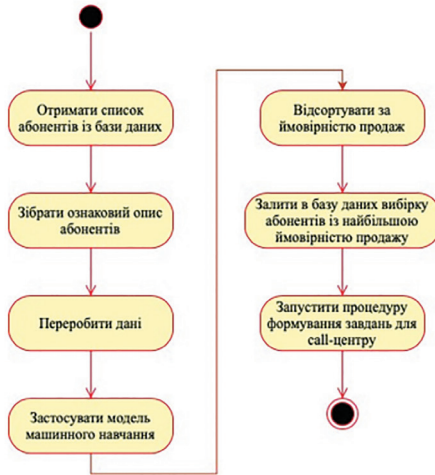
#### *Основний функціонал системи*

Розроблювана система має виконувати три основні функції, а саме:

1. Щоденне формування завдань для call-центру з максимальною ймовірністю продажу.
2. Оновлення навченої моделі машинного навчання.
3. Розсилка звіту з результатами роботи системи.

Виконання кожної з цих функцій має бути ініційоване операційною системою в наперед встановлений час.

На рисунку 4 представлена загальна схема роботи щоденного процесу формування завдань для call-центру.



**Рис. 4. Загальна схема процесу формування завдань**

З істотним перебігом часу, тенденції на ринку послуг інтернет-провайдерів можуть зазнавати змін. Певні ознаки можуть втратити свою прогностичну вагу, тоді як інші можуть набувати більшої важливості. З цієї причини, необхідно реалізувати функціонал системи, спрямований на підтримку актуальності моделі. На рисунку 5 наведена загальна схема оновлення моделі машинного навчання.



**Рис. 5. Загальна схема оновлення моделі**



Для моніторингу ефективності системи та своєчасного прийняття рішень, необхідно врахувати можливість збору статистики і формування звіту про результати роботи системи. Також, важливим є автоматична відправка цього звіту через корпоративний поштовий сервіс. Схема роботи, зображена у формі діаграми діяльності, представлена на рисунку 6.



**Рис. 6. Схема роботи процесу формування звіту**

Процес надсилання звітів може бути повторюваним, що дозволяє системі регулярно надсилати звіти про роботу моделі машинного навчання. Звіти, що відправляються, можуть бути використані для аналізу результатів, моніторингу продуктивності та прийняття рішень щодо подальшого розвитку моделі.

**Висновки та пропозиції.** Основна ціль роботи полягала в проектуванні системи, яка могла б ранжувати потенційних клієнтів залежно від ймовірності здійснення продажу. Це дозволило б не тільки замінити рутинну роботу супервайзерів під час формування завдань для call-центру, але й підвищити конверсію продажів. Для досягнення цієї цілі були виконані такі завдання:

- проведено аналіз бізнес-процесу формування завдань для call-центру;
- проведено порівняльний аналіз алгоритмів класифікації;
- спроектована система ранжування.

У подальшому розвитку планується виконати наступні кроки:

1. Провести розробку нових ознак, які зможуть якісно описати потенційних клієнтів компанії.

2. Здійснити експерименти з новими алгоритмами класифікації з метою покращення якості метрик.

3. Розробити зручний веб-інтерфейс для управління системою ранжування.

Ці кроки спрямовані на поліпшення функціональності та ефективності системи, забезпечуючи більш точний опис потенційних клієнтів, покращення класифікаційних алгоритмів та зручний інтерфейс для їх керування.

© **Завгородній В.В., Завгородня Г.А., Головачук В.О., 2023**

## ЛІТЕРАТУРА

1. Karen Hao. We analyzed 16,625 papers to figure out where AI is headed next. URL: <https://www.technologyreview.com/2019/01/25/1436/we-analyzed-16625-papers-to-figure-out-where-ai-is-headed-next/> (дата звернення: 30.05.2023).

2. Eka Ponkratova. Finally available and accessible, but wrong way? URL: <https://towardsdatascience.com/finally-available-and-accessible-but-wrong-way-bec731a151a6> (дата звернення: 30.05.2023).

3. Gartner Reveals. Five Major Trends Shaping the Evolution of Analytics and Business Intelligence. URL: <https://www.gartner.com/en/newsroom/press-releases/2019-10-02-gartner-reveals-five-major-trends-shaping-the-evoluti> (дата звернення: 30.05.2023).

4. Business Decision Makers. Machine learning implementation strategy for a customer service center. URL: <https://cloudblogs.microsoft.com/dynamics365/bdm/2018/02/07/machine-learning-implementation-strategy-for-a-customer-service-center/> (дата звернення: 30.05.2023).

5. Gartner Predicts. The Future Of AI Technologies. URL: <https://www.gartner.com/smarterwithgartner/gartner-predicts-the-future-of-ai-technologies> (дата звернення: 30.05.2023).

6. Pankaj E. Kasar. BPMN To REO Modeling Transformation. LAP LAMBERT Academic Publishing. 2017. P. 52. ISBN 9783330072145.

7. Vivek Kale. Enterprise Process Management Systems. CRC Press, 2018. P. 437. ISBN 9780429842337.

8. Philippe Desfray, Gilbert Raymond. Modeling Enterprise Architecture with TOGAF. Morgan Kaufmann. 2014. P. 350. ISBN 9780124199958.

9. Matthew Mayo. Top Data Science and Machine Learning Methods Used in 2018, 2019. URL: <https://www.kdnuggets.com/2019/04/top-data-science-machine-learning-methods-2018-2019.html> (дата звернення: 30.05.2023).

10. Royce Reinger. A Game Theoretic Approach to Explain The Output Of any ML Model. URL: <https://morioh.com/p/70356b3ecbc8> (дата звернення: 30.05.2023).

## REFERENCES

1. Hao, K. (2019), «We analyzed 16,625 papers to figure out where AI is headed next», available at: <https://www.technologyreview.com/2019/01/25/1436/we-analyzed-16625-papers-to-figure-out-where-ai-is-headed-next/> (Accessed 30 May 2023).

2. Ponkratova, E. (2020), «Finally available and accessible, but wrong way?», available at: <https://towardsdatascience.com/finally-available-and-accessible-but-wrong-way-bec731a151a6> (Accessed 30 May 2023).

3. Gartner Reveals (2019), «Five Major Trends Shaping the Evolution of Analytics and Business Intelligence», available at: <https://www.gartner.com/en/newsroom/press-releases/2019-10-02-gartner-reveals-five-major-trends-shaping-the-evoluti> (Accessed 30 May 2023).

4. Business Decision Makers (2018), «Machine learning implementation strategy for a customer service center», available at: <https://cloudblogs.microsoft.com/dynamics365/bdm/2018/02/07/machine-learning-implementation-strategy-for-a-customer-service-center/> (Accessed 30 May 2023).

5. Gartner Predicts (2019), «The Future Of AI Technologies», available at: <https://www.gartner.com/smarterwithgartner/gartner-predicts-the-future-of-ai-technologies> (Accessed 30 May 2023).

6. Kasar, P.E. (2017), «BPMN To REO Modeling Transformation». LAP LAMBERT Academic Publishing. P.52. ISBN 9783330072145.

7. Kale, V. (2018), «Enterprise Process Management Systems». CRC Press. P.437. ISBN 9780429842337.

8. Desfray P., Raymond G. (2014), «Modeling Enterprise Architecture with TOGAF». Morgan Kaufmann. P.350. ISBN 9780124199958.

9. Mayo, M. (2019), «Top Data Science and Machine Learning Methods Used in 2018, 2019», available at: <https://www.kdnuggets.com/2019/04/top-data-science-machine-learning-methods-2018-2019.html> (Accessed 30 May 2023).

10. Reinger, R. (2023), «A Game Theoretic Approach to Explain The Output Of any ML Model», available at: <https://morioh.com/p/70356b3ecbc8> (Accessed 30 May 2023).

**СТАТТЯ НАДІЙШЛА ДО РЕДАКЦІЇ 23.05.2023**

УДК 004:4'2

DOI: <https://doi.org/10.53920/ITS-2023-1-2>

**Марина Анатоліївна ДЕРЖЕВЕЦЬКА,**

кандидат економічних наук,  
викладач кафедри медичної фізики  
та інформаційних технологій  
Донецького національного медичного університету  
ORCID ID: 0000-0002-9952-4992

**Тетяна Володимирівна БАУЛІНА,**

доктор економічних наук, професор,  
професор кафедри менеджменту Закладу вищої освіти  
«Міжнародний науково-технічного університет  
ім. академіка Ю. Бугая»  
ORCID ID: 0000-0002-0110-849X

**Тамара Василівна КУХТІК,**

доктор технічних наук, професор,  
професор кафедри комп'ютерних наук Закладу вищої освіти,  
«Міжнародний науково-технічного університет  
ім. академіка Ю. Бугая»  
ORCID ID: 0000-0002-2000-3669

**Тетяна Юріївна СОЛОМКО,**

кандидат технічних наук, доцент,  
доцент кафедри цифрових технологій, менеджменту  
та адміністративного управління  
Донбаського інституту техніки та менеджменту  
Закладу вищої освіти «Міжнародний науково-технічного  
університет ім. академіка Ю. Бугая»  
ORCID ID: 0000-0002-3029-7920

**ПРОЄКТ ПРОГРАМНОГО КОМПЛЕКСУ ДЛЯ ОБРОБКИ  
МЕДИЧНИХ ДАНИХ З ВИКОРИСТАННЯМ МЕТОДІВ  
ДИСПЕРСІЙНОГО АНАЛІЗУ**

**Метод багатофакторного дисперсійного аналізу є одним з найважливіших інструментів в статистиці і покликаний дати оцінку впливу, як одного фактору, так і декількох на досліджувані вихідні змінні (ознаки). Актуальність даного статистичного методу визначається тим, що область застосування дисперсійного аналізу до-**

*сить широка. Його застосовують у медичних дослідженнях, хімічних експериментах, інженерних дослідженнях. Сучасні технології дозволяють досить швидко реалізувати даний метод і отримати результати з досить низькою ймовірністю помилки. Це сприяє зростанню продуктивності в багатьох сферах нашого життя і дозволяє швидше приймати вірні і найменш ризиковані рішення. Виконано порівняльний аналіз засобів розробки для програмування обробки медичних даних з використанням методів дисперсійного аналізу, існуючих програмних комплексів і систем для обробки медичних даних. Побудовано інформаційні та логічні моделі предметної області, і процесу обробки з використанням діаграмних методик SADT. Було проведено опис потоків, створено структуру баз даних, обмеження на дані, моделі програмного забезпечення мовою UML у вигляді діаграм. Спроектовано програмний комплекс для обробки медичних даних з використанням методів дисперсійного аналізу. Спроектований програмний комплекс в подальшому надає можливість обробки медичних даних без завантаження спеціальних програмних засобів статистичного аналізу даних, що заощадить гроші на купівлю таких додатків і придбання ліцензії на них.*

**Ключові слова:** медичні дані, діаграма, дисперсійний аналіз, UML, спеціаліст, користувач, однофакторний дисперсійний аналіз, PHP, CODEIGNITER.

**Maryna DERZHEVETSKA**

PhD in Economics,  
Lecturer at the Department of Medical Physics  
and information technologies,  
Donetsk National Medical University

**Tetyana BAULINA**

Doctor of Economics, Professor  
Professor of the Department of Management,  
Higher Educational Institution «Academician Yuriy Bugay  
International Scientific and technical university»

**Tamara KUKHTYK**

Doctor of Technical Sciences, Professor,  
Professor of the Department of Computer Science,  
Higher Educational Institution «Academician Yuriy Bugay  
International Scientific and technical university»

**Tetiana SOLOMKO**

Candidate of Technical Sciences, Associate Professor,  
Associate Professor of the Department of Digital Technologies,  
Management and administrative management  
Donbas Institute of Technology and Management,  
Higher Educational Institution «Academician Yuriy Bugay  
International Scientific and technical university»

## **PROJECT OF A SOFTWARE COMPLEX FOR PROCESSING MEDICAL DATA USING METHODS OF DISPERSION ANALYSIS**

***The method of multivariate variance analysis is one of the most important tools in statistics and is designed to provide an estimate of the influence of one or several factors on the studied output variables (characteristics). The relevance of this statistical method is determined by the fact that the field of application of variance analysis is quite wide. It is used in medical research, chemical experiments, and engineering research. Modern technologies make it possible to quickly implement this method and obtain results with a fairly low probability of error. This contributes to the growth of productivity in many areas of our lives and allows us to make the right and least risky decisions faster. A comparative analysis of development tools for programming medical data processing using methods of dispersion analysis, existing software complexes and systems for medical data processing was performed. Informational and logical models of the subject area and the processing process were built using diagrammatic SADT techniques. Flows were described, database structure, data restrictions, software models in the UML language in the form of diagrams were created. A software complex for processing medical data using methods of dispersion analysis was designed. In the future, the designed software complex provides the possibility of processing medical data without downloading special software tools for statistical data analysis, which will save money on the purchase of such applications and the purchase of a license for them.***

**Keywords:** *medical data, diagram, dispersion analysis, UML, specialist, user, one factor analysis of variance, PHP, CODEIGNITER.*

**Постановка проблеми.** У даний час швидкість і якість отримання та обробки інформації стали найважливішою умовою існування і прогресу всіх галузей наукової та практичної діяльності.

Ця тенденція не обійшла стороною і медицину. Кожен медичний працівник щохвилини має справу з великим об'ємом інформації представленої в чисельному, текстовому, графічному, звуковому і інших видах. Від ефективності її збору, зберігання, передачі та інтерпретації залежить якість і своєчасність діагностичних, лікувальних, профілактичних заходів та результативність роботи системи охорони здоров'я в цілому. Аналізом медичних даних займається медична статистика, яка уявляє собою одну з найважливіших інструментів доказової медицини.

Метод багатфакторного дисперсійного аналізу є одним з найважливіших інструментів у статистиці і покликаний дати оцінку впливу, як одного фактору, так і декількох на досліджувані вихідні змінні (ознаки). Актуальність даного статистичного методу визначається тим, що область застосування дисперсійного аналізу досить широка. Його застосовують у медичних дослідженнях, хімічних експериментах, інженерних дослідженнях. Сучасні технології дозволяють швидко реалізувати даний метод і отримати результати з низькою ймовірністю помилки. Це сприяє зростанню продуктивності в багатьох сферах нашого життя і дозволяє швидше приймати вірні і найменш ризиковані рішення.

Через вартість, англломовність і наявність короткострокових доступних демо-версій найкращим варіантом буде створити програмний продукт власної розробки, який не вимагає ні фінансових витрат, ні прив'язки до Інтернету, ні наявності ліцензії. Також його інтерфейс в будь-який момент можна адаптувати під мову використання і прив'язати до нього різну довідкову літературу [1 – 4].

У цій статті докладно розглядається програмна система, яка має більшу спеціалізацію, ніж існуючі системи, зокрема в медичній сфері, з урахуванням їх недоліків.

**Аналіз останніх досліджень і публікацій.** Аналіз сучасного стану статистичних програмних засобів показав, що зараз існує широкий вибір програмних пакетів для статистичної обробки даних [5, 6]. У цьому переліку представлені різноманітні програмні засоби, від EXCEL [7] до STATISTICA [8].

В залежності від поставлених завдань користувачеві потрібно вибрати оптимальний статистичний пакет. Зазвичай, оптимальним варіантом є програмний засіб, який поєднує високу продуктивність, потрібні функціональні можливості і доступну ціну.

При виборі пакету потрібно урахувувати такі основні характеристики: відповідність системним вимогам програмного забору до комп'ютерного обладнання користувача; відповідність можливостей вимогам поставлених завдань; обсяг можливих використаних даних для статистичного аналізу; кваліфікацію користувача в галузі статистики.

Розглянемо найбільш популярні та функціонально повні програмні продукти, які мають вбудовані інструменти для статистичного аналізу даних.

MS Excel – це найбільш згадуваний в літературі і використовуваний користувачами додаток. Це пов'язано з широким застосуванням цього програмного забезпечення, наявності не англійських версій, проте деякі статистичні функції є просто додатковими вбудованими формулами. Розрахунки зроблені при її допомозі не визнаються авторитетними у біомедичних журналах та відсутня можливість побудувати якісні наукові графіки [7].

SPSS – найбільш старий і використовуваний пакет для обробки медичних даних. Від своїх аналогів він відрізняється гнучкістю застосування. Але для роботи з цією програмою необхідна висока потужність і продуктивність комп'ютера, а також недоліком є висока ціна [9].

Пакет SAS має великий набір статистичних алгоритмів. Крім того, SAS надає користувачеві можливість підключення власних оригінальних алгоритмів, але поряд із цими перевагами виділяють головні недоліки системи – громіздкість, труднощі в освоєнні, високі вимоги до статистичної кваліфікації користувача, жорсткі вимоги до апаратної частини, а також ціна [10].

SYSTAT – це статистична система для персональних комп'ютерів, яка володіє зручним інтерфейсом, широким діапазоном та глибиною опрацювання функціонального наповнення. Можливості програми передбачені як для слабо підготовленого в статистиці користувача, так і для досвідченого статиста. Але на сьогоднішній день має несучасний інтерфейс, який є незручним і громіздким у користуванні, частина операцій доступна лише з командного рядка, відсутність відповідного редактора звітів [11].

MINITAB 14 є повнофункціональним пробним варіантом програми, яка працює 30 днів. Він досить зручний в роботі, має хороший інтерфейс і візуалізацію, проте відсутність формул для статистик у довідковому посібнику ускладнює аналіз значимості впливу різних чинників на прийняті рішення [12].



Поряд з комерційними статистичними пакетами існує досить велика кількість повністю безкоштовних статистичних програм і додатків. При цьому ряд безкоштовних програм не тільки не поступається, але і перевершує по функціональності комерційні додатки. Наведемо перелік основних безкоштовних програм для статистичної обробки даних: R – безумовно, найбільш потужний безкоштовний програмний інструмент з неймовірно широким набором бібліотек; EpiInfo – безкоштовний статистичний пакет, розробка якого підтримується Центром по контролю за захворюваннями США; OpenEpi – набір статистичних функцій, що дозволяє швидко застосувати відносно прості і часто використовувані статистичні тести; SOFA – дозволяє виконувати основні статистичні тести, але не дає можливості проводити регресійний аналіз; SEER-Stat – орієнтований на застосування в онкології безкоштовний статистичний пакет, розробка якого підтримується Інститутом онкології США; WINPEPI – програма для аналізу епідеміологічних даних. Поряд з цими існує ще близько 30 безкоштовних програм для статистичної обробки даних, які мають різну функціональність і «заточені» під різні цілі. Проте недоліком цих програм є обмежений строк їх використання, відсутній переклад інтерфейсу українською мовою та неспроможність їх використання для вузької спеціалізації, зокрема медичної [13].

**Метою статті** є опис спроектованого програмного комплексу для обробки медичних даних з використанням методів дисперсійного аналізу, що має більшу спеціалізацію, ніж існуючі системи, зокрема в медичній сфері, з урахуванням їх недоліків.

**Виклад основного матеріалу дослідження.** На даний момент існують системи, які здатні виконувати обробку даних з використанням методів дисперсійного аналізу. Завдяки сучасним програмним продуктам (таким як Statistica, SPSS), процес реалізації даного методу займає невелику кількість часу і зусиль. Однак більшість даних систем є комерційними, не володіють локалізацією.

Дослідження характеристик програмних засобів для статистичної обробки даних, зокрема медичних, показав, що усі програми статистичної обробки даних можна розділити на професійні, напівпрофесійні (популярні) і спеціалізовані. Статистичні програми відносяться до наукомісткого програмного забезпечення, ціна їх часто недоступна індивідуальному користувачеві. Професійні пакети мають велику кількість методів аналізу, популярні пакети –

кількість функцій, достатніх для універсального застосування. Спеціалізовані ж пакети орієнтовані на якусь вузьку область аналізу даних, причому всі творці таких продуктів заявляють, що саме їх продукт перевершує аналоги. Відсутність у більшості користувачів часу для освоєння таких програм, робить непростим її вибір [1].

Під час аналізу готових програмних продуктів для обробки медичних даних було виділено кілька пакетів які за своєю концепцією та функціональністю є схожими на заплановану програмну розробку. Зважаючи на цей факт запропоновано переглянути ці готові пакети, та визначити їх переваги та недоліки. Головним недоліком, як було визначено раніше є їх комерційність і дуже висока вартість, а також необхідність наявності ліцензії. Інтерфейс, як правило, англійською мовою. У більшості не наводяться математичні залежності, реалізовані в програмах, а просто вказуються прізвища авторів тих чи інших методів розрахунку.

Використовуючи результати тестування кращих зарубіжних статистичних програмних продуктів загального призначення [8 – 12] можна представити порівняльну характеристику програмних продуктів (табл. 1) зі шкалою оцінок від 0 до 10, до якої також додано аналіз авторської розробки програмного комплексу для обробки медичних даних з використанням методів дисперсійного аналізу.

**Таблиця 1. Порівняльна характеристика програмних продуктів**

Назва пакета	Різноманітність	Швидкість роботи	Якість вихідних форм	Легкість використання	Легкість навчання	Загальні оцінки потужності	Загальні оцінки зручності загальні оцінки зручності	Ціна	Інтегральна оцінка
SYSTAT	7	7	7	7	7	7	7	1	50
SAS	7	7	7	8	7	7	7	2	52
STATGRAPHICS	6	6	6	6	6	6	6	1	43
SPSS/PC+	6	6	6	6	6	6	6	1	43
PC-90 (BMDP)	7	7	7	7	7	7	7	1	50
MINITAB	4	4	4	4	4	4	4	2	30

Джерело: Розроблено авторами

Порівняльний аналіз підтвердив доцільність розробки програмного комплексу для обробки медичних даних з використанням методів дисперсійного аналізу.

Для обробки медичних даних за допомогою алгоритмів дисперсійного аналізу необхідно вибрати параметри експерименту, зокрема кількість вимірювань і кількість рівнів фактору, це необхідно для створення форми введення самих даних для аналізу. Після вибору параметрів експерименту проводиться створення і виведення форми введення експериментальних даних. Далі в цю форму вводяться дані, призначені для статистичної обробки. Вхідними даними для процесу автоматизації обробки даних з використанням методів дисперсійного аналізу є матриця даних, що складається зі згрупованих за певними властивостями експериментальних даних. Далі введені дані, які пройшли перевірку на коректність, проходять обробку, що управляється правилами виконання алгоритму розрахунку. Дані піддаються попередній обробці для виконання розрахунку. Далі проводиться розрахунок і інтерпретація даних. На виході отримуємо інтерпретовані дані для подання в веб-формі і оформлений звіт з аналізом. Дійовими особами в процесі дисперсійного аналізу є користувач і фахівець-статист (математик). На рисунку 1 зображена контекстна SADT – діаграма нульового рівня для бізнес-процесу «Обробка медичних даних з використанням методів дисперсійного аналізу».



**Рис. 1. Контекстна SADT-діаграма 0-го рівня бізнес-процесу «Обробка медичних даних з використанням методів дисперсійного аналізу»**

Джерело: Розроблено авторами

Як видно з діаграми (рис. 1), на вході процес отримує опції форми введення і вхідні дані. Управління здійснюють: параметри введення, правила перевірки, правила виконання алгоритму, шаблони представлення даних. Виконавцями даного процесу є фахівець-статист (математик) і користувач. На виході отримуємо веб-форму з поданням даних і звіт про аналіз.

Для більш точного проєктування бізнес-процесу «Обробка медичних даних з використанням методів дисперсійного аналізу» деталізуємо SADT-діаграму нульового рівня.

Цей бізнес-процес можна розділити на декілька етапів: вибір параметрів; підготовка даних; виконання алгоритму розрахунку; подання даних.

На вхід активності A1 (Вибір параметрів) подаються опції форми введення. Виконавцем, який проводить введення опцій є фахівець-статист (математик). Введення опцій регламентується правилами введення. На виході отримуємо форму введення для вхідних даних.

На вхід активності A2 (Підготовка даних) виходить підготовлена форма для введення експериментальних даних, введення даних в форму здійснює фахівець-статист (математик). Тут також проводиться перевірка введених даних на коректність. Введення регламентується правилами введення даних. На виході отримуємо дані, підготовлені для обробки.

В активності A3 (Виконання алгоритму розрахунку) на вхід подаються перевірені на коректність дані. Дані піддаються обробці, далі проводиться розрахунок і подальша інтерпретація даних для виведення. На виході отримуємо дані, готові для виведення користувачу. Виконавцем є фахівець-статист (математик).

В активності A4 (Представлення даних) на вхід подаються інтерпретовані дані, які виходять після розрахунку за заданим алгоритмом. Ці дані виводяться в спеціальній веб-формі, також формується звіт про аналіз. Звіт формує користувач.

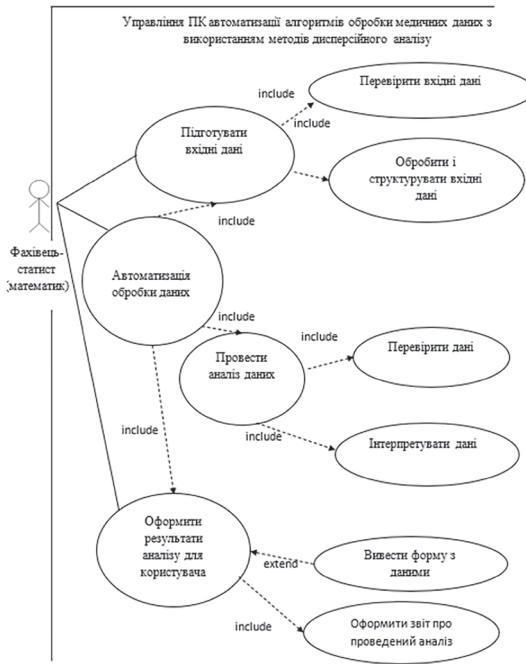
Деталізуюча структурно-функціональна діаграма першого рівня представлена на рисунку 2.

На основі аналізу предметної області та виявлених вимог щодо програмного продукту розроблена діаграма прецедентів використання, представлена на рисунку 3.



**Рис. 2. Деталізуюча структурно-функціональна SADT – діаграма першого рівня для бізнес-процесу «Обробка медичних даних з використанням методів дисперсійного аналізу»**

Джерело: Розроблено авторами



**Рис. 3. Діаграма прецедентів ПК ANOVA «Обробка медичних даних з використанням методів дисперсійного аналізу»**

Джерело: Розроблено авторами

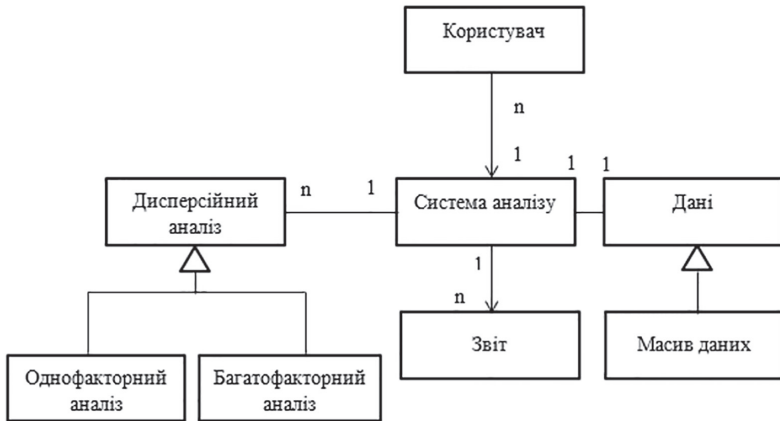
На етапі автоматизації обробки даних з системою взаємодіють один актор - фахівець-статист (математик). Він виконує основну частину дій, пов'язаних з веденням програмної частини ПК. Він обробляє дані, створює інтерфейси користувача, веде розрахунки, контролює процес аналізу даних і перевіряє адекватність даних.

Початковим етапом аналізу є підготовка. Дані можуть приходити в підготовленому вигляді, а саме в табличному вигляді, у вигляді масивів.

Наступним етапом є аналіз даних. Він включає в себе обробку статистичних даних методом дисперсійного аналізу та інтерпретацію даних для подання користувачеві.

На завершальному етапі виконується оформлення отриманих даних в зручній користувачеві формі.

На основі аналізу предметної області та виявлених вимог щодо розроблюваного програмного продукту, який розроблявся, було спроектовано діаграму класів предметної області, представлену на рисунку 4.



**Рис. 4. Діаграма класів предметної області**

Джерело: Розроблено авторами

На діаграмі класів відображені класи предметної області та їх зв'язки.

Клас користувач представляє користувача системи, він взаємодіє з елементами системи через клас Система аналізу, який надає йому для використання функції системи. Клас Дані – це дані, які отримує система для обробки за допомогою методів дисперсійного аналізу і успадковує клас Масив даних. Цей клас є матрицею вхідних даних заданого користувачем формату. Клас Звіт виконує функції формування і виведення звіту про випробування користувачеві. Клас Дисперсійний аналіз виконує безпосередньо аналіз даних і успадковує класи Однофакторний аналіз і багатofакторний аналіз, що представляють види дисперсійного аналізу.

Розроблено діаграму послідовності ПК для виклику функції «Підготовка звіту з аналізу». Розглянемо послідовність взаємодії об'єктів розробленої системи на прикладі використання функції «Отримати звіт з аналізу заданої інформації». В процесі дані обробляються, перевіряються на коректність відповідно до правил перевірки і піддаються обробці за допомогою методів дисперсійного аналізу. У разі успішного сценарію користувач ПК отримує готовий звіт з результатами проведеного аналізу.

Користувач взаємодіє з системою через інтерфейс, який представляє собою сукупність функцій, які можуть виконувати окремі частини системи. Інтерфейс викликає у об'єкта Валідатор даних функцію «Перевірити вхідні дані». Валідатор повертає перевірені відповідно до необхідних правил на коректність дані для подальшої статистичної обробки.

Далі інтерфейсний об'єкт викликає у об'єкта Система аналізу функцію «Провести аналіз ()», яка виробляє обробку даних за допомогою необхідних алгоритмів розрахунку дисперсійного аналізу і повертає результати, призначені для подальшої інтерпретації. Дані повертаються інтерфейсному об'єкту, який звертається до об'єкта Інтерпретатор даних. Цей об'єкт виконує функцію підготовки та обробки результатів дисперсійного аналізу і повертає їх у вигляді структур даних, з яких формується звіт з результатами аналізу. Після цього дані аналізу у вигляді звіту надаються для подальшої роботи з ними Користувачеві. Діаграма послідовності для виклику функції «Підготовка звіту з аналізу» представлена на рисунку 5.



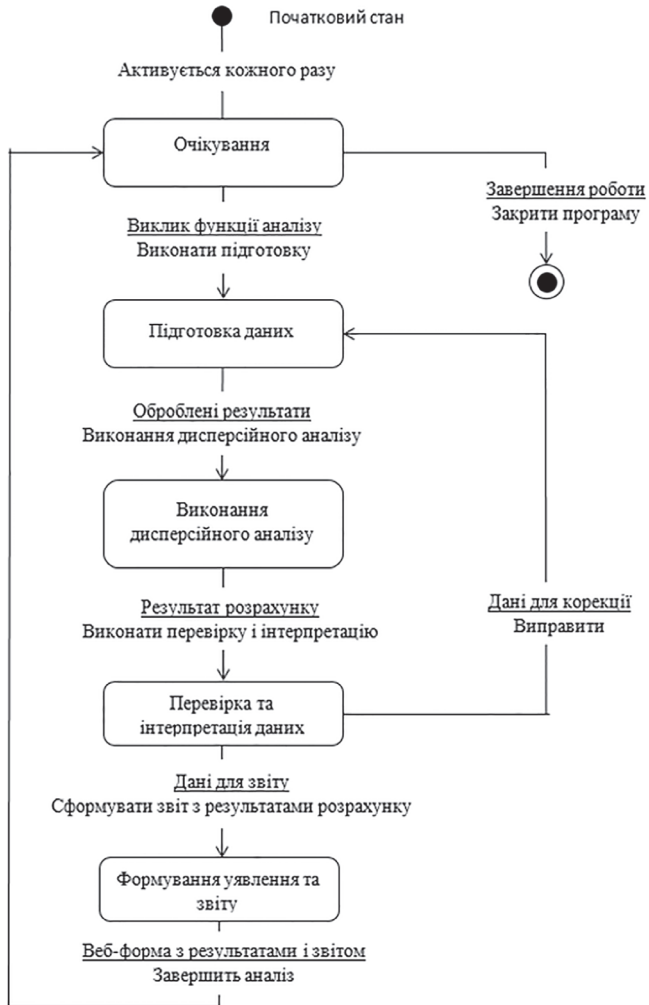
**Рис. 5. Діаграма послідовності для виклику функції «Підготовка звіту з аналізу»**

Джерело: Розроблено авторами

Наступним кроком було розроблено діаграму станів «ПК для обробки медичних даних з використанням дисперсійного аналізу». Початковим станом системи є Очікування. При надходженні нових інструкцій про необхідність проведення аналізу даних відбувається перехід до стану Підготовка даних. Умовою переходу в даний стан є наявність вхідних даних, які будуть оброблені та проаналізовані в подальшому. Дані перевіряються і обробляються за певними шаблонами. Далі необхідно виконати безпосередньо аналіз представлених даних. Відбувається перехід в стан Виконання аналізу. Умовою переходу в даний стан є наявність оброблених вхідних даних. Дія для переходу – виконання аналізу. Дією для наступного стану є виконання перевірки та інтерпретації даних. На даному етапі відбувається перевірка адекватності отриманих результатів. У разі виявлення значних недоліків отриманих результатів відбувається перехід на етап Підготовки даних для додаткової обробки даних для аналізу. У разі, коли модель вихідних даних пройшла перевірку, дані інтерпретуються в вид, зручний для подальшого подання і відбувається перехід в стан Формування відображення і звіту. Умовою переходу є дані для звіту. На даному етапі відбувається представлення результатів дисперсійно-



го аналізу в веб-формі і формування звіту. Після видачі системою звіту, процес аналізу вважається завершеним і система переходить в стан очікування, після цього можна завершити роботу з програмою, закривши її (рис. 6).



**Рис. 6. Діаграма станів для «ПК для обробки медичних даних з використанням дисперсійного аналізу»**

Джерело: Розроблено авторами

**Висновки та пропозиції.** Було проаналізовано існуючі програмні продукти для обробки медичних даних та виділено їх недоліки, з урахуванням яких було спроектовано програмний комплекс для обробки медичних даних з використанням методів дисперсійного аналізу. Розроблені діаграми бізнес–процесу, які відображають методологію структурного аналізу і проектування, процес моделювання, управління конфігурацією проекту. Розроблено діаграму прецедентів, яка показує з чим може взаємодіяти фахівець–статист (математик). Розроблено діаграму послідовностей, яка відображає як крок за кроком фахівець–статист (математик) взаємодіє з програмою. Розроблено діаграму класів у якій представлені класи, які взаємодіють між собою у системі розрахунків. Було розроблено діаграму відносин між таблицями баз даних у якій показується концептуальні схеми предметної області. Був проведений порівняльний аналіз серед систем обробки даних та проаналізовано їх переваги та недоліки.

Спроекований програмний комплекс дозволяє проводити обробку медичних даних без установки спеціальних додатків, що заощадить гроші на купівлю доповнених пакетів аналізу і ліцензії на них. Також його інтерфейс в будь-який момент можна адаптувати під мову використання і прив'язати до нього різну довідкову літературу.

© **Держевецька М. А., Бауліна Т.В., Кухтік Т.В., Соломко Т. Ю., 2023**

## ЛІТЕРАТУРА

1. Держевецька М. А., Рекова Н. Ю. ВИКОРИСТАННЯ СИСТЕМ КОМП'ЮТЕРНОЇ АЛГЕБРИ В НАУКОВИХ ДОСЛІДЖЕННЯХ // Автоматизація та комп'ютерно-інтегровані технології у виробництві та освіті: стан, досягнення, перспективи розвитку. С. 122. URL: [https://conference.ikto.net/pub/akit\\_2023\\_13-19march.pdf#page=122](https://conference.ikto.net/pub/akit_2023_13-19march.pdf#page=122) (дата звернення: 28.04.2023).

2. Рекова Н.Ю., Держевецька М.А. Використання методів візуалізації даних у наукових дослідженнях // *Сучасні інформаційні технології, засоби автоматизації та електропривод*: матеріали VII Всеукраїнської наук.-практ. конф. / За заг. ред. О. Ф. Тарасова. м. Краматорськ, 2023: ДДМА. 2023. С. 26 – 29.

3. Маценко К. В., Гетьман І. А. Аналіз програмного забезпечення для обробки медичних даних з використанням методів дисперсійного аналізу. 2021.

4. Васильєва Л.В. Автоматизовані системи наукових досліджень: посібник для студентів вищих навчальних закладів спеціальності «Інформаційні технології проектування»/ Л.В.Васильєва, І.А.Гетьман. Краматорськ : ДДМА, 2016. 114 с. ISBN 978-966-379-755-7.

5. Програми для математичної і статистичної обробки даних. URL: <http://chem-bio.com.ua/aspirant/grant/item/> (дата звернення 29.04.2023).

6. Програми для наукових розрахунків URL: <http://softlist.com.ua/catalog/programmy-dlya-nauchnykh-raschetov/> (дата звернення: 30.04.2023).

7. Функції Excel (за категоріями) URL: <https://support.office.com/uk-ua/article> (дата звернення: 30.04.2023).

8. StatSoft's Electronic Statistics URL: <http://www.statsoft.com/textbook/stathome.html> (дата звернення: 30.04.2023).

9. IBM SPSS Software URL: <https://www.ibm.com/analytics/us/en/technology/spss> (дата звернення: 30.04.2023).

10. SAS. Продукти та рішення URL: [https://www.sas.com/ru\\_ua/home.html](https://www.sas.com/ru_ua/home.html) (дата звернення: 30.04.2023).

11. Systat Software Inc.(SSI) URL: <http://www.sigmaplot.co.uk> (дата звернення: 30.04.2023).

12. Minitab URL: <https://www.minitab.com/en-us/products/minitab/> (дата звернення: 30.04.2023).

13. 26 найкращих програм для аналізу даних. URL: <https://techukraine.net/26-найкращих-програм-для-аналізу-даних/> (дата звернення: 1.05.2023).

14. Гетьман, І. А. Прогнозування раціону харчування людини за допомогою ІТ – технологій = Forecasting ration of the human nutrition with the help of IT technology / І. А. Гетьман, Т. В. Кухтик, М. А. Держевецька // 36. наук. пр. НУК. Миколаїв : НУК, 2021. № 1 (484). С. 80 – 85.

15. Select 'Admin'. URL: [https://www.emagazin.info/uk/select\\_admin](https://www.emagazin.info/uk/select_admin) (дата звернення: 1.05.2023).

## REFERENCES

1. Derzhevetska, M., Rekova, N. (2023) «VYKORYSTANNIA SYSTEM KOMPIUTERNOI ALHEBRY V NAUKOVYKH DOSLIDZHENNIakh //Avtomatyzatsiia ta kompiuterno-intehrovani tekhnolohii u vyrobnytstvi ta osviti: stan, dosiahnennia, perspektyvy rozvytku» [USE OF COMPUTER ALGEBRA SYSTEMS IN SCIENTIFIC RESEARCH //Automation and computer-integrated technologies in production and education: status, achievements, development prospects], available at: [https://conference.ikto.net/pub/akit\\_2023\\_13-19march.pdf#page=122](https://conference.ikto.net/pub/akit_2023_13-19march.pdf#page=122) (Accessed 30.04.2023).

2. Reкова, N., Derzhevetska, M. (2023) «Vykorystannia metodiv vizualizatsii danykh u naukovykh doslidzhenniakh» [Use of data visualization methods in scientific research]. Kramatorsk, Ukraine.
3. Matsenko, K., Getman I. (2021) «Vykorystannia metodiv vizualizatsii danykh u naukovykh doslidzhenniakh» [Analysis of software for processing medical data using variance analysis methods].
4. Vasylieva, L., Getman, I. (2016) «Avtomatyзовani systemy naukovykh doslidzen:posibnyk dlia studentiv vyshchyykh navchalnykh zakladiv spetsialnosti "Informatsiini tekhnolohii proektuvannia"» [Automated systems of scientific research: a manual for students of higher educational institutions majoring in «Information Design Technologies»]. Kramatorsk, Ukraine. ISBN 978-966-379-755-7.
5. «Prohramy dlia matematychnoi i statystychnoi obrobky danykh» [Programs for mathematical and statistical data processing] available at: <http://chem-bio.com.ua/aspirant/grant/item/> (Accessed 29.04.2023).
6. «Prohramy dlia naukovykh rozrakhunkiv» [Programs for scientific calculations] available at: <http://softlist.com.ua/catalog/programmy-dlya-nauchnykh-raschetov/> (Accessed 30.04.2023).
7. Excel functions (by category) available at: <https://support.office.com/uk-ua/article> (Accessed 30.04.2023).
8. StatSoft's Electronic Statistics available at: <http://www.statsoft.com/textbook/stathome.html> (Accessed 30.04.2023).
9. IBM SPSS Software available at: <https://www.ibm.com/analytics/us/en/technology/spss> (Accessed 30.04.2023).
10. SAS. Продукты та рішення available at: [https://www.sas.com/ru\\_ua/home.html](https://www.sas.com/ru_ua/home.html) (Accessed 30.04.2023).
11. Systat Software Inc.(SSI) available at: <http://www.sigmaplot.co.uk> (Accessed 30.04.2023).
12. Minitab available at: <https://www.minitab.com/en-us/products/minitab/> (Accessed 30.04.2023).
13. «26 найкращих програм для аналізу даних» [26 Best Software for Data Analysis], available at: <https://techukraine.net//26-найкращих-програм-для-аналізу-даних/> (Accessed 29.04.2023).
14. Getman, I., Kukhtyk, T., Derzhevetska, M. (2021) «Prohnozuvannia ratsionu kharchuvannia liudyny za dopomohoiu IT – tekhnolohii» [Forecasting ration of the human nutrition with the help of IT technology]. Mykolayiv, Ukraine.
15. Select 'Admin', available at: [https://www.emagazin.info/uk/select\\_admin](https://www.emagazin.info/uk/select_admin) (Accessed 30.04.2023).

**СТАТТЯ НАДІЙШЛА ДО РЕДАКЦІЇ 12.06.2023**

УДК 004.9:658

DOI: <https://doi.org/10.53920/ITS-2023-1-3>

**Андрій Вікторович ЛЕМЕШКО,**

доктор філософії з комп'ютерної інженерії, доцент,  
Державний університет телекомунікацій  
ORCID ID: 0000-0001-8003-3168

**Артем Васильович АНТОНЕНКО,**

кандидат технічних наук, доцент,  
Державний університет телекомунікацій  
ORCID ID: 0000-0001-9397-1209

**Дмитро Ігорович КОСТЕЦЬКИЙ,**

магістр,  
Державний університет телекомунікацій  
ORCID ID: 0009-0007-2821-7876

**Максим Миколайович ШРАМ,**

магістр,  
Державний університет телекомунікацій  
ORCID ID: 0009-0007-0640-1349

**Антон Сергійович ЗАКРЕНИЧНИЙ,**

магістр,  
Державний університет телекомунікацій  
ORCID ID: 0009-0006-3838-3539

## **МОДЕЛЮВАННЯ БЕЗПРОВОДОВИХ МЕРЕЖ У СЕРЕДОВИЩІ OMNET++ З ВИКОРИСТАННЯМ INET FRAMEWORK**

**Стаття присвячена опису методів моделювання безпроводових мереж у середовищі OMNeT++ з використанням INET Framework. У статті розглядаються основні поняття, пов'язані з моделюванням безпроводових мереж, а також описано основні вимоги до моделей безпроводових мереж. Далі, у статті розглядається середовище моделювання OMNeT++, його основні характеристики та переваги в порівнянні з іншими середовищами моделювання. Стаття розглядає основні складові моделей безпроводових мереж, такі як моделі мережевого рівня, моделі фізичного рівня, а також моделі радіоканалу. Для кожної з цих складових статті розглядаються методи моделювання та описуються можливі нюанси, пов'язані з вибором різних параметрів моделювання. У статті пропонується**

детальний опис структури моделі безпроводової мережі на прикладі мережі стандарту IEEE 802.11. У цій частині статті розглядаються такі поняття, як вузол, точка доступу, а також різні типи пакетів, що передаються в мережі. Також у статті розглядаються методи моделювання різних типів мережевих протоколів, наприклад, протоколу MAC, протоколу рівня мережі та інших. У заключній частині статті розглядаються питання тестування моделей безпроводових мереж та їх аналізу. Описується методика тестування та порівняння різних моделей, а також розглядається питання аналізу результатів моделювання та їх інтерпретації. У цілому, стаття пропонує детальний опис методів моделювання безпроводових мереж у середовищі OMNeT++ з використанням INET Framework. Розглянуто один із можливих підходів до проєктування та дослідження безпроводових мереж у середовищі імітаційного моделювання OMNeT++ з використанням фреймворку INET. Показано процес моделювання як самих моделей, так і різних режимів їх роботи. Також розглянуто методологію аналізу роботи моделі на прикладі тимчасової діаграми. При розгляді підходу використано метод послідовного укладання модельованих безпроводових мереж з урахуванням особливостей режимів їх роботи. Показано простоту реалізації підходу, в якому для моделювання безпроводових мереж використовуються готові компоненти зі складу фреймворку INET. Наведено методику проєктування безпроводових мереж з використанням готових компонентів зі складу фреймворку INET. Продемонстровано можливість перевизначення вбудованих компонентів складеного модуля для створення моделі пристрою, що відповідає необхідним вимогам. Розглянуто можливі режими роботи безпроводових мереж при прямому взаємодії вузлів, і при непрямому через проміжні вузли. Продемонстровано підхід до моделювання фізичного рівня та явища інтерференції. Розглянуто підхід дослідження проблемної ситуації, що полягає у аналізі тимчасової діаграми. Практична значимість. Запропоноване середовище імітаційного моделювання дозволяє проводити дослідження проєктних рішень під час проєктування безпроводових мереж. Обґрунтовано доцільність застосування розроблених проєктів у проєктній діяльності.

**Ключові слова:** безпроводові мережі, комп'ютерні мережі, Wi-Fi, імітаційне моделювання, інтерференція, OMNeT++, INET, framework.

**Andriy LEMESHKO**

Doctor of philosophy in computer engineering,  
Associate professor,  
State University of Telecommunications

**Artem ANTONENKO**

Candidate of technical sciences, Associate professor,  
State University of Telecommunications

**Dmytro KOSTETSKYI**

Master's student,  
State University of Telecommunications

**Maksym SHRAM**

Master's student,  
State University of Telecommunications

**Anton ZAKRENYCHNYI**

Master's student,  
State University of Telecommunications

**MODELING OF WIRELESS NETWORKS  
IN OMNET ++ ENVIRONMENT INVOLVING INET FRAMEWORK**

*The article is devoted to the description of the methods of modeling wireless networks in the OMNeT++ environment using the INET Framework. The article discusses the main concepts related to the modeling of wireless networks, and also describes the main requirements for models of wireless networks. Next, the article examines the OMNeT++ simulation environment, its main characteristics and advantages compared to other simulation environments. The article considers the main components of wireless network models, such as network layer models, physical layer models, and radio channel models. For each of these components, the article considers modeling methods and describes possible nuances associated with the choice of various modeling parameters. The article offers a detailed description of the structure of the wireless network model using the IEEE 802.11 network as an example. This part of the article covers concepts such as a node, an access point, and the different types of packets transmitted on a network. The article also discusses methods of modeling various types of network protocols, for example, MAC protocol, network layer protocol, and others. The final part of the article deals with testing wireless network models and their analysis. The methodology of testing and comparing different models is described, as*

***well as the issue of analysis of modeling results and their interpretation is considered. In general, the article offers a detailed description of the methods of modeling wireless networks in the OMNeT++ environment using the INET Framework. One of the possible approaches to the design and research of wireless networks in the OMNeT++ simulation environment using the INET framework is considered. The process of modeling both the models themselves and their various modes of operation is shown. The methodology of analyzing the model's operation using the example of a time diagram is also considered. When considering the approach, the method of sequential laying of simulated wireless networks was used, taking into account the peculiarities of their operation modes. The ease of implementation of the approach is shown, in which ready-made components from the INET framework are used for modeling wireless networks. The method of designing wireless networks using ready-made components from the INET framework is presented. The possibility of redefining the built-in components of the assembled module to create a device model that meets the necessary requirements is demonstrated. Possible modes of operation of wireless networks with direct interaction of nodes, and with indirect interaction through intermediate nodes are considered. An approach to modeling the physical level and the phenomenon of interference is demonstrated. The approach of studying the problem situation, which consists in the analysis of the time diagram, is considered. Practical significance. The proposed environment of simulation modeling allows to conduct research of design solutions during the design of wireless networks. The expediency of using the developed projects in project activity is substantiated.***

***Keywords:*** wireless networks, computer networks, Wi-Fi, simulation modeling, interference, OMNeT ++, INET Framework.

**Постановка проблеми.** Проектування будь-якої сучасної інформаційної системи, що має складну структуру з широким набором протоколів, завжди починається з побудови та дослідження її імітаційної моделі [1]. Метою моделювання є визначення оптимальної топології, адекватний вибір мережного обладнання, визначення робочих характеристик мережі та можливий розвиток. Однією з переваг імітаційного моделювання є можливість проведення низки досліджень, що дозволяють визначити надійність системи та її стійкість у разі відмови обладнання [2, 3]. Проведення подібних досліджень на працюючій



мережі неможливо, тому що це може негативно позначитися на стабільності її роботи, до того ж при виході з ладу обладнання, що використовується, є ризик зазнати фінансових втрат. Точне моделювання досліджуваного устаткування дозволяє одержати такі самі результати роботи, як і при реальному використанні цього устаткування, при цьому дозволяє заощадити кошти з його купівлі.

На сьогоднішній день існує безліч засобів імітаційного моделювання, до яких застосовуються досить жорсткі вимоги, такі як: детальна реалізація протоколів усіх рівнів, можливість підключення власних модулів, можливість зміни параметрів імітаційної моделі, платформна незалежність, розвинений графічний інтерфейс, а також доступність продукту та його ціна. Одним з таких інструментів, що відповідає переліченим вимогам, є середовище імітаційного моделювання OMNeT++ [4], що має розвинений графічний інтерфейс як для побудови моделей, так і для аналізу отриманих результатів. Ще однією важливою перевагою є її доступність, при цьому функціонал середовища не поступається іншим платним засобам імітаційного моделювання.

**Аналіз останніх досліджень і публікацій.** На даний момент, однією з найбільш цікавих статей з цієї теми є «Design and implementation of wireless sensor network simulation using the OMNeT++ simulator» авторів A. Al-Gburi та S. Ngan. У цій статті автори розглядають питання проектування та реалізації безпроводових сенсорних мереж з використанням OMNeT++ та INET Framework. Дослідження проводилося шляхом створення моделі мережі, яка відображає реальну мережу, та її подальшої симуляції. Ще однією цікавою статтею на цю тему є «Comparative Analysis of Wireless Sensor Network Simulation Tools» авторів M. Shahbaz, S. Ullah та S. S. Khan. У цій статті проведено порівняльний аналіз інструментів моделювання мереж, зокрема OMNeT++, NS2, NS3, QualNet та OPNET. Дослідження було зосереджено на аналізі різних аспектів, таких як можливості моделювання мереж, продуктивність, точність результатів та можливості розширення. Крім того, в статті «OMNeT++ based Wireless Sensor Network Simulation» авторів P. Yadav та S. Singh описано використання OMNeT++ та INET Framework для моделювання безпроводових сенсорних мереж. У дослідженні автори зосередились на аналізі продуктивності та точності результатів моделювання.

Загалом, зазначені статті свідчать про великий інтерес до використання OMNeT++ та INET Framework для моделювання безпроводових мереж. Водночас, є певна кількість статей, які порівнюють різні інструменти моделювання, щоб визначити їх переваги та недоліки у порівнянні один з одним. Наприклад, у статті «Comparative study of simulation tools for wireless sensor network» авторів N. Kumar та M. Kumar проведено порівняльний аналіз різних інструментів моделювання, включаючи OMNeT++, NS2, NS3 та QualNet. Автори порівняли ці інструменти за кількома критеріями, такими як продуктивність, точність результатів та можливості розширення, та прийшли до висновку про переваги та недоліки кожного з них. Також, стаття «Performance evaluation of wireless sensor network protocols using OMNeT++ simulation» авторів S. Singh та P. Yadav присвячена оцінці продуктивності протоколів безпроводових сенсорних мереж з використанням OMNeT++ та INET Framework. У дослідженні було проведено порівняльний аналіз різних протоколів, а також вивчено вплив різних факторів на продуктивність мережі.

Одна з найбільш цікавих українських статей є «Моделювання та аналіз безпроводових мереж на основі стандарту IEEE 802.15.4 за допомогою платформи OMNeT++» авторів М. Кульчицької, Є. Макаренка та Ю. Хрусталева. У цій статті розглядається питання проектування та моделювання безпроводових мереж на основі стандарту IEEE 802.15.4 з використанням OMNeT++ та INET Framework. Автори провели дослідження різних параметрів мережі, таких як продуктивність та ефективність мережі, а також порівняли результати моделювання з експериментальними даними.

Окрім цього, стаття «Моделювання та дослідження безпроводових мереж на базі протоколу ZigBee з використанням платформи OMNeT++» авторів І. Гринь та Є. Дроздової присвячена моделюванню та дослідженню безпроводових мереж на базі протоколу ZigBee з використанням OMNeT++. У дослідженні автори вивчили вплив різних параметрів мережі на її продуктивність та ефективність.

Таким чином, можна сказати, що стаття про моделювання безпроводових мереж у середовищі OMNeT++ з використанням INET Framework є актуальною та важливою, оскільки ця тема є досить розгалуженою та вимагає подальших досліджень. Останні дослідження вказують на те, що OMNeT++ разом з INET Framework є потужним інструментом для моделювання та дослідження без-

проводових мереж, що дозволяє вивчати різноманітні аспекти продуктивності та ефективності мережі.

**Метою статті** є дослідження аналіз одного з можливих підходів до проектування та безпроводових мереж у середовищі імітаційного моделювання OMNeT++ з використанням фреймворку INET.

Предметом дослідження є процес моделювання безпроводових мереж, різні режими їх роботи та методологію аналізу роботи моделі.

Об'єктом статті є методи та інструменти, які можна використовувати для моделювання та дослідження безпроводових мереж.

**Виклад основного матеріалу.** Розглянемо один із можливих підходів до проектування та аналізу безпроводових локальних мереж (Wireless Local Area Network — WLAN) з використанням середовища імітаційного моделювання OMNeT++. Поставлено завдання всебічного дослідження імітаційної моделі найпростішої мобільної WLAN шляхом її послідовного ускладнення для врахування особливостей різних режимів роботи безпроводової мережі. Як основу для розробки моделей пропонується використовувати типові компоненти фреймворку INET, зміна конфігурації яких дозволяє моделювати пристрої, що відповідають необхідним вимогам.

Об'єктом дослідження є WLAN, які повністю відповідають стандарту таких провідних мереж, як Ethernet [5], але використовують інше середовище передачі даних: інфрачервоне випромінювання або радіохвилі НВЧ-діапазону. Ці мережі можуть бути стаціонарними, мобільними та мобільно мандрівними. Стаціонарні вузли мережі жорстко прив'язані до певної точки простору. Мобільні допускають переміщення вузлів мережі в межах зони дії однієї точки доступу або одного сегмента мережі, а мобільні мандрівні допускають не тільки переміщення вузлів мережі, але забезпечують їх роумінг за рахунок автоматичного переключення від однієї точки доступу до іншої. За характером підключення WLAN можуть підтримувати два основні режими роботи:

- точка-точка (point-to-point), або режим Ad-Hoc [6, 7], при якому зв'язок між вузлами встановлюється безпосередньо без використання спеціальних точок доступу;
- точка-багатоточка (point-to-multipoint), або режим інфраструктури, при якому мережа складається як мінімум з однієї точки доступу, підключеної до проводової мережі та деякого набору безпроводових вузлів.

У цій статті основна увага приділена аналізу моделей мобільних Ad-Hoc мереж, що працюють як за прямої взаємодії вузлів, і при непрямій через проміжні вузли. Це дозволило дослідити роботу WLAN, що самоорганізуються, з динамічною маршрутизацією повідомлень до вузлів, що знаходяться поза зоною радіодоступу конкретного малопотужного трансівера вузла, використовуючи для цього проміжні вузли і істотно розширивши зону роботи конкретного сегменту WLAN [8, 9].

Для імітаційного моделювання такого роду мереж пропонується використовувати фреймворк INET, що входить у поставку OMNeT++. Він містить великий набір компонентів для моделювання як мережі в цілому, так і окремих її елементів, а саме: фізичного середовища та режимів розповсюдження сигналу в ній, різного типу антен, приймачів, передавачів та мережевих карток з можливістю обліку їх енергоспоживання [10].

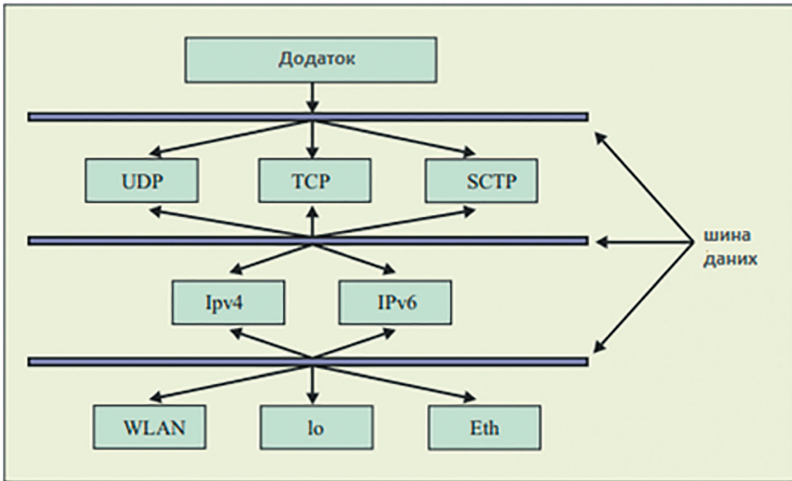
Першим етапом на шляху вирішення поставленого завдання стала розробка технології процесу побудови та дослідження найпростішої WLAN, що складається всього з двох вузлів, пов'язаних між собою радіоканалом для передачі UDP-дейтаграм. Для проектування імітаційної моделі такої мережі достатньо використовувати лише три компоненти фреймворку INET, а саме:

- складовий модуль WirelessHost;
- складовий модуль IdealRadioMedium;
- простий модуль IPv4NetworkConfigurator.

Складовий модуль WirelessHost представляє собою модель хоста безпроводової мережі і є одним з розширень модуля StandartHost, у якому базуються й інші моделі хостів TCP/IP. Внутрішня структура цього модуля (рис. 1) складається з чотирьох рівнів: прикладного, транспортного, мережевого та каналного, представленого мережевим інтерфейсом.

Кожен із рівнів включає до свого складу набір зумовлених компонентів із складу фреймворку INET, тип яких може бути змінений через файл конфігурації. Зокрема прикладний рівень містить безрозмірний масив компонентів типу «додаток», кожен з яких моделює роботу додатка на моделі хоста. Даний модуль може генерувати вихідний трафік, передаючи його на розташовані нижче рівні, так і приймати вхідний трафік. Транспортний рівень включає до свого складу три компоненти, які моделюють протокол користувацьких датаграм (User Datagram Protocol – UDP),

протокол управління передачею (Transmission Control Protocol – TCP) та протокол передачі з керуванням потоком (Stream Control Transmission Protocol – SCTP). У складі мережного рівня є два компоненти: інтернет-протокол четвертої версії (Internet Protocol version 4 – IPv4) та інтернет-протокол шостої версії (Internet Protocol version 6). Канальний рівень містить компоненти, що реалізують мережеві інтерфейси, такі як дротовий Ethernet (Eth), безпроводовий (wlan), внутрішній петльовий інтерфейс (Loopback – lo) та ін. Кожен із рівнів з'єднуються між собою через загальні шини даних.



**Рис. 1. Внутрішня структура хоста WirelessHost**

Для імітації фізичного середовища, в якому відбувається безпроводовий зв'язок, використовується складовий модуль IdealRadioMedium. Він відповідає за моделювання поширення сигналу, його ослаблення по мірі видалення, облік завад та інших фізичних явищ. Модель фізичного середовища визначає, коли, де і як передача та сторонній шум надходять на приймачі. Модуль IdealRadioMedium має складну структуру і включає до свого складу інші зумовлені компоненти, що моделюють наступні фізичні явища та процеси:

- розповсюдження радіосигналу в просторі (propagation) – компонент описує, як радіосигнал поширюється через простір у часі;

- аналогове уявлення радіосигналу (analogModel) – моделює процес, як аналогове уявлення передач перетворюється на аналогове уявлення прийомів;
- фоновий шум (backgroundNoise) – представляє собою модель фонового шуму та описує тепловий шум, космічний фоновий шум та інші випадкові коливання електромагнітного поля, що впливають на якість каналу зв'язку;
- зменшення потужності від відстані (pathLoss) – описує зменшення потужності по мірі поширення сигналу через простір;
- зменшення радіосигналу при проходженні через завади (obstacleLoss) – являє собою модель втрат і описує зменшення потужності сигналу при його проходженні через завади та фізичні об'єкти.

У загальному випадку реалізовані в INET моделі фізичного середовища описують фізичний пристрій, який здатний передавати та приймати сигнали. До їх складу входять: модель антени, моделі приймача та передавача, а також модель споживання енергії. Модель антени розподіляється між моделлю передавача та моделлю приймача. Поділ моделі передавача та моделі приймача дозволяє використовувати асиметричні конфігурації.

Розглянемо модель взаємодії двох безпроводових вузлів за протоколом UDP. Підхід до побудови цієї моделі буде розглянуто на прикладі безпроводової мережі, що складається з двох вузлів, що взаємодіють один з одним за протоколом UDP. У даній моделі вузли будуть розташовуватися на відстані 400 м один від одного, при цьому радіус дії приймача кожного з вузлів дорівнюватиме 500 м. Використання в якості протокола передачі даних UDP означає, що вузли повинні функціонувати не тільки на транспортному рівні, але й на рівні користувацьких додатків. Фреймворк INET містить два типи модулів, які працюють за протоколом UDP:

- UDPBasicApp – модуль, що забезпечує генерацію UDP-пакетів заданої довжини на задану IP-адресу із заданим часовим інтервалом;
- UDPSink – модуль, що імітує програму, яка виконує прийом повідомлень, що надійшли з рівня UDP, підрахунок їх кількості та припинення їх подальшої обробки.

У досліджуваній моделі необхідно, щоб вузол hostA генерував UDP-повідомлення довжиною 1 КБ і передавав його вузол hostB через випадковий проміжок часу. Використовувана модель без-

проводового вузла WirelessHost зі складу фреймворку INET вже має модуль інтерфейсу програми, тому для реалізації поставленого завдання необхідно лише перевизначити тип модуля, що використовується, у файлі конфігурації, який може мати наступний вигляд:

```
*.hostB.numUdpApps = 1
*.hostB.udpApp[0].typename = "UDPSink"
*.hostB.udpApp[0].localPort = 5000
*.hostA.numUdpApps = 1
*.hostA.udpApp[0].typename = "UDPBasicApp"
*.hostA.udpApp[0].destAdresses = "hostB"
*.hostA.udpApp[0].destPort = 5000
*.hostA.udpApp[0].messageLength = 1000B
*.hostA.udpApp[0].sendInterval = exponential(10ms)
```

Ця настройка повідомляє, що на вузлі hostA використовується модуль UDPBasicApp, який формує UDP-повідомлення довжиною 1 КБ і відправляє їх через випадковий інтервал часу, який описується експоненціальним розподілом із середнім значенням 10 мс. Для того, щоб сформовані повідомлення доходили до адресата, у файлі конфігурації вказується найменування вузла, якому призначене повідомлення, а також номер порту, наприклад, 5000. Вузол hostB використовує модуль UDPSink і також працює на порту 5000.

Крім рівня додатку, файл конфігурації повинен містити визначення фізичного рівня, який представляється в моделі хоста WirelessHost як мережевий адаптер, до складу якого входить антена та радіотрансмівер. У середовищі INET існує велика різноманітність радіомодулів, що підтримують різні протоколи фізичного рівня, проте в даному прикладі використовуватиметься ідеалізована модель фізичного середовища, хости якого містять модуль IdealRadio як частину IdealWirelessNic. На підставі викладеного, конфігурація фізичного рівня матиме вигляд:

```
*.host*.wlan[*].typename = "IdealWirelessNic"
*.host*.wlan[*].radio.transmitter.CommunicationRange = 500m
*.host*.wlan[*].radio.receiver.ignoreInterference = true
* *.bitrate = 1Mbps
```

Після запуску створеної імітаційної моделі на виконання відкриється графічне вікно середовища виконання, де буде представлена вся мережа та модулі, що входять до її складу. Середовище дозволяє в будь-який момент відобразити внутрішню

структуру складеного модуля двічі клацнувши по його піктограмі. Треба звернути увагу, що структура модуля, що відобразилася, буде відрізнятися від тієї, що була представлена на рис. 1. Це пов'язано з тим, що після запуску моделі виконання середовище імітаційного моделювання обробила файл конфігурації та налаштувала параметри компонентів WirelessHost.

Кожен хост має лише по одному мережному адаптеру, але з двома мережевими інтерфейсами кожен. При цьому один із них є внутрішнім петлевим (lo), а другий є зовнішнім (wlan) і підключеним до бездротової мережі через внутрішній трансівер. Обидва вузли мають однакову структуру як на фізичному рівні, так і на канальному, мережевому і транспортному рівнях. Відмінність спостерігається лише на рівні додатків, а саме, у типі використаного додатка.

Запустивши модель на виконання, можна спостерігати, як програма UDPBasicApp, що працює на вузлі hostA, формує UDP-пакети з випадковою часовою шпаруватістю. Після чого ці пакети, пройшовши через рівні UDP і IPv4, надходять в мережевий інтерфейс wlan, який ставить пакети, що надійшли, в чергу і здійснює їх передачу, як тільки виникає така можливість. Використання стекової реалізації в мережному інтерфейсі дозволяє узгодити швидкість надходження пакетів з верхнього рівня з пропускною здатністю каналу, а також швидкістю роботи середовища. Це означає, що поки в стіковій черзі є пакети, вони передаватимуться один за одним без пробілів між ними.

За результатами імітаційного моделювання видно, що UDP-додаток на вузлі hostA за 25 с сформував 2427 пакетів. При цьому 2426 пакетів було опрацьовано на транспортному рівні UDP. З них 2422 пакети було передано по радіоканалу і надійшло на вузол hostB. Тут вони пройшли через канальний, мережевий та транспортний рівні цього вузла і надійшли в UDP-додаток хоста hostB. Таким чином, лічильник кількості пакетів вказує, що за 25 с було прийнято 2422 пакети довжиною 1028 Б (1000 Б повідомлення + 8 Б UDP + 20 Б IP) кожен, а це означає, що швидкість передачі склала близько 800 кбіт/с.

Розглянемо модель статичної маршрутизації для взаємодії віддалених вузлів. Можливості застосування імітаційного середовища OMNet++ не обмежуються процесом проектування та дослідження лише простих моделей. Вона дозволяє також досліджувати складніші моделі та ситуації, такі як у роботі [11], де



на прикладі створеної імітаційної моделі проводової комп'ютерної мережі було проаналізовано ефективність використання резервування передач. Інший приклад, який буде розглянуто нижче, демонструє підхід до проектування та дослідження складнішої структури, але вже безпроводової мережі [12, 13]. Припустимо, що є необхідність забезпечення безпроводового зв'язку та передачі UDP-повідомлень від вузла hostA до вузла hostB з радіусом дії їх приймачів всього 250 м. При цьому відстань між цими вузлами становить 400 м, що виключає можливість їх прямої взаємодії. Додатково між вузлами розташовані ще три вузли hostR1, hostR2 та hostR3, які в процесі своєї роботи можуть надавати інтерференційні завади на роботу сусідніх вузлів. Особливість досліджуваної моделі полягає в тому, що приймачі мережних адаптерів, володіючи малою потужністю, обмежують радіус їхньої дії, що призводить до неможливості прямого зв'язку між двома вузлами. Однак з'єднання може бути встановлене, якщо між цими вузлами будуть розташовуватися інші вузли, які можуть транслювати та передавати мережні пакети. Для цього всі або частина проміжних вузлів мають підтримувати маршрутизацію.

Якщо до вихідної моделі (рис. 3) додати ще три вузли, обмежити потужність всіх приймачів до 250 м і запустити модель на виконання, то можна побачити, що вузол hostA відправляє UDP-пакети на сусідні вузли, проте останні не приймають їх, оскільки ці пакети їм не призначені. Для того, щоб сусідні вузли не відкидали отримані пакети, а передавали їх на інший вузол, необхідно, щоб на проміжних вузлах було налаштовано таблицю маршрутизації [14]. У цьому найпростішому випадку розглянуто приклад статичної маршрутизації, яка налаштовується через простий модуль IPv4NetworkConfiguration у файлі конфігурації, який має такий вигляд:

```
#Автоматичне налаштування статичних маршрутів
*.configurator.config = xml("<config>
<interface host='**' address='10.0.0.x' netmask='255.255.255.0' />
<autoroute metric='errorRate' />
</config>")
#Вимкнення оптимізації записів таблиці маршрутизації
*.configurator.optimizeRoutes = false
#Вимкнення записів таблиці маршрутизації, створених з мережевої маски
**.RoutingTable.netmaskRoutes = ""
```

Ця установка здійснюється за допомогою рядка XML, параметри якого повідомляють конфігуратору, щоб він призначав IP-адреси в діапазоні 10.0.0.x і використовував оцінну частоту помилок передач між вузлами мережі для налаштування статичних маршрутів. Таким чином, маршрути будуть формуватися так, щоб мінімізувати сумарні помилки, що призводить до створення правильно налаштованої IPv4 мережі без будь-яких додаткових ручних налаштувань. Сформовані таблиці маршрутизації зберігаються у параметрі routingTable кожного з хостів, яка доступна для перегляду у графічному середовищі виконання (рис. 2).

```

routes (std:vector<inet:IPv4Route *>)
  routes[4] (inet:IPv4Route *)
    [0] = dest:10.0.0.2 gw:10.0.0.3 mask:255.255.255.255 metric:0 if:wlan0(10.0.0.1) REMOTE MANUAL
    [1] = dest:10.0.0.3 gw:* mask:255.255.255.255 metric:0 if:wlan0(10.0.0.1) DIRECT MANUAL
    [2] = dest:10.0.0.4 gw:* mask:255.255.255.255 metric:0 if:wlan0(10.0.0.1) DIRECT MANUAL
    [3] = dest:10.0.0.5 gw:10.0.0.3 mask:255.255.255.255 metric:0 if:wlan0(10.0.0.1) REMOTE MANUAL
  
```

**Рис. 2. Маршрутизація вузла hostA:**

**dest** – адреса призначення, **gw** – шлюз; **mask** – маска підмережі; **metric** – метрика мережі, **REMOTE MANUAL** – віддалений доступ; **DIRECT MANUAL** – прямий доступ

З наведеної таблиці видно, що вузол hostA (10.0.0.1) має прямий інтерфейс зв'язку з вузлами hostR1 (10.0.0.3) та hostR2 (10.0.0.4). Також у таблиці є маршрут, який інформує, що до вузла hostB (10.0.0.2) можна дістатися через вузол hostR1, використовуючи останній як шлюз, а вузол hostR3 (10.0.0.5) може бути доступний через шлюз hostR2. Закінчивши конфігурування моделі мережі та запустивши її на виконання, можна в динаміці побачити, як згенерований вузлом hostA пакет UDP надходить на вхід вузла hostR1, проте тепер він не відкидає цей пакет, а приймає його та здійснює його пересилання іншому вузлу.

Цей процес відбувається відповідно до налаштувань статичної маршрутизації, при якій проміжні вузли даного сегмента WLAN відіграють роль шлюзів, збільшуючи практично вдвічі діапазон можливої безпроводової взаємодії вузлів hostA і hostB. Основна відмінність цієї моделі полягає в тому, що вона моделює процес, при якому сформований вузлом hostA пакет (UDPBasicAppData-0) надходить у радіоканал не безпосередньо до вузла hostB, а проходячи через мережевий та канальний рівні вузла hostA, адресується

ся тепер вузлу hostR1. Цей процес виконується на базі, що зберігається в пам'яті вузла hostA, таблиці маршрутизації, на основі якої всі пакети, призначені для доставки на вузол hostB, адресуються для передачі на вхід вузла hostR1. Пакети, що надійшли на його вхід, будуть підніматися до мережного рівня і відразу відправлятися зворотно в мережу відповідно до таблиці маршрутизації, що зберігається в пам'яті вузла hostR1. Так як вузол hostR1 має прямий маршрут доступу до вузла hostB, то UDP-пакет, що надійшов на його вхід, буде безпосередньо трансльований вузлу hostB.

Розглянемо модель обліку взаємних завад. У попередній моделі WLAN були реалізовані ідеалізовані умови роботи мережі, в якій не враховувалося таке фізичне явище як інтерференція радіохвиль, що виникає при вступі на вхід радіоприймача двох і більше радіосигналів, через що вони стикаються, спотворюються та збивають нормальну роботу приймача радіосигналу [1]. До цього моменту цей ефект не враховувався, і по суті моделювалися пристрої з повним дуплексним зв'язком [15]. При цьому в середовищі OMNeT++ в якості базових реалізовано чотири способи опису та подання сигналів.

Перше уявлення називається range-based (заснований на діапазонах). Саме воно реалізовано у компоненті IdealRadioMedium. Перевагою цієї структури є компактність, передбачуваність та висока продуктивність. Однак її недолік полягає в тому, що вона не точно відбиває реальну поведінку фізичного середовища.

Друга структура є вузькосмуговим сигналом зі скалярною потужністю сигналу, що несе частотою і смугою пропускання. Його перевага в тому, що дозволяє обчислювати ставлення сигналу/шуму, а також те, що в більшості випадків його достатньо для моделювання мереж стандарту IEEE 802.11.

Третя структура даних визначає сигнал, потужність якого змінюється пізніше. У цьому випадку потужність сигналу представлена однимірним годинниковим значенням, яке точно слідує за переданими імпульсами. Дане подання використовується при моделюванні радіохвиль стандарту IEEE 802.15.4a UWB.

Четверте уявлення використовує багатовимірні значення для опису потужності сигналу, що змінюється як за годинами, так і частотою. Дане уявлення може бути використане при моделюванні радіохвиль стандарту IEEE 802.11b.

У всіх розглянутих прикладах використовувалася найпростіша модель фізичного середовища на основі модуля

IdealRadioMedium, який використовує діапазонне уявлення, де ступінь впливу сигналу на сусідні вузли залежить від відстані, на якому вони знаходяться. Працюючи з цим модулем та її конфігурацією виділяються лише три основні діапазони.

1. Communication range – діапазон впевненого прийому та передачі радіосигналу;

2. Interference Range – діапазон, у якому нормальний зв'язок вже неможливий, але він ще суттєво впливає на приймачі інших пристроїв;

3. Detection range – діапазон, де немає впливу вузла на приймачі інших пристроїв, але є можливість виявити факт наявності та роботи цього пристрою.

Для дослідження ступеня взаємного впливу радіохвиль у файлі конфігурації компонент IdealRadioMedium був налаштований на інтерференційний діапазон з відстанню 500 м. Присвоєні параметри описують той факт, що радіосигнали стають слабше з відстанню, але існує діапазон, в якому вони більше не можуть бути правильно прийняті, але вони все ще досить сильні, щоб впливати на інші сигнали, що призводить до збою прийому.

```
#Допускаємо облік завад у радіоприймачі від роботи сусідніх вузлів *.host*.wlan[*].radio.receiver.ignoreInterference = false
```

```
#Робимо інтерференційний діапазон, що дорівнює подвійному діапазону зв'язку *.host*.wlan[*].radio.transmitter.maxInterferenceRange = 500m
```

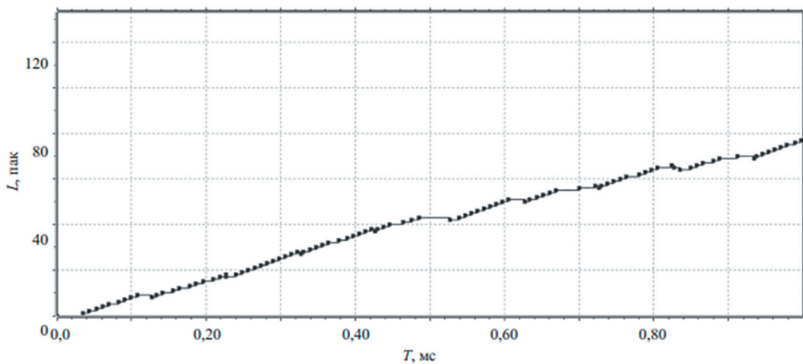
```
#Відображення на схемі інтерференційного діапазону вузла hostA*.hostA.wlan[0].radio.displayInterferenceRange=true
```

Для дослідження цього режиму файл конфігурації був включений режим реєстрації та запису подій, які відбуваються у моделі за годину її прогону. Після закінчення роботи моделі було сформовано файли звітів, згідно з якими за час роботи моделі:

- вузол hostA сформував 92 UDP-повідомлення розміром 1 КБ кожний;
- рівень UDP вузла hostA пройшла 91 дейтаграма розміром 1008 Б;
- до мережі, пройшовши рівень MAC, відправлено 89 пакетів розміром 1028 Б;
- проте на вузол hostB надійшов лише 1 пакет за 1 з модельного часу.

Якщо тепер проаналізувати тимчасову діаграму зареєстрованого процесу моделювання мережі, то можна виявити, що тільки

на 800 мс модельного часу на вхід wlanB.hostB.udpApp(0) надійшло одне з перших повідомлень, яке було згенеровано вузлом hostA і ретрансльовано вузлом hostR1. Висока інтенсивність (середня година на 10 мс) випадкової генерації UDP-пакетів вузлом hostA, наявність інтерференційних завад та боротьба за доступ до безпроводового середовища передачі даних, призвели до того, що відправлення повідомлення з вузла hostA на вузол hostB було здійснено лише в момент значної паузи на вузлі hostA. При цьому боротьба за доступ до безпроводового середовища передачі даних велася між вузлами hostA і hostR1, які хотіли передати вузлу hostB як прямі, так і ретрансльовані UDP-пакети. Але якщо на вузол hostB надійшло одне з перших, відправлених із вузла hostA повідомлень, це говорити про те, що воно в мережі не пропало і змогло десь зберегтися. На цей питання дає відповідь графік залежності від часу (T) довжини черги (L), що формується в буфері адаптера мережевого шлюзу hostR1, який ретрансльює пакети між вузлами hostA та hostB. З наведеного на рис. 3 графіка видно, що за 1 с модельного часу в цьому буфері збереглося майже 80 відправлених вузлом hostA пакетів, які вузол hostR1 не встиг ретрансльувати в hostB. На даний момент значення максимальної довжини черги буфера адаптера визначено за замовчуванням у відповідному модулі фреймворку INET. У реальній безпроводовій мережі довжина буфера або його об'єм мають велике значення для роботи мережі, особливо якщо цей адаптер використовується у такому пристрої, як шлюз.



**Рис. 3. Графік довжини черги у буфері мережевого шлюзу вузла hostR1**

Причини які заважали в моделі UDP-паketу, що моделюється, з буфера шлюзу hostR1 дійти до вузла hostB раніше? Відповідь це питання дає аналіз зареєстрованих у мережі подій, які відбувалися у процесі її роботи [16 – 19]. Розглянуто часову діаграму роботи всіх елементів для всіх вузлів безпроводової мережі, звідки видно, що:

- перший пакет (UDPData-0) надходить у мережу (подія #22) і через деякий інтервал часу досягає вузлів hostR1 (#23) і hostB (#26). Причому друга подія відбувається пізніше, оскільки вузол hostB розташований далі;
- вузол hostB приймає радіосигнали від вузла hostA (#26 - #45), але не розпізнає їх через значну віддаленість від джерела передачі;
- вузол hostR1 починає успішний прийом пакета від вузла hostA (# 23 - # 32);
- в процесі передачі першого пакета, додаток wlanB.hostA.udpApp(0) (#27) генерує друге повідомлення UDPBasicAppData-1, яке не може бути відправлено в мережу. Воно поміщається в буфер мережного адаптера і буде там до кінця роботи мережного адаптера з першим пакетом. Цей момент описує подію #36 і зелена пунктирна стрілка, що пов'язує його з подією #27;
- вузол hostR1, успішно закінчивши прийом першого пакета (#32), звертається до своєї таблиці маршрутизації (#35). На її основі він переадресує прийнятий пакет на вузол hostB і відправляє його до мережі (#39);
- проте трохи раніше (#32) вузол hostA почав передачу другого пакета, який раніше зберігався в буфері його мережевого адаптера;
- на вхід приймача вузла надходять одночасно два радіосигнали (#48 і #51), які інтерферують між собою, не дозволяючи вузлу hostB розпізнати UDP-пакет, що йде від шлюзу.

Підсумовуючи дослідження моделі WLAN з урахуванням інтерференційних завад від роботи сусідніх пристроїв, слід зазначити, що спостерігається суттєве зниження продуктивності роботи. Більшу частину часу передавальний вузол і шлюз працюють одночасно, що викликає прихід на трансивер приймаючого вузла відразу двох сигналів та їхнє «зіткнення» між собою.

Такий підхід до організації безпроводових мереж є неприйнятним, і щоб звести до мінімальної завади, потрібен деякий протокол доступу до мультимедіа середовища, що дозволить визначати, який хост і коли має право на передачу даних [17, 18]. Одним із таких протоколів може бути працюючий на рівні MAC протокол на основі CSMA/CA з додатковими підтвердженнями та механізмом повтору, якою в середовищі OMNeT++ реалізує модуль CsmCaMac, який при відповідних налаштуваннях може успішно апроксимувати базовий режим 802.11b Ad-Hoc [20 – 22].

**Висновки та пропозиції.** Наведені дослідження дозволяють продемонструвати, як імітаційне моделювання використовується при проектуванні та вивченні роботи реальних безпроводових мереж. Показано можливість створення імітаційних моделей безпроводових мереж з використанням готових компонентів зі складу фреймворку INET, які допомагають описувати різні елементи мережі, а також виконувати їх гнучке налаштування для моделювання необхідної поведінки.

Продемонстровано підхід моделювання різних режимів роботи безпроводових мереж на прикладі реалізації прямої взаємодії при проектуванні режиму Ad-Hoc, а також непрямого, за якого реалізується процес маршрутизації через проміжні вузли. Дослідження у статті такого фізичного явища, як інтерференція, дозволив виявити факт негативного впливу один на одного розташованих вузлів, які погіршують якість радіозв'язку і можуть призводити до зниження швидкості передачі даних або їх повної втрати.

У статті також продемонстровано підхід дослідження роботи безпроводової мережі шляхом аналізу файлу часової діаграми, вивчення якого дозволило пояснити причину суттєвих затримок у ретрансляції мережних пакетів.

**© Лемешко А.В., Антоненко А.В., Костецький Д.І., Шрам М.М., Закреничний А.С., 2023**

## ЛІТЕРАТУРА

1. Varga, A. The OMNeT++ discrete event simulation system. In Proceedings of the European Simulation Multiconference. Prague. Czech Republic. 2001.
2. Sommer, C., Dressler, F. and Gansen, T. (2008) On the need for bidirectional coupling of road traffic microsimulation and network simulation.

In Proceedings of the 11th Communications and Networking Simulation Symposium (CNS), Ottawa, ON, Canada. 2008. P. 71 – 79.

3. Hämmäinen, H., Mäkelä, J., Mahonen, P. and Niemi, V. (2008) OMNeT++ network simulation framework. In Proceedings of the 5th ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), Vancouver, BC, Canada. 2002. P. 171 – 178.

4. INET Framework User's Guide, <https://inet.omnetpp.org/docs/users-guide/>.

5. Bhattacharjee, A., Rahmani, A.M. and Salim, U.A. (2015) Modeling and simulation of wireless sensor networks using OMNeT++ and MiXiM framework. In Proceedings of the 2015 IEEE International Conference on Computer, Communication and Control (IC4), Indore, India. 2015. P. 1 – 6.

6. Castañeda, L.E., Moya, F., Casilari, E. and Lloret, J. (2016) A comparative study of network simulators for wireless sensor networks. In Proceedings of the International Conference on Wireless Networks and Mobile Communications (WINCOM), Fez, Morocco. 2016. P. 1 – 6.

7. Твердохліб А.О., Коротін Д.С. Ефективність функціонування комп'ютерних систем при використанні технології блокчейн і баз даних // *Таврійський науковий вісник*. Серія: Технічні науки. 2022. (6).

8. Цвик О.С. Аналіз і особливості програмного забезпечення для контролю трафіку // *Вісник Хмельницького національного університету*. Серія: Технічні науки. 2023. (1).

9. Новіченко Є.О. Актуальні засади створення алгоритмів обробки інформації для логістичних центрів // *Таврійський науковий вісник*. Серія: Технічні науки. 2023. (1).

10. Зайцев Є.О. Smart засоби визначення аварійних станів у розподільних електричних мережах міст // *Таврійський науковий вісник*. Серія: Технічні науки. 2022. (5).

11. Li, F., Li, X., Li, B. and Li, Q. (2017) A comprehensive survey of network simulators for wireless networks. *Journal of Network and Computer Applications*. 88. P. 18 – 44.

12. Singh, S., Purohit, P. and Kothari, A. (2017) Comparative analysis of wireless sensor network simulators: A survey. In Proceedings of the International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST), Jaipur, India, 2017. P. 1 – 5.

13. Ali, M., Khan, M., Memon, Q. and Kumar, D. (2019) Performance comparison of network simulators for wireless sensor networks. In Proceedings of the 3rd International Conference on Advanced Computational and Communication Paradigms (ICACCP), Sikkim, India. 2019. P. 1 – 6.



14. Khajehpour, H. and Roudsari, M.H. (2019) A survey on network simulators for wireless sensor networks // *Journal of Sensor and Actuator Networks*. 8(1). P. 2.

15. Koucheryavy, A. Quality of Service (QoS) classes for Ubiquitous Sensor Networks / A. Koucheryavy, A. Prokopiev // ICACT'2009: Proceedings, 15–18 February, Phoenix Park, Korea, 2009. P. 107 – 109.

16. Kolomoitcev V.S., Bogatyrev V.A. The fault-tolerant structure of multilevel secure access to the resources of the public network // *Communications in Computer and Information Science*. 2016. V. 678. P. 302 – 313.

17. Bogatyrev V.A., Slastikhin I.A. The models of the redundant transmission through the aggregated channels // *ACSR-Advances in Computer Science Research*. 2017. V. 72. P. 294 – 299.

18. Bogatyrev S.V., Bogatyrev V.A. Analysis of the Timeliness of Redundant Service in the System of the Parallel-Series Connection of Nodes with Unlimited Queues // *Proc. 2018 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF)*. 2018. P. 8604379.

19. Семенчук, О., Татарчук, Т., & Чебанова, Н. (2016). Моделювання технології LTE в системах масового обслуговування. Наукові праці Національного університету «Львівська політехніка».

20. Яворський, Б., Дубинський, Ю., & Дорошко, С. (2016). Моделювання технології Wi-Fi у середовищі мережевого симулятора OMNeT++. Збірник наукових праць НТУ «ХПІ».

21. Кулікова, О., Рева, О., & Яковлева, І. (2018). Моделювання безпроводових мереж на основі WSN у середовищі OMNeT++. Наукові праці ДонНТУ. Серія «Інформатика, кібернетика та обчислювальна техніка».

22. Васильченко, Д. В., & Перетягін, О. В. (2019). Моделювання технології NB-IoT у середовищі OMNeT++. Наукові праці Херсонського державного університету. Серія: Електроніка і телекомунікації.

## REFERENCES

1. Varga, A. The OMNeT++ discrete event simulation system. In *Proceedings of the European Simulation Multiconference, Prague, Czech Republic, 2001* [in English].

2. Sommer, C., Dressler, F. and Gansen, T. (2008) On the need for bidirectional coupling of road traffic microsimulation and network simulation. In *Proceedings of the 11th Communications and Networking Simulation Symposium (CNS), Ottawa, ON, Canada, 2008*, P. 71 – 79 [in English].

3. Hämmäinen, H., Mäkelä, J., Mahonen, P. and Niemi, V. (2008) OMNeT++ network simulation framework. In Proceedings of the 5th ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems (MSWiM), Vancouver, BC, Canada. 2002. P. 171 – 178 [in English].

4. INET Framework User's Guide, <https://inet.omnetpp.org/docs/users-guide/> [in English].

5. Bhattacharjee, A., Rahmani, A.M. and Salim, U.A. (2015) Modeling and simulation of wireless sensor networks using OMNeT++ and MiXiM framework. In Proceedings of the 2015 IEEE International Conference on Computer, Communication and Control (IC4), Indore, India. 2015. P. 1 – 6 [in English].

6. Castañeda, L.E., Moya, F., Casilari, E. and Lloret, J. (2016) A comparative study of network simulators for wireless sensor networks. In Proceedings of the International Conference on Wireless Networks and Mobile Communications (WINCOM), Fez, Morocco. 2016. P. 1 – 6 [in English].

7. Tverdokhlib A.O., Korotin D.S. Efektyvnist funkcionuvannia kompiuternykh system pry vykorystanni tekhnolohii blokchein i baz dannykh. Tavriiskyi naukovi visnyk. Serii: Tekhnichni nauky, 2022, (6) [in Ukrainian].

8. Tsvyk O.S. Analiz i osoblyvosti prohramnoho zabezpechennia dlia kontroliu trafiku. Visnyk Khmelnytskoho natsionalnoho universytetu. Serii: Tekhnichni nauky, 2023. (1) [in Ukrainian].

9. Novichenko Ye.O. Aktualni zasady stvorennia alhorytmiv obrobky informatsii dlia lohistychnykh tsentriv. Tavriiskyi naukovi visnyk. Serii: Tekhnichni nauky, 2023 (1) [in Ukrainian].

10. Zaitsev Ye.O. Smart zasoby vyznachennia avariinykh staniv u rozpodilnykh elektrychnykh merezhakh mist. Tavriiskyi naukovi visnyk. Serii: Tekhnichni nauky, 2022. (5) [in Ukrainian].

11. Li, F., Li, X., Li, B. and Li, Q. (2017) A comprehensive survey of network simulators for wireless networks // *Journal of Network and Computer Applications*. 88. P. 18 – 44 [in English].

12. Singh, S., Purohit, P. and Kothari, A. (2017) Comparative analysis of wireless sensor network simulators: A survey. In Proceedings of the International Conference on Emerging Trends in Engineering, Science and Technology (ICETEST), Jaipur, India. 2017. P. 1 – 5 [in English].

13. Ali, M., Khan, M., Memon, Q. and Kumar, D. (2019) Performance comparison of network simulators for wireless sensor networks. In Proceedings of the 3rd International Conference on Advanced Computational and Communication Paradigms (ICACCP), Sikkim, India. 2019. P. 1 – 6 [in English].

14. Khajehpour, H. and Roudsari, M.H. (2019) A survey on network simulators for wireless sensor networks. *Journal of Sensor and Actuator Networks*. 8(1). P. 2 [in English].

15. Koucheryavy, A. Quality of Service (QoS) classes for Ubiquitous Sensor Networks / A. Koucheryavy, A. Prokopiev // ICACT'2009: Proceedings, 15–18 February, Phoenix Park, Korea. 2009. – P. 107 – 109 [in English].

16. Kolomoitcev V.S., Bogatyrev V.A. The fault-tolerant structure of multilevel secure access to the resources of the public network // Communications in Computer and Information Science. 2016. V. 678. P. 302 – 313. [in English].

17. Bogatyrev V.A., Slastikhin I.A. The models of the redundant transmission through the aggregated channels // ACSR-Advances in Computer Science Research. 2017. V. 72. P. 294 – 299. doi: 10.2991/itsmssm-17.2017.60 [in English].

18. Bogatyrev S.V., Bogatyrev V.A. Analysis of the Timeliness of Redundant Service in the System of the Parallel-Series Connection of Nodes with Unlimited Queues // Proc. 2018 Wave Electronics and its Application in Information and Telecommunication Systems (WECONF). 2018. P. 8604379. [in English].

19. Semenchuk, O., Tatarchuk, T., & Chebanova, N. (2016). Modeling LTE technology in mass service systems. Scientific works of the Lviv Polytechnic National University [in Ukrainian].

20. Yavorsky, B., Dubinsky, Yu., & Doroshko, S. (2016). Modeling Wi-Fi technology in the OMNeT++ network simulator environment. Collection of scientific works of NTU «KhPI» [in Ukrainian].

21. Kulikova, O., Reva, O., & Yakovleva, I. (2018). Simulation of WSN-based wireless networks in the OMNeT++ environment. Scientific works of DonNTU. Series «Informatics, Cybernetics and Computer Engineering» [in Ukrainian].

22. Vasylychenko, D. V., & Peretyagin, O. V. (2019). Simulation of NB-IoT technology in the OMNeT++ environment. Scientific works of Kherson State University. Series: Electronics and telecommunications [in Ukrainian].

**СТАТТЯ НАДІЙШЛА ДО РЕДАКЦІЇ 12.05.2023**

УДК 004.02, 004.455.2

DOI: <https://doi.org/10.53920/ITS-2023-1-4>

**Олександр Андрійович ТКАЧЕНКО,**

кандидат фізико-математичних наук, доцент,  
доцент кафедри інформаційних технологій,  
Державний університет інфраструктури та технологій  
ORCID ID: 0000-0001-6911-2770

**Ольга Іванівна ТКАЧЕНКО,**

кандидат фізико-математичних наук, доцент,  
доцент кафедри інформаційних технологій,  
Державний університет інфраструктури та технологій  
ORCID ID: 0000-0003-1800-618X

**Олександр Олександрович ДЕЛАНТ,**

магістрант кафедри інформаційних технологій,  
Державний університет інфраструктури та технологій  
ORCID ID: 0009-0006-9785-5006

## **СИСТЕМА ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ ЩОДО ВИБОРУ ВЕЛОСИПЕДУ**

***В наш час системи підтримки прийняття рішень використовуються в багатьох сферах економіки, науки, освіти, в нашому повсякденному житті. Задачею таких систем є генерація управлінських (керуючих) рішень на основі аналізу великих обсягів інформації щодо обраної предметної області. При виборі та подальшої купівлі велосипедів в сучасному Інтернет-магазині проблема полягає у тому, що зараз користувач спирається в основному на параметр зросту і власні побажання щодо марки/кольору/типу. Але з часом з'ясується, що вибраний велосипед не відповідає повністю або частково потребам користувача. Тому актуальність проблеми розробки системи підтримки прийняття рішень щодо вибору велосипеду не викликає сумнівів.***

***Метою роботи є аналіз та дослідження проблем щодо розробки програмного забезпечення відповідної системи прийняття рішень, яка б підтримувала вибір користувачами транспортного засобу – велосипеду – для більш точного підбору оптимального велосипеду з урахуванням потреб користувача (наприклад, стилю їзди, покриття, пори року, частоти використання велосипеду), які визначаються шляхом відповідного опитування бажаючих придбати велосипед.***

**Використання розробленої авторської системи підтримки прийняття рішень вибору велосипеда дозволить, зокрема: надавати користувачеві зручну можливість при визначенні щодо його вибору і покупки велосипеда; створити простий зв'язок між користувачем та системою за рахунок використання графічного інтерфейсу; користувачеві не потрібно знатись на технічних аспектах та характеристиках велосипеда, через те що запитання ставляться у зрозумілому форматі без використання термінології та технічних назв, у той час як відповіді є однозначним висвітленням потреби користувача; згенерувати декілька варіантів рішень вибору велосипедів, що відповідають потребам (пожаданням) користувача згідно з розрахунком відповідності характеристик різних марок велосипедів потребам (пожаданням) користувача.**

**Ключові слова:** системи підтримки прийняття рішень, вебсистема, методи вибору рішень, графічний інтерфейс, велосипед, оцінювання потреб, аналіз даних опитування.

**Olexandr TKACHENKO**

PhD of physical and mathematical sciences, associate professor,  
associate professor at the department of information  
technologies,  
State University of Infrastructure and Technology

**Olha TKACHENKO**

PhD of physical and mathematical sciences, associate professor,  
associate professor at the department of information  
technologies,  
State University of Infrastructure and Technology

**Oleksandr DELANT**

Undergraduate at the department of information technologies,  
State University of Infrastructure and Technology

**DECISION SUPPORT SYSTEM REGARDING  
THE CHOICE OF A BICYCLE**

**Nowadays, decision support systems are used in many areas of economy, science, education, and in our everyday life. The task of such systems is the generation of management (management) decisions based on the analysis of large volumes of information regarding the selected subject area. When choosing and subsequently buying bicycles**

***in a modern online store, the problem is that now the user relies mainly on the height parameter and his own wishes regarding the brand/color/type. But over time, it turns out that the chosen bike does not fully or partially meet the needs of the user. Therefore, the relevance of the problem of developing the decision support system for choosing a bicycle is beyond doubt.***

***The aim of the work is the analysis and research of problems related to the development of software for a suitable decision-making system that would support the choice of a vehicle – the bicycle – by users for a more accurate selection of the optimal bicycle, taking into account the needs of the user (for example, riding style, coverage, seasons, frequency of bicycle use ), which are determined by a corresponding survey of those wishing to purchase a bicycle.***

***The use of the author's developed decision support system for choosing a bicycle will allow, in particular: to provide the user with a convenient opportunity when determining his choice and purchase of a bicycle; create a simple connection between the user and the system through the use of a graphical interface; the user does not need to know the technical aspects and characteristics of the bicycle, due to the fact that the questions are asked in an understandable format without the use of terminology and technical names, while the answers are an unambiguous reflection of the user's needs; generate several options for choosing bicycles that meet the needs (wishes) of the user according to the calculation of the compatibility of the characteristics of different brands of bicycles with the needs (wishes) of the user.***

***Keywords:*** decision support systems, decision selection methods, websystem, graphical interface, bicycle, needs assessment, survey data analysis, user system.

**Постановка проблеми.** Велосипедний транспорт є зручним засобом переміщення і використання вільного часу. Через невелику ціну та легкість керування його часто обирають й для переміщення у місті. Через це перед людьми, які не знаються на цій техніці, постає проблема, як та який велосипед обрати. Доступність Інтернету обумовлює і легкість, і складність здійснення такого вибору. Легкість забезпечується за рахунок отримання інформації щодо характеристик і параметрів усіх сучасних велосипедів, але при такому підході користувачі до вибору велосипеду втратили можливість одразу ж протестувати їх на собі, бо, зазвичай, підбір

на сайтах відбувається від всього одного параметру – зросту користувача.

В наш час спостерігається стрімкий розвиток нових технологій, які полегшують життя різним категоріям людей. Наприклад, нейромережі можуть використовуватись при вирішенні багатьох проблем, пов'язаних із транспортом: від так званого «вирізнання» об'єктів з кадрів відеоспостереження за дорожнім рухом до повноцінного керування транспортним засобом.

Системи підтримки прийняття рішень (СППР), які використовуються в багатьох сферах економіки, науки, освіти, медицини, все більше використовуються в нашому повсякденному житті [1, 2]. Адже задачею СППР є генерація управлінських (керуючих) рішень на основі аналізу великих обсягів інформації.

В наш час, коли більшість товарів купується через Інтернет та доставляється до користувача, втрачається можливість відразу наживо побачити замовлений товар, і, відповідно, неможливо одразу оцінити зручність та відповідність товару нашим вимогам.

При виборі велосипедів проблема полягає у тому, що користувач при своєму рішенні спирається на параметр зросту і власні побажання щодо марки/кольору/типу. Але при такому підході він з часом розуміє, що вибраний велосипед не відповідає його вимогам (повністю або частково). У деяких випадках куплений велосипед взагалі може бути незручний через те, що користувач може мати інші від стандартних параметри та потреби щодо комфорту.

Обираючи велосипед за картинками, що надаються Інтернет магазином, користувач не має уяви, наскільки зручним та практичним у використанні буде для нас той чи інший велосипед, адже більшість з користувачів не знається на технічних характеристиках таких транспортних засобів.

Вибираючи велосипед у звичайному магазині, часто користувач не має можливості протестувати велосипед перед покупкою, а продавці не завжди мають достатньо знань і розумінь клієнта (потенційного покупця), щоб порекомендувати, порадити їм підібрати потрібний. Тому визначена проблема є актуальною, а для її вирішення необхідно розробити програмний продукт, який отримуватиме в якості вхідних даних відповіді та побажання потенційного клієнта (покупця) Інтернет-магазину, видаючи в результаті перелік велосипедів, які задовільнять вимогам користувача такого програмного продукту.

**Аналіз останніх досліджень і публікацій.** В сфері електронної комерції проблеми вибору товару розглядалися, зокрема в [8]. Покупці, приймають рішення щодо покупки, зважаючи альтернативи або, у більшості випадків, базуючись на ціні товару.

Конкретний приклад використання СППР при формуванні рішення щодо покупки товару розглянуто в [9]. Авторами головна увага при виборі товару приділялася визначенню впевненості у покупці та рівня задоволеності товаром шляхом порівняння статистичного аналізу серед покупців, що обирали товар самостійно, таких, що користувалися при своєму виборі рекомендаціями СППР, та таких, що обирали тільки той товар, який їм обрала відповідна СППР.

При аналізі систем підтримки прийняття рішень при купівлі велосипеда та іншої техніки було виявлено що вони найчастіше надаються компаніям на платній основі. В [3, 7] розробка СППР для електронної комерції базувалась на аналізі коментарів та відгуків про товари у онлайн-магазині.

**Мета статті.** Метою є аналіз та дослідження проблем щодо розробки програмного забезпечення відповідної системи прийняття рішень, яка б підтримувала вибір користувачами транспортного засобу – велосипеда – для більш точного підбору оптимального велосипеда з урахуванням потреб користувача (наприклад, стилю їзди, покриття, пори року, частоти використання велосипеда), які визначаються шляхом відповідного опитування бажаних придбати велосипед.

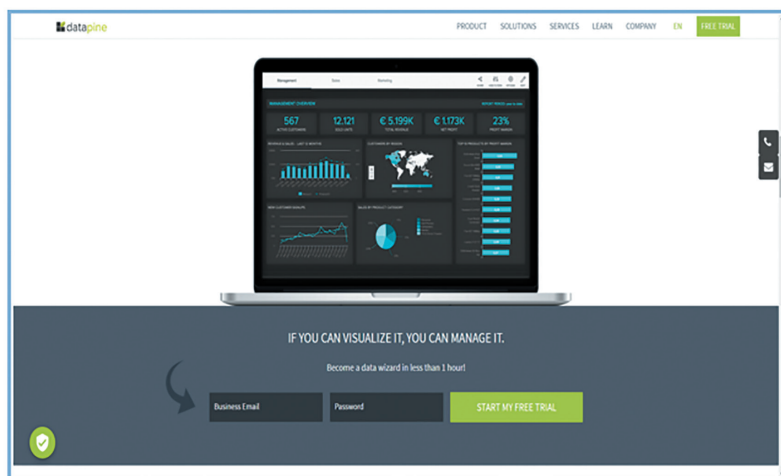
**Виклад основного матеріалу дослідження.** Системи по допомозі підбору будь-яких покупок є доволі специфічними, оскільки, зазвичай, вони потрібні для вибору тільки товарів, параметри та характеристики яких потребують спеціальних знань та компетенцій. Тому, найчастіше такими товарами є техніка (мобільні телефони, комп'ютери, транспортні засоби тощо).

При виборі товару користувачу, який не розуміється на відповідності характеристик того товару, який йому потрібен, складно виявити який велосипед та з якими параметрами йому потрібно обирати.

Багато СППР представляють систему для оцінки ефективності підприємства електронної комерції, аналізуючи інформацію про продажі, закупки, клієнтів та видаючи у результаті графіки розвитку, що більше відноситься до класу задач бізнес-аналітики [4, 5]. Приклад безпосередньо систем електронної комерції:

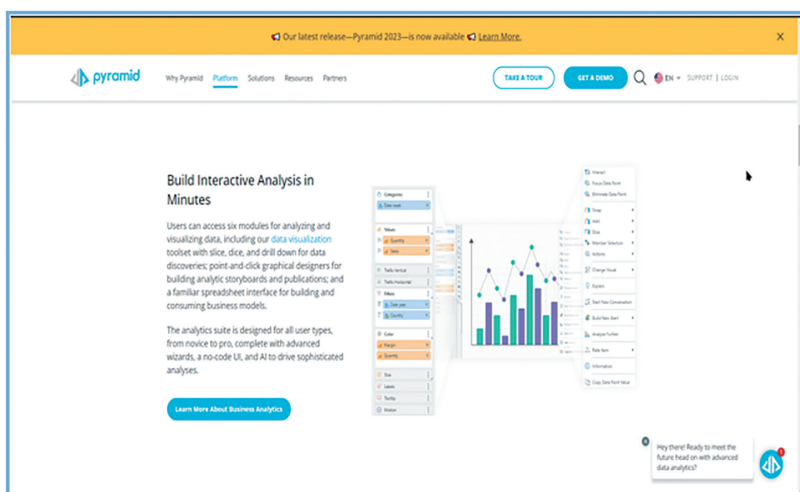


- системи Datarpine [7] наведено на рис. 1;
- системи Pyramid [4] наведено на рис. 2.



**Рис. 1. Сторінка системи підтримки прийняття рішень Datarpine**

Джерело: [7]

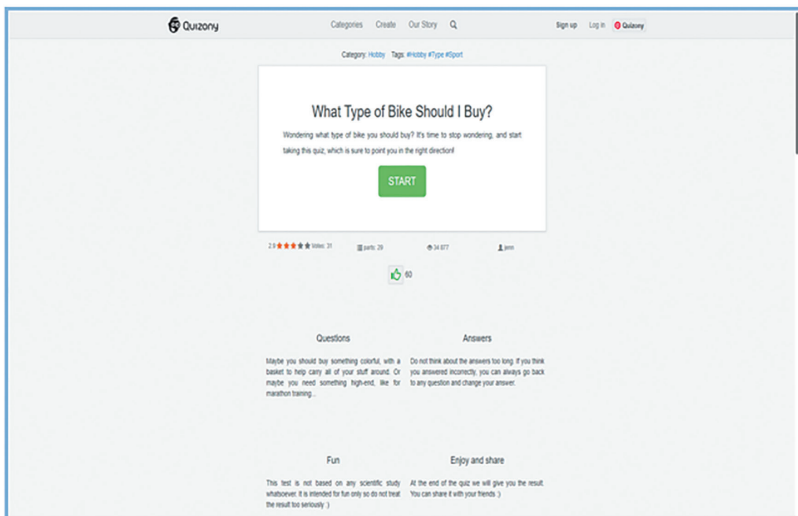


**Рис. 2. Сторінка системи підтримки прийняття рішень Pyramid**

Джерело: [4]

Було проведено дослідження різноманітних інформаційних ресурсів, які можуть допомогти вирішити проблему вибору велосипеда. Зокрема, в [6] використано опитування користувачів, яке має допомогти покупцеві з вибором типу велосипеда для покупки, але це опитування більше схоже на звичайну гру, яка не має жодного відношення до справжнього вибору велосипеда. В системі [6] для користувача також є текстові сторінки (рис.3), на яких розглядаються конкретні питання щодо використання конкретних технічних характеристик тих чи інших транспортних засобів (зокрема, велосипедів).

Весь вибір велосипеда користувачем при цьому може спиратися тільки на текстові підказки щодо вибору, і часто мають вже застарілу чи неповну інформацію (щодо технічних характеристик, призначення, нових марок тощо). Такі рекомендації не можуть сприйматися користувачами як достовірні на 100%.



**Рис. 3. Сторінка опитування для підбору велосипеда**

Джерело: [6]

Робота [10] більше розкриває проблему саме вибору конкретних типів велосипедів з урахуванням деяких їх характеристик (рис 4).

## 1 Стиль руху VS велосипед

Універсальних велосипедів не існує, кожен призначений для конкретного стилю руху на певній території. Розповімо про основні 15 типів.

1. **Гірський велосипед (MTB, mountain bike)** – найпопулярніший байк, його завдання їздити поганими шляхами й пересіченою місцевістю.

У таких моделях посилена рама, як мінімум передній амортизатор, широкі покриття з яскраво вираженим протектором. На всіх гірських моделях велосипедист сидить нахилившись вперед.

2. **Найнер** – це гірський велосипед з колесами 29", на відміну від звичайних 26".

Велосипеди-найнери краще долають перешкоди та вибоїни, їдуть плавно, у них менший тиск на ґрунт. Відмінне рішення для високих байкерів.

3. **Фрірайдовий велосипед** створений для катання складними, природними або штучними трасами з перешкодами – стрибки на швидкості, подопання схилів, рух вузькими звивистими стежками.

Моделі для фрірайду – різновид гірських, з міцною рамою, передньою та задньою підвісками великого ходу, потужними дисковими гальмами.

У них:

- міцна та легка рама для ефективних стрибків і підйомів в гору;
- передній і задній амортизатор з великим ходом для ефективного гасіння коливань;
- потужні дискові гідравлічні гальма;
- широкі шини з глибоким малюнком протектора для підвищеної прохідності.

4. **Крос-кантрі велосипед** – досить важка модель для перегонів пересіченою місцевістю – зі спусками, зтяжними підйомами, перешкодами, швидкісними і технічними ділянками. Такий велосипед – ще один різновид гірського, але зроблений під потреби дисципліни крос-кантрі.

У крос-кантрі моделі міцна рама з низькою посадкою, дискові гальма, трансмісія високого рівня, вилка з амортизацією, контактні педалі та шини з вираженим протектором.

### Рис. 4. Тестові рекомендації зі сторінки «Як вибрати велосипед»

Джерело: [10]

Але в [10] використовуються терміни, які часто незрозумілі звичайному покупцеві, тому не дуже обізаному у цьому виді транспортних засобів, все рівно доведеться обирати велосипед самостійно, хоч і схиляючись до цікавого для нього типу велосипеду.

Оскільки СППР по вибору велосипеду має бути доступна кожному, то її слід реалізувати у вебпросторі, адже кількість користувачів, які мають доступ до Інтернету значно більша, ніж тих, хто використовує тільки телефони, чи тільки персональні комп'ютери.

Такий підхід також дозволить збирати інформацію про результати вибору велосипеду, додавати до системи нові знання і варіанти остаточного вибору.

Окрім цього модернізація такої системи для всіх користувачів не стане великою проблемою, адже усі користувачі будуть мати найактуальнішу версію цієї вебсистеми при зверненні до неї.

Головне завдання авторської системи – допомога у виборі велосипеда для людей, які на них не знаються, тому ця система має бути максимально зручною для використання і для в ній слід підтримувати такі можливості, зокрема, як:

- простий і зрозумілий стиль зв'язку між користувачем та системою (для зв'язку використовується графічний інтерфейс, який надає перелік питань і декілька варіантів відповідей на ці питання);
- просто і зрозуміло сформульовані питання та відповіді (для спрощення комунікації, користувачу надаються прості питання на кшталт «Для чого Вам велосипед?» та зрозумілі відповіді, такі як «Для катання по місту, як відпочинок», «Для їзди по трасі» та «Для їзди на роботу/навчання» (рис. 5);
- достатня кількість даних в інформаційній базі системи (для коректної оцінки потреб користувачів системи кількість питань має бути саме така, щоб було виявлено усі головні потреби користувача; а кількість велосипедів у розпорядженні системи така, щоб на кожний набір відповідей було б декілька варіантів);
- простий вибір з результатів (велосипеди схожі між собою за реалізацію, але кожен з них має ті чи інші особливості, тому при виведенні результатів опитування важливо надати користувачеві можливість вибору з декількох варіантів, які будуть виводитись від найбільшого співпадіння з побажаннями користувача до найменшого, при цьому надається перелік неспівпадаючих параметрів) (рис. 6).



**Рис. 5. Форма опитування**

Джерело: авторська система

Вся структура авторської системи базується на встановленні ваг характеристик (потреб, вподобань, тощо) при опитуванні та отримання на їх основі найбільш підходящих результатів щодо конкретних моделей велосипедів.

Саме опитування можна представити наступним чином:

- користувачеві надається питання та варіанти відповідей на нього;
- проходження опитування;
- надання користувачеві списку моделей велосипедів, які найбільш підходять під його потреби та побажання.



**Рис. 6. Пройдене опитування та виведені результати**

Джерело: авторська система

Крім повних співпадінь користувачеві надаються також моделі велосипедів, у яких не стовідсотковий збіг, бо вони можуть зацікавити користувача візуально (своїм виглядом), а разом із моделлю виводиться причина, чому та чи інша модель може не підійти користувачеві (його потребам і побажанням).

Алгоритм роботи авторської системи підтримки прийняття рішень щодо вибору велосипедів (рис. 7) складається, зокрема з:

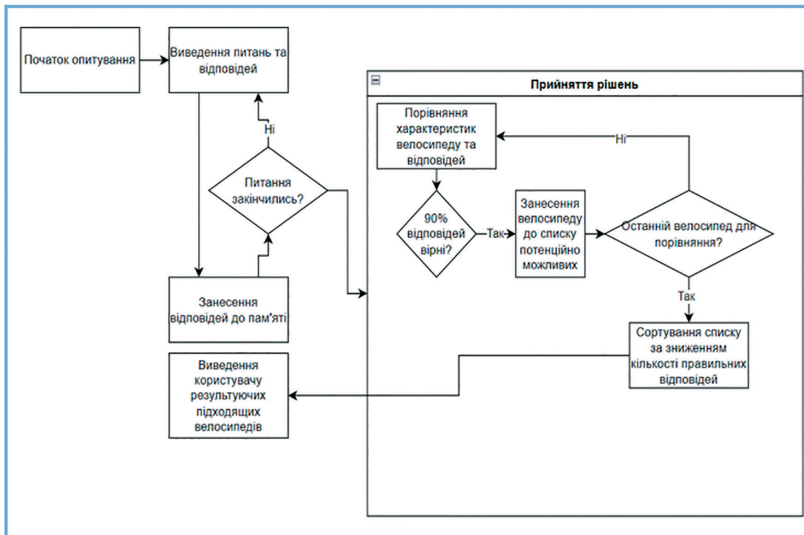
- проведення опитування користувача та отримання від нього відповідей на задані питання;

- проведення аналізу відповідей користувача на задані питання;
- виявлення підходящих під відповіді моделей велосипедів;
- формування результату, в якій потрапляють моделі що мають відсоток сумісності з потребами (вподобаннями користувача) більший за пороговий, який становить 90% сумісності з відповідями користувача.

Адміністратор СППР вибору велосипедів повинен постійно заповнювати систему новими даними.

Характеристики моделей велосипедів визначаються експертами.

Беручи до уваги проблеми вибору звичайним користувачем будь-якої техніки (в нашому випадку – велосипеду), розробка авторської СППР щодо вибору такої речі, як велосипед, є проектом, що дозволить потенційним покупцям зменшити час при виборі потрібної моделі велосипеду.



**Рис. 7. Алгоритм роботи СППР вибору велосипедів**

Джерело: авторська система

**Висновки та пропозиції.** Використання розробленої авторської СППР вибору велосипеда дозволяє, зокрема:

- надавати користувачеві зручну можливість при визначенні щодо його вибору і покупки велосипеда;
- створити простий зв'язок між користувачем та системою за рахунок використання графічного інтерфейсу;
- користувачеві не потрібно знатись на технічних аспектах та характеристиках велосипеда, через те що запитання ставляться у зрозумілому форматі без використання термінології та технічних назв, у той час як відповіді є однозначним висвітленням потреби користувача;
- згенерувати декілька варіантів рішень вибору велосипедів, що відповідають потребам (побажанням) користувача згідно з розрахунком відповідності характеристик різних марок велосипедів потребам (побажанням) користувача.

Використання користувачами авторської СППР вибору велосипеда дозволить полегшити багатьом здійснити свій вибір у цьому класі техніки, а зі сторони магазинів та виробників, при модернізації системи зворотнім зв'язком із відповідями, отримати уявлення щодо найпопулярніших моделей велосипедів.

© **Ткаченко О.А., Ткаченко О.І., Делант О.О., 2023**

## ЛІТЕРАТУРА

1. Система підтримки рішень. URL: [https://uk.wikipedia.org/wiki/Система\\_підтримки\\_рішень](https://uk.wikipedia.org/wiki/Система_підтримки_рішень) (дата звернення: 19.05.2023).
2. Segal T. (2022). Decision support system (DSS): what it is and how businesses use them. URL: <https://www.investopedia.com/terms/d/decision-support-system.asp> (дата звернення: 19.05.2023).
3. Chandra Ya., Karya S., Hendrawaty M. Comments from Marketplace E-Commerce Sites in Indonesia: A Proposed Model. International Journal on Advanced Science Engineering and Information Technology, 2019. 9(4):1171. DOI:10.18517/ijaseit.9.4.6505. URL: [https://www.researchgate.net/publication/336584949\\_Decision\\_Support\\_Systems\\_forCustomer\\_to\\_Buy\\_Products\\_with\\_an\\_Integration\\_of\\_Reviews\\_and\\_Comments\\_from\\_Marketplace\\_E-Commerce\\_Sites\\_in\\_Indonesia\\_A\\_Proposed\\_Model](https://www.researchgate.net/publication/336584949_Decision_Support_Systems_forCustomer_to_Buy_Products_with_an_Integration_of_Reviews_and_Comments_from_Marketplace_E-Commerce_Sites_in_Indonesia_A_Proposed_Model) (дата звернення: 21.05.2023).

4. The pyramid decision intelligence platform. URL: (дата звернення: 22.05.2023).
5. Minitab's solutions solve your greatest analytics challenges. URL: <https://www.minitab.com/en-us/> (дата звернення: 22.05.2023).
6. What Type of Bike Should I Buy? URL: <https://www.quizony.com/what-type-of-bike-should-i-buy> (дата звернення: 23.05.2023).
7. Datapine – the smart decision support software. URL: <https://www.datapine.com/decision-support-system> (дата звернення: 22.05.2023).
8. What Is the Consumer Decision Support System? URL: <https://smallbusiness.chron.com/consumer-decision-support-system-81538.html>. (дата звернення: 28.05.2023).
9. Westerman S.J., Tuck G.C., Booth S.A., Khakzar K. Consumer decision support systems: Internet versus in-store application. *Computers in Human Behavior*, 2007. Vol. 23 (6). P. 2928-2944. URL: <https://www.sciencedirect.com/science/article/abs/pii/S0747563206000914>. (дата звернення: 28.05.2023).
10. Андрейчук М. Як вибрати велосипед. URL: <https://hotline.ua/guides/yak-vibrati-velosiped/> (дата звернення: 19.05.2023).

## REFERENCES

1. «Decision support system», available at: [https://uk.wikipedia.org/wiki/Система\\_підтримки\\_рішень](https://uk.wikipedia.org/wiki/Система_підтримки_рішень) (Accessed 19 May 2023).
2. Segal, T. (2022). Decision support system (DSS): what it is and how businesses use them, available at: <https://www.investopedia.com/terms/d/decision-support-system.asp> (Accessed 19 May 2023).
3. Chandra, Ya., Karya, S., Hendrawaty, M. (2019). Comments from Marketplace E-Commerce Sites in Indonesia: A Proposed Model. *International Journal on Advanced Science Engineering and Information Technology*. 9(4):1171. DOI:10.18517/ijaseit.9.4.6505, available at: [https://www.researchgate.net/publication/336584949\\_Decision\\_Support\\_Systems\\_for\\_Customer\\_to\\_Buy\\_Products\\_with\\_an\\_Integration\\_of\\_Reviews\\_and\\_Comments\\_from\\_Marketplace\\_E-Commerce\\_Sites\\_in\\_Indonesia\\_A\\_Proposed\\_Model](https://www.researchgate.net/publication/336584949_Decision_Support_Systems_for_Customer_to_Buy_Products_with_an_Integration_of_Reviews_and_Comments_from_Marketplace_E-Commerce_Sites_in_Indonesia_A_Proposed_Model) (Accessed 21 May 2023).
4. The pyramid decision intelligence platform, available at: <https://www.pyramidanalytics.com/decision-intelligence-platform> (Accessed 22 May 2023).
5. Minitab's solutions solve your greatest analytics challenges, available at: <https://www.minitab.com/en-us/> (Accessed 22 May 2023).
6. What Type of Bike Should I Buy? available at: <https://www.quizony.com/what-type-of-bike-should-i-buy> (Accessed 23 May 2023).



7. Datapine – the smart decision support software, available at: <https://www.datapine.com/decision-support-system> (Accessed 22 May 2023).

8. What Is the Consumer Decision Support System? available at: <https://smallbusiness.chron.com/consumer-decision-support-system-81538.html>. (Accessed 28 May 2023).

9. Westerman, S.J., Tuck, G.C., Booth, S.A., Khakzar, K. (2007). Consumer decision support systems: Internet versus in-store application. *Computers in Human Behavior*. Vol. 23 (6). P. 2928-2944. available at: <https://www.sciencedirect.com/science/article/abs/pii/S0747563206000914> (Accessed 28 May 2023).

10. Andrejchuck, M. «How to choose a bicycle», available at: <https://hotline.ua/guides/yak-vibrati-velosiped/> (Accessed 19 May 2023).

**СТАТТЯ НАДІЙШЛА ДО РЕДАКЦІЇ 06.06.2023**

УДК 004.415.2

DOI: <https://doi.org/10.53920/ITS-2023-1-5>

**Валерій Вікторович ЗАВГОРОДНІЙ,**

доктор технічних наук, професор,  
завідувач кафедри інформаційних технологій,  
Державного університету інфраструктури та технологій  
ORCID ID: 0000-0002-8347-7183

**Ганна Анатоліївна ЗАВГОРОДНЯ,**

кандидат технічних наук, доцент,  
доцент кафедри інформаційних технологій,  
Державний університет інфраструктури та технологій  
ORCID ID: 0000-0001-8523-1761

**Ігор Андрійович ЯКИМЕНКО,**

магістр кафедри інформаційних технологій,  
Державний університет інфраструктури та технологій  
ORCID ID: 0009-0005-7631-9469

**Максим Юрійович САВЧУК,**

магістр кафедри інформаційних технологій,  
Державний університет інфраструктури та технологій  
ORCID ID: 0009-0007-0665-2332

## **ПРОЄКТУВАННЯ ВІРТУАЛЬНИХ СЕРВЕРІВ НА ОСНОВІ ТЕХНОЛОГІЇ КОНТЕЙНЕРИЗАЦІЇ**

***Було надано детальний опис архітектури проєкту, включаючи взаємодію компонентів додатка та інші аспекти проєктування хмарної платформи компанії Netcracker.***

***Роз'яснено поняття віртуалізації та основні типи віртуалізації, які існують на сьогоднішній день, а також обґрунтовано вибір конкретного стеку технологій для проєктування хмарної платформи компанії Netcracker. Незважаючи на те, що хмарна платформа компанії Netcracker вже працює та показує високу продуктивність, зі збільшенням кількості контейнерів зростатиме навантаження на оркестратор та кластер в цілому. Тому в даний час необхідно провести дослідження з метою виявлення слабких місць у системі та вжиття заходів для їх усунення, використовуючи нові підходи до технології контейнеризації, що постійно оновлюються новими інструментами та отримують підтримку від різних виробників рішень, базованих на віртуалізації на рівні операційної системи.***

**Завдяки використанню каналів передачі даних компанії Netcracker було розроблено рішення, яке характеризується високою продуктивністю, масштабованістю та легкістю внесення змін.**

**Були наведені ілюстрації та діаграми майбутньої хмарної платформи Netcracker, яка об'єднує різні технології для створення повноцінного веб-додатку. Також ця платформа включає технології, необхідні для створення розширеного середовища, шляхом виділення окремих елементів системи ізольованих сервісів.**

**Було проведено системний аналіз проєкту, вивчено архітектуру мікросервісів та технологію віртуалізації з метою розробки системи на основі мікросервісної архітектури. В результаті дослідження було встановлено, що для розробки систем, які ґрунтуються на мікросервісах, найбільш підходящою є технологія віртуалізації на рівні операційної системи.**

**Проаналізовано приклади абстрактних проєктів, які надають короткий опис принципу взаємодії контейнерів між собою. Виявлено, що контейнеризація є ефективним підходом для розробки таких систем, оскільки дозволяє ізолювати та масштабувати окремі компоненти системи, забезпечуючи гнучкість та надійність у взаємодії між контейнерами.**

**Ключові слова:** архітектура, мікросервіси, технологія віртуалізації, хмарна платформа, Netcracker, контейнеризація.

**Valerii ZAVGORODNII**

Doctor of technical sciences, Professor,  
Head of the Department of Information Technologies,  
State University of Infrastructure and Technologies

**Anna ZAVGORODNYA**

Candidate of technical sciences, Associate Professor,  
Associate Professor of the Department of Information  
Technologies,  
State University of Infrastructure and Technologies

**Ihor YAKYMENKO**

Master of the Department of Information Technologies,  
State University of Infrastructure and Technologies

**Maksym SAVCHUK**

Master of the Department of Information Technologies,  
State University of Infrastructure and Technologies

## DESIGN OF VIRTUAL SERVERS BASED ON CONTAINERIZATION TECHNOLOGY

*A detailed description of the project architecture was provided, including the interaction of the application components and other aspects of Netcracker's cloud platform design.*

*The concept of virtualization and the main types of virtualization that exist today are explained, as well as the choice of a specific stack of technologies for the design of the cloud platform of the Netcracker company is justified. Despite the fact that Netcracker's cloud platform is already working and showing high performance, with the increase in the number of containers, the load on the orchestrator and the cluster as a whole will increase. Therefore, it is currently necessary to conduct research in order to identify weak points in the system and take measures to eliminate them, using new approaches to containerization technology, which are constantly updated with new tools and receive support from various manufacturers of solutions based on virtualization at the level of the operating system.*

*Thanks to the use of Netcracker's data channels, a solution has been developed that is characterized by high performance, scalability and ease of modification.*

*Illustrations and diagrams of the upcoming Netcracker cloud platform, which combines various technologies to create a full-fledged web application, were shown. Also, this platform includes the technologies necessary to create an extended environment by separating individual elements of the system of isolated services.*

*System analysis of the project was carried out, microservices architecture and virtualization technology were studied in order to develop a system based on microservices architecture. As a result of the research, it was established that for the development of systems based on microservices, virtualization technology at the level of the operating system is the most suitable.*

*Examples of abstract projects that provide a brief description of the principle of interaction between containers have been analyzed. Containerization has been found to be an effective approach for the development of such systems, as it allows isolation and scaling of individual system components, providing flexibility and reliability in the interaction between containers.*

**Keywords:** *architecture, microservices, virtualization technology, cloud platform, Netcracker, containerization.*

**Постановка проблеми.** У сучасному світі багато компаній, що займаються розробкою програмного забезпечення, використовують віртуалізацію для створення масштабованих, надійних та повнофункціональних систем. Ці системи також є економічно вигідними і конкурентоспроможними у швидкозростаючій бізнес-моделі, заснованій на хмарних сервісах [1, 2].

Актуальність і наукова значимість даного дослідження полягають у тому, що в наш час, особливо враховуючи епідеміологічну ситуацію, зростає попит на онлайн-бізнес. Це призводить до значного збільшення користувачів як на нових, так і на існуючих системах. У даній роботі проводяться дослідження різних технологій віртуалізації та архітектур для розробки програмного забезпечення, що відповідають вимогам високої надійності, продуктивності та гнучкості розгортання.

Головна проблема полягає в пошуку потрібної технології, яка змогла б вирішити низку питань, що виникли через епідеміологічну ситуацію у світі для компаній, що займаються інтернет-бізнесом. Факт полягає в тому, що з наведених причин велика кількість користувачів почала робити покупки або здійснювати дії в цифровому сегменті, що призвело до перевантаження систем. З метою вирішення цієї проблеми різні компанії почали масштабувати своє апаратне забезпечення, що призвело до значних фінансових витрат. Як результат, виникла суперечка щодо проектування систем у багатьох компаній, і тому виникло питання про більш ефективне використання технологій з метою зниження витрат.

**Аналіз останніх досліджень і публікацій.** Згідно з [3], віртуалізація відіграє важливу роль у технології хмарних обчислень. У хмарних обчисленнях користувачі зазвичай обмінюються даними, що містяться в додатках, розгорнутих на хмарних платформах. «Хмара», або хмарна обчислювальна система, представляє собою онлайн-сховище, що складається з великої кількості серверів, які об'єднані в одну загальну мережу. Доступ до цього сховища здійснюється за допомогою спеціальних веб-сайтів або додатків. Але насправді, завдяки використанню віртуалізації, дана інфраструктура розподіляється для досягнення гнучкості і стабільності системи.

Згідно з дослідженням [4], усі облаштовані на облікованій платформі хмарного хостингу ресурси повинні ефективно розподілятися між численними користувачами з метою забезпечення

безпеки додатків, що є важливими згідно вимог замовника. Такий підхід передбачає спільне використання ресурсів, таких як обчислювальна потужність, оперативна пам'ять та дисковий простір, на загальній фізичній інфраструктурі у форматі багатокористувачького середовища.

За допомогою розвитку хмарних технологій, розгортання мікросервісних систем стало більш ефективним, гнучким і економічно вигідним підходом. Проте в [5] відзначається, що мікросервіси є складною темою, яку з особливим інтересом вивчають академічні установи, IT-компанії та промисловість. Вперше термін «мікросервіси» обговорювався на конференції «Архітектура програмного забезпечення» у травні 2011 року з метою узгодження загального архітектурного стилю серед учасників. Через рік, на цій же конференції, термін «мікросервіси» був офіційно підтверджений інженерною групою. Фактично, мікросервісна архітектура була розроблена як реакція на проблеми, що виникають у монолітних додатках, які мають обмежену масштабованість та гнучкість [6, 7].

Монолітна архітектура відноситься до способу побудови програмного додатка, де модулі не можуть працювати незалежно один від одного. Однак розробка, ґрунтована на мікросервісній архітектурі, розглядається як єдиний спосіб виконувати незалежні інструкції, що не взаємодіють між собою [8].

За словами Річардсона, мікросервісна архітектура представляє собою варіант сервісно-орієнтованої архітектури програмного забезпечення, що акцентує на взаємодії невеликих, слабо зв'язаних та легко змінюваних модулів, відомих як мікросервіси. Цей підхід став широко розповсюдженим у середині 2010-х років завдяки розвитку гнучких методик розробки та DevOps [9].

На відміну від традиційних варіантів сервісно-орієнтованої архітектури, де модулі можуть бути складними програмними системами, а взаємодія між ними зазвичай ґрунтується на стандартизованих тяжких протоколах (наприклад, SOAP, XML-RPC), мікросервісна архітектура ґрунтується на компонентах, що виконують прості функції, та використовує ефективні мережеві комунікаційні протоколи (наприклад, REST з використанням JSON, Protocol Buffers, Thrift) для взаємодії між ними [10].

Зважаючи на необхідність ізоляції різних компонентів додатка та підтримки модульної незалежності при використанні техно-

логії мікросервісів, інженерна спільнота стикалася з проблемою вибору відповідної технології для досягнення цієї мети [8, 10]. Таким рішенням стала технологія віртуалізації.

Використання технології контейнерної віртуалізації разом із мікросервісною архітектурою дозволяє розробляти масштабні веб-додатки, що задовольняють вимогам високої продуктивності, надійності та гнучкості системи.

**Мета статті** – дослідження можливостей технологій контейнерної віртуалізації та мікросервісної архітектури для розробки хмарної платформи, що вирішує проблеми високої продуктивності, гнучкості розгортання та надійності.

**Виклад основного матеріалу дослідження.** Технологія віртуалізації є основною складовою хмарних обчислень. У традиційному підході до хмарних обчислень використовуються віртуальні машини для розподілу ресурсів та створення ізольованого середовища для користувачів. На одному фізичному сервері в хмарній інфраструктурі можуть бути розгорнуті та запущені кілька віртуальних машин з власною операційною системою та службами. Проте, останнім часом все більш популярною стає легковажна технологія віртуалізації на основі контейнерів. Основна відмінність між віртуальними машинами та контейнерами полягає в тому, що контейнери використовують одну загальну базову операційну систему.

Мікросервісна архітектура з використанням контейнерної віртуалізації широко прийнята як основний інструмент для розробки великих додатків. Ця концепція отримала значну підтримку від спільноти Інтернету, що привело до потреби використання цієї технології при розробці нових хмарних рішень, зокрема в сфері електронної комерції. Запит на такі рішення виник серед замовників разом із розширенням хмарних обчислень. У зв'язку з цим компанія Netcracker прийняла рішення розробити хмарну платформу, яка надасть замовникам гнучку, масштабовану та надійну систему, засновану на мікросервісній архітектурі та супутніх технологіях.

Розвиток хмарних технологій та технологій віртуалізації активізує роботу компаній, що спеціалізуються на наданні програмних рішень на основі хмарних обчислень. Однією з таких компаній є Netcracker.

Netcracker Technology є підрозділом корпорації NEC, спеціалізуються на розробці, впровадженні та підтримці систем опера-

тивної підтримки (OSS), систем підтримки бізнесу (BSS) і рішень SDN/NFV для операторів зв'язку, великих підприємств і державних установ.

Протягом багатьох років рішення Netcracker BSS і OSS не були масштабованими і стали занадто складними для подальшого використання, що обмежувало їх конкурентоспроможність. У зв'язку з великим ризиком масштабних трансформацій системи, постачальники послуг зараз частіше надають перевагу тому, щоб почати з невеликої системи та поетапно розширювати її під свої потреби.

Використання хмарних технологій стає необхідним кроком для забезпечення бізнесу та операцій наступного покоління, оскільки це сприяє збільшенню гнучкості, безпеки, надійності та масштабованості. Netcracker пропонує власну хмарну платформу, відому як Netcracker Cloud Platform, яка дозволяє клієнтам ефективно керувати своїм бізнесом та операціями, використовуючи переваги хмарних технологій.

#### *Розділення додатків на мікросервіси*

З метою визначення необхідних мікросервісів для хмарної платформи компанії Netcracker було проведено аналіз у співпраці з замовником. Основною метою аналізу було зібрати критерії та узгодити потреби замовника з можливостями платформи. Були виокремлені такі завдання:

1. Розробка платформи, яка дозволить користувачам купувати хмарні сервіси з можливістю підписки. Для досягнення цієї мети необхідно створити портал для цифрових сервісів нового покоління, хмарних додатків і віртуалізованих сервісів. Основна функція цього порталу полягатиме в забезпеченні взаємодії з кінцевими користувачами. Для нього пропонується назва «Shopping Portal» або «інтернет-магазин».

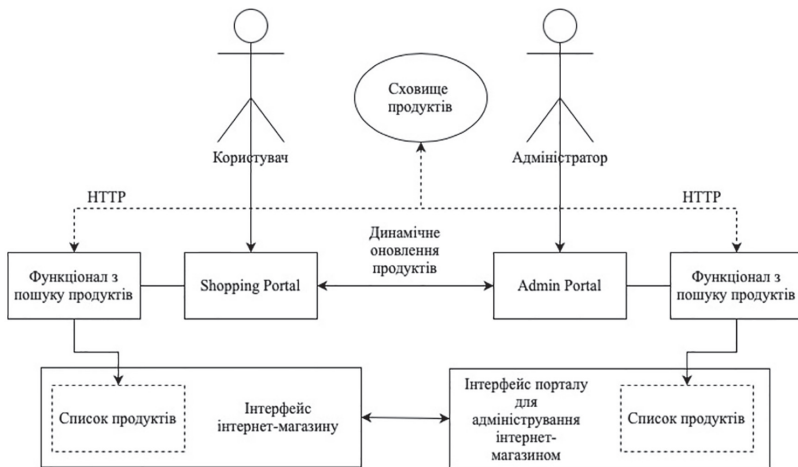
2. Розробка порталу для адміністрування інтернет-магазину, який буде відповідальним за редагування, видалення та додавання нових продуктів, а також за всі взаємодії з користувачами та налаштування інтернет-магазину. Цей портал отримав назву «Admin Portal» або «портал для адміністрування інтернет-магазину».

У цьому випадку, використання віртуальних серверів дозволить забезпечити швидку та безпечну роботу з порталами. Шляхом розбиття додатку на окремі сервіси, які незалежно виконують



свої функції, можна досягти більшої ефективності. Це дозволить легко та швидко редагувати та оновлювати окремі частини додатку без зайвих труднощів.

Візьмемо, наприклад, модель взаємодії між користувачем інтернет-магазину та динамічним оновленням списку продуктів після внесення змін через портал адміністрування. Обидва ці портали вже є окремими мікросервісами, які будуть мати власні контейнери та віртуальні сервери. Поділ додатка на такі окремі частини спростить взаємодію між порталами та дозволить виділити загальну логіку в окремий контейнер (рис. 1).



**Рис. 1. Клієнтська модель взаємодії користувача з інтернет-магазином та динамічне оновлення списку продуктів**

Як бачимо на рисунку 1, інтернет-магазин та портал адміністрування мають обидва відображати список продуктів. Це вимагає реалізації функції отримання продуктів з бази даних на одному з порталів, що створює певні труднощі, оскільки цей функціонал повинен бути доступним для використання на інших порталах. Щоб вирішити цю проблему, ми можемо скористатися можливостями мікросервісної архітектури та контейнеризації.

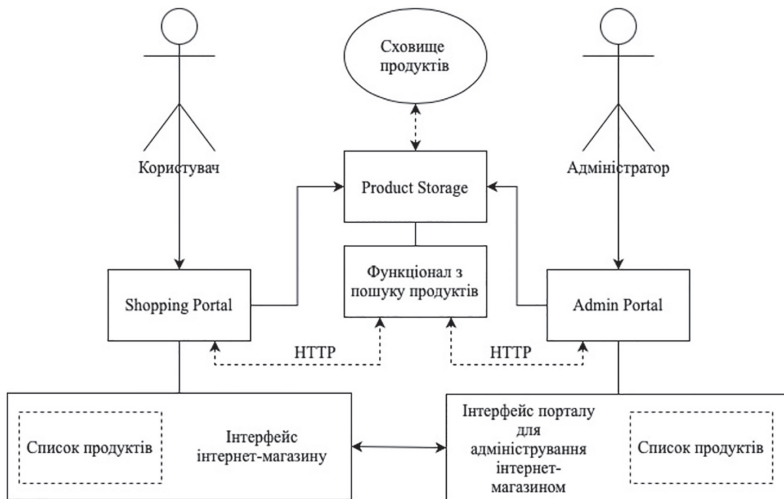
Для цього ми виділимо всю необхідну логіку в окремий мікросервіс, який буде працювати у власному контейнері. Мікросервіс

матиме свої власні змінні середовища та може мати власну базу даних або використовувати базу даних інших мікросервісів. Таке розподілення функціоналу дозволить нам забезпечити незалежність цієї функціональності від конкретного порталу і спростить її використання на інших платформах.

Таким способом досягається відокремлення логіки роботи з продуктами від загального сервісу. На рисунку 2 показано демонстрацію цього сервісу.

Новий мікросервіс отримує назву «Product Storage» і включатиме наступні функціональні завдання:

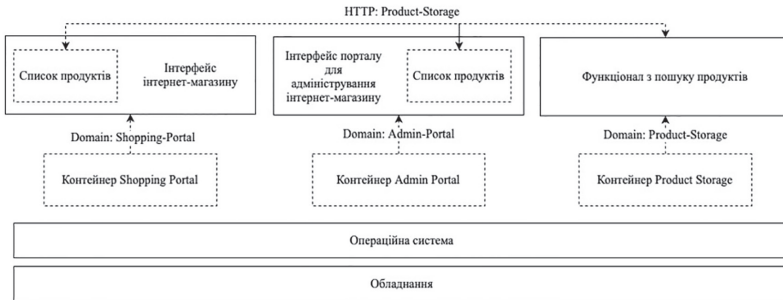
1. Зберігання продуктів у базі даних та можливість змінювати, додавати та видаляти продукти зі списку.
2. Надання відкритих посилань для взаємодії із продуктами через протокол HTTP з іншими мікросервісами.
3. Універсальність функціоналу, що підходить для різних порталів.



**Рис. 2. Виділення функціоналу для отримання списку продуктів на окремий сервіс**

У даному випадку, технологія контейнеризації грає пряму роль, оскільки для перевірки функціоналу необхідно оновлення версій контейнерних образів.

На рисунку 3 відображений реальний досвід роботи з контейнерами та представлено початкову схему розробки хмарної платформи компанії Netcracker.



**Рис. 3. Логічна схема контейнерів та встановлених на них додатків**

Ключові моменти даної схеми:

1. Для кожного додатку виділено свій контейнер, на якому запущений додаток. Додаток може мати кілька сутностей для забезпечення надійності, при цьому кожен контейнер незалежний один від одного та керується контейнерним рушієм.

2. Кожен контейнер має свій домен, що відповідає концепції доменного управління для проектування мікросервісів. Обмежений контекст явно визначає межі моделі.

3. Взаємодія між додатками та сервісами відбувається за протоколом HTTP.

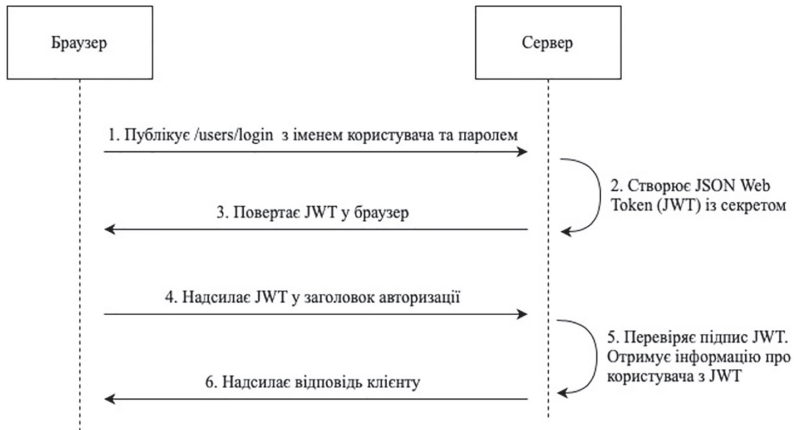
Перевага такої узгодженості в різних додатках полягає у наявності стандарту при виконанні різних дій.

#### *Сервіси авторизації та механізм виявлення служб*

Для створення хмарної платформи Netcracker необхідно побудувати таку архітектуру проекту, щоб мікросервіси взаємодіяли між собою та утворювали компактні компоненти додатка, які разом створюють стійку екосистему. Розглянемо певну частину сервісів, що складають основну частину хмарної платформи Netcracker, а також їх взаємодію, починаючи з сервісу авторизації.

Сервіс авторизації використовує токен для аутентифікації користувача і не зберігає статус користувача. Клієнт повинен надсилати токен на сервер для аутентифікації кожного разу при запиті.

Основний процес аутентифікації користувача з використанням токена зображений на рисунку 4.



**Рис. 4. Процес аутентифікації користувача в режимі токена**

Процес аутентифікації користувача аналогічний базовому процесу аутентифікації з токеном, з додаванням шлюзу API як вхідної точки зовнішнього запиту. Цей сценарій передбачає, що всі запити проходять через шлюз API, що ефективно приховує мікросервіси.

З точки зору продуктивності системи порівняно з монолітною системою, такий підхід є вигідним, оскільки сервіси авторизації працюють через механізм виявлення сервісів шлюзу API, що є правильним та безпечним з архітектурної точки зору. Щодо шлюзу API, за допомогою технології контейнеризації можна створити кілька екземплярів цього сервісу для збільшення пропускної здатності системи, оскільки більшість запитів будуть направлені саме до цього сервісу.

*Модель взаємодії мікросервісів у хмарній платформі, що використовує канали передачі даних компанії Netcracker*

Монолітний додаток взаємодіє між своїми компонентами шляхом виклику методів або функцій на мовному рівні. Натомість, мікросервісна архітектура полягає у розподіленій системі, що працює на кількох віртуальних машинах. Зазвичай, кожен мікро-

сервіс є окремим процесом. Тому для взаємодії між сервісами необхідно використовувати механізми міжпроцесової взаємодії (IPC).

Кожен сервіс може використовувати різні методи взаємодії. Деяким сервісам достатньо одного механізму IPC, тоді як інші можуть використовувати комбінацію різних механізмів. На прикладі інтернет-магазину (рис. 5), сервіси взаємодіють за допомогою сповіщень, запитів/відповідей та публікацій/підписок, коли користувач додає товар до кошика і отримує сповіщення.



**Рис. 5. Взаємодія сервісів між собою при додаванні товару в кошик та виведення повідомлення**

Сервіси використовують комбінацію сповіщень, запитів/відповідей та публікацій/підписок. Наприклад, коли користувач взаємодіє з інтернет-магазином і надсилає запит до сервісу кошика, сервіс «Портал покупок» передає запит через «Шлюз API» до сервісу, який відповідає за кошик. Після отримання запиту, сервіс «Кошик покупок» звертається до «Сховища цінових пропозицій», щоб зберегти товар у кошику та перевірити його валідність. У той же час «Кошик покупок» очікує відповіді від сервісу «Сховище цінових пропозицій» і, коли вона надходить, виводить сповіщення за допомогою сервісу «Сервіс сповіщень».

Під час використання обміну повідомленнями, процеси взаємодіють асинхронно шляхом передачі повідомлень. Клієнт надсилає запит до сервісу, відправляючи йому повідомлення. Якщо передбачається отримання відповіді від сервісу, він відправляє

окреме повідомлення з відповіддю клієнту. Оскільки комунікація є асинхронною, клієнт не блокується під час очікування відповіді. Замість цього клієнт очікує відповіді, припускаючи, що вона не буде надіслана негайно.

**Висновки та пропозиції.** Науковий характер даного дослідження полягає у тому, що хмарна платформа компанії Netcracker має здатність не лише надавати клієнтам рішення щодо продажу цифрових продуктів, але й гнучку можливість створення веб-додатків, які можуть бути швидко адаптовані до потреб певної галузі бізнесу і оперативно введені в експлуатацію.

Використання можливостей віртуалізації дозволяє створювати й підтримувати великі та складні системи, які характеризуються високою гнучкістю, ефективністю, зручністю у супроводженні та високою продуктивністю. В даному контексті віртуалізація означає запуск декількох операційних систем на одному комп'ютері, з використанням спільних апаратних ресурсів.

Даний підхід дозволяє покращити не лише кодову базу, а й сприяє підвищенню надійності та масштабованості системи. Досягнення цих поліпшень забезпечується принципом ізоляваності відмов: мікросервіси повинні бути здатні до самостійного розгортання та роботи. Без цього принципу, хмарна платформа Netcracker втратить свої основні критерії – масштабованість та стійкість до відмов.

Практична розробка хмарної платформи на основі контейнерної віртуалізації та мікросервісної архітектури має значущість для комерційних організацій, що потребують розробки стійкої та гнучкої системи з можливістю масштабування.

© **Завгородній В.В., Завгородня Г.А., Якименко І.А., Савчук М.Ю., 2023**

## ЛІТЕРАТУРА

1. Priyanka Tyagi. E Commerce for Entrepreneurs. BPB Publications, 2020. P. 168. ISBN 9789389898408.
2. Dan Croxen-John, Johann van Tonder. E-Commerce Website Optimization. Kogan Page, 2020. P. 272. ISBN 9781789664430.
3. Plauth, M., Feinbube, L., Polze, A. A Performance Survey of Lightweight Virtualization Techniques. In: De Paoli, F., Schulte, S., Broch Johnsen, E. (eds) Service-Oriented and Cloud Computing. ESOC 2017. Lecture Notes in Com-

puter Science(), vol 10465. Springer, Cham. [https://doi.org/10.1007/978-3-319-67262-5\\_3](https://doi.org/10.1007/978-3-319-67262-5_3).

4. J. Bogner, J. Fritzsich, S. Wagner and A. Zimmermann. *Microservices in Industry: Insights into Technologies, Characteristics, and Software Quality*, 2019 IEEE International Conference on Software Architecture Companion (ICSA-C), Hamburg, Germany, 2019, pp. 187-195, doi: 10.1109/ICSA-C.2019.00041.

5. Xia, C., Zhang, Y., Wang, L., Coleman, S., & Liu, Y. *Microservice-based cloud robotics system for intelligent space*. *Robotics and Autonomous Systems*, 2018, 110, 139-150. <https://doi.org/10.1016/j.robot.2018.10.001>.

6. Chris Richardson. *Inter-Process Communication in a Microservices Architecture*. URL: <https://dzone.com/articles/building-microservices-inter-process-communication-2> (дата звернення: 30.05.2023).

7. Kalske, M., Mäkitalo, N., Mikkonen, T. (2018). *Challenges When Moving from Monolith to Microservice Architecture*. In: Garrigós, I., Wimmer, M. (eds) *Current Trends in Web Engineering. ICWE 2017. Lecture Notes in Computer Science()*, vol 10544. Springer, Cham. [https://doi.org/10.1007/978-3-319-74433-9\\_3](https://doi.org/10.1007/978-3-319-74433-9_3).

8. Chris Richardson. *Microservices Patterns*. Manning Publications, 2018. P.520. ISBN 9781617294549.

9. Julien Vehent. *Securing DevOps*. Manning Publications, 2018. P.520. ISBN 9781617294136.

10. Shayank Jain. *Designing Microservices using Django*. BPB Publications, 2020. P.332. ISBN 9789389328790.

## REFERENCES

1. Tyagi, P. (2020), «E Commerce for Entrepreneurs». BPB Publications. P. 168. ISBN 9789389898408.

2. Dan Croxen-John, Johann van Tonder (2020), «E-Commerce Website Optimization». Kogan Page. P. 272. ISBN 9781789664430.

3. Plauth, M., Feinbube, L., Polze, A. (2017), *A Performance Survey of Lightweight Virtualization Techniques*. In: De Paoli, F., Schulte, S., Broch Johnsen, E. (eds) *Service-Oriented and Cloud Computing. ESOC 2017. Lecture Notes in Computer Science()*, vol 10465. Springer, Cham. [https://doi.org/10.1007/978-3-319-67262-5\\_3](https://doi.org/10.1007/978-3-319-67262-5_3).

4. J. Bogner, J. Fritzsich, S. Wagner and A. Zimmermann (2019), «*Microservices in Industry: Insights into Technologies, Characteristics, and Software Quality*», 2019 IEEE International Conference on Software Architecture Companion (ICSA-C), Hamburg, Germany, P. 187-195, doi: 10.1109/ICSA-C.2019.00041.

5. Xia, C., Zhang, Y., Wang, L., Coleman, S., & Liu, Y. (2018). Microservice-based cloud robotics system for intelligent space. *Robotics and Autonomous Systems*, 110, 139 – 150. <https://doi.org/10.1016/j.robot.2018.10.001>.

6. Richardson, C. (2016), «Inter-Process Communication in a Microservices Architecture», available at: <https://dzone.com/articles/building-microservices-inter-process-communication-2> (Accessed 30 May 2023).

7. Kalske, M., Mäkitalo, N., Mikkonen, T. (2018). Challenges When Moving from Monolith to Microservice Architecture. In: Garrigós, I., Wimmer, M. (eds) *Current Trends in Web Engineering. ICWE 2017. Lecture Notes in Computer Science*, vol 10544. Springer, Cham. [https://doi.org/10.1007/978-3-319-74433-9\\_3](https://doi.org/10.1007/978-3-319-74433-9_3).

8. Richardson, C. (2018), «Microservices Patterns», Manning Publications. P.520. ISBN 9781617294549.

9. Vehent, J. (2018), «Securing DevOps», Manning Publications. P. 520. ISBN 9781617294136.

10. Jain, S. (2020), «Designing Microservices using Django», BPB Publications. P. 332. ISBN 97893389328790.

**СТАТТЯ НАДІЙШЛА ДО РЕДАКЦІЇ 29.05.2023**



УДК 004.9:658.78

DOI: <https://doi.org/10.53920/ITS-2023-1-6>

**Андрій Вікторович ЛЕМЕШКО,**

доктор філософії з комп'ютерної інженерії, доцент,  
Державний університет телекомунікацій  
ORCID ID: 0000-0001-8003-3168

**Артем Васильович АНТОНЕНКО,**

кандидат технічних наук, доцент,  
Державний університет телекомунікацій  
ORCID ID: 0000-0001-9397-1209

**Віталій Олександрович СЛЮСАР,**

магістр,  
Державний університет телекомунікацій  
ORCID ID: 0009-0005-9975-2177

**Дмитро Вадимович БАХУРИНСЬКИЙ,**

магістр,  
Державний університет телекомунікацій  
ORCID ID: 0009-0004-1967-5363

**Микита Олександрович КУЦЕНКО,**

магістр,  
Державний університет телекомунікацій  
ORCID ID: 0000-0001-9720-5181

## **ТЕХНОЛОГІЇ ХМАРНИХ ОБЧИСЛЕНЬ ТА ДИНАМІКА ЇХ РОЗВИТКУ**

**Хмарні обчислення розглядаються як технологія наступного покоління. Це веб-технологія, за допомогою якої користувачам надаються якісні послуги, включаючи дані та програмне забезпечення, на віддалених серверах. Хмарні обчислення схожі на аутсорсинг даних, оскільки зовнішній постачальник надає клієнту послуги зберігання даних. При цьому клієнти отримують хороший результат, без великих витрат на обладнання та програмування для зберігання інформації. Хмарні обчислення усувають необхідність мати повну інфраструктуру програмного та апаратного забезпечення для задоволення вимог клієнтів і додатків.**

**Цей вид технології можна розглядати як повний або неповний аутсорсинг апаратних і програмних ресурсів. Для доступу до хмар-**

**них програм потрібне швидке підключення до інтернету та стандартний Інтернет-браузер. Хмарна технологія пропонує масштабований доступ за запитом до спільного пулу ресурсів, розміщених у центрі обробки даних на сайті провайдера.**

**Ключові слова:** хмарні обчислення, публічна хмара, приватна хмара, колективна хмара, гібридна хмара, апаратна віртуалізація, програмна віртуалізація, сервісні моделі.

**Andriy LEMESHKO**

Doctor of philosophy in computer engineering,  
Associate Professor,  
State University of Telecommunications

**Artem ANTONENKO**

Candidate of technical sciences, Associate Professor,  
State University of Telecommunications

**Vitalii SLIUSAR**

Master's student,  
State University of Telecommunications

**Dmytro BAKHURYNSKYI**

Master's student,  
State University of Telecommunications

**Mykyta KUTSENKO**

Master's student,  
State University of Telecommunications

## **CLOUD COMPUTING TECHNOLOGIES AND DYNAMICS OF THEIR DEVELOPMENT**

***The cloud is made up of millions of computers – big ones and small ones – connected together through physical wires or satellite systems that converge in vast data storage centers, themselves containing tens of thousands of «servers», each one with millions of chips, which provide memory and processor power. Businesses store data on it; anything that requires lots of memory and number-crunching uses it.***

***Cloud computing brings significant benefits to business and the economy in general, and the speed of its adoption is critical to creating new businesses and improving the efficiency of existing ones. Cloud computing is seen as the next generation technology. It is a web technology that provides users with quality services, including data***

***and software, on remote servers. Cloud computing is similar to data outsourcing in that an external provider provides data storage services to the client. At the same time, clients get a good result, without high costs for equipment and programming for information storage. Cloud computing eliminates the need to have a complete software and hardware infrastructure to meet customer and application requirements. This type of technology can be considered as full or partial outsourcing of hardware and software resources. Access to cloud applications requires a fast Internet connection and a standard Internet browser. Cloud technology offers scalable, on-demand access to a shared pool of resources hosted in a data center at the provider's site. This paper provides a brief overview of cloud computing, deployment models and service models, cloud computing techniques, and virtualization. A study of the global market for providing services in the field of cloud technologies was conducted. The largest providers of cloud services have been identified, the total market share of which reaches 80%. As a result of the marketing research, it was determined that the main providers of cloud services are American companies and Chinese companies.***

***As a result of the sanctions imposed on Russia due to its invasion of Ukraine, most major cloud service providers have stopped selling products to Russian customers, limited services, foregone revenue and impeded Russia's access to advanced technologies. This, in turn, led to a slowdown in the growth and gradual degradation of cloud computing technologies.***

***Keywords:*** *cloud computing, public cloud, private cloud, community cloud, hybrid cloud, hardware virtualization, software virtualization, service models.*

**Постановка проблеми.** Термін «хмара» означає мережу або інтернет. Це означає, що хмара – це щось, що доступне у віддаленій місцевості. Хмара може надавати переваги над системою чи інтернетом, тобто у відкритих або приватних системах. Такі програми, як електронна пошта, веб-конференції, управління взаємовідносинами з клієнтами (CRM), усі працюють у хмарі. Хмарні обчислення залежать від веб-обчислень, де віртуальні спільні сервери надають клієнтам пристрої платформи програмної інфраструктури та інші ресурси та хостинг з огляду на компенсацію за використання послуг. Усі дані, які може запропонувати оцифрована система, надаються як послуга в моделі хмарних обчислень.

Хмарні обчислення надають своїм клієнтам численні можливості, як-от доступ до великої кількості використання без вимоги наявності дозволу, купівлі, впровадження чи завантаження будь-якої з цих програм. Це також зменшує витрати на експлуатацію та встановлення комп'ютерів і програмного забезпечення, оскільки немає потреби мати будь-яку інфраструктуру. Клієнти можуть отримати доступ до даних будь-де; все, що їм потрібно, це взаємодія з системою (зазвичай інтернет). Клієнти хмарних обчислень не претендують на фізичну структуру, а орендують її у стороннього постачальника. Вони споживають ресурси як послугу та платять лише за ресурси, які вони використовують. Більшість розподілених обчислювальних основ складається з адміністрацій, що передаються через регулярні фокуси та базуються на серверах. У сучасному світі кожна організація повинна почати шукати, де саме хмарні обчислення (СХ) потрібні в її бізнесі, щоб отримати конкурентну перевагу, ставши і залишаючись конкурентоспроможними у своєму бізнес-секторі. Винятковою характеристикою хмарних обчислень є оплата за використання, оскільки користувач хмари повинен платити лише за використані послуги [1].

Хмарні обчислення пропонують більш розширене сховище, ніж традиційні системи зберігання. Програмне забезпечення оновлюється та дуже роботизується зі зменшенням кількості найнятого виключно талановитого ІТ-персоналу [2]. Ця стаття пояснює короткий огляд хмарних обчислень, їх послуг і моделей розгортання, визнаючи різні характеристики хмарних обчислень і технологій хмарних обчислень.

**Аналіз останніх досліджень і публікацій.** Наукові дослідження технологій хмарних обчислень та ринку надання послуг у сфері хмарних технологій є актуальним завданням, розв'язання якого дозволить узагальнити ключові фактори впровадження хмарних технологій у світі, виявити особливості хмарного ринку та проаналізувати перспективи хмарних обчислень у Росії.

Значний внесок у вирішення фундаментальних питань у сфері технологій хмарних обчислень є дослідження таких вітчизняних та зарубіжних вчених: О Лазарева, О Гавриленка, О Куца, Жерара Ле Мау, Дейва Катлера, Ларрі Еллісона та ін. [1 – 13].

**Метою роботи** є: узагальнення особливостей технологій хмарних обчислень, огляд ринку постачальників хмарних послуг та аналіз перспектив хмарних обчислень у Росії з урахуванням санкцій.

Об'єкт дослідження – технології хмарних обчислень та їх розгортання.

Предмет дослідження – особливості побудови технологій хмарних обчислень та динаміка їх розвитку.

**Виклад основного матеріалу.** Існують різні типи служб і моделей, які дозволяють хмарним обчисленням бути практичними та відкритими для кінцевих клієнтів. Здебільшого існує два типи робочих моделей для розподілених обчислень: моделі розгортання і сервісні моделі.

Моделі розгортання: моделі розгортання характеризують тип доступності хмарного середовища. Хмара поділяє доступність на чотири типи: приватна, публічна, колективна та гібридна.

1) Публічна хмара: ця хмара доступна для всіх зовнішніх клієнтів через інтернет, які можуть зареєструватися в хмарі та використовувати ресурси хмари за моделлю оплати за використання. Ця хмара не є безпечною, як приватна хмара. Завдяки своїй відкритості вона доступна для всіх користувачів інтернету. Вона відносно менше налаштовується, ніж приватна хмара. Хмарною інфраструктурою володіє та керує великий постачальник хмарних послуг (CSP). Постачальник хмарних технологій відповідає за створення та постійну підтримку загальнодоступної хмари та її IT-ресурсів. Відкриту хмару так само називають зовнішньою хмарою, де ресурси надаються через інтернет. Прикладами загальнодоступної хмари є: електронна пошта, Google AppEngine, Microsoft Azure або Window Azure та Amazon Elastic Compute Cloud (EC2).

2) Приватна хмара: ця хмара налаштована спеціально для організації в межах власного центру обробки даних. Організації керують усіма хмарними ресурсами, які їм належать. Приватна хмара забезпечує більшу безпеку на відміну від відкритої або гібридної хмари. Ресурси приватної хмари не такі економічні, як загальнодоступні хмари, але вони пропонують більшу продуктивність, ніж відкрита хмара. Хмара контролюється асоціацією та обслуговується виключно нею; хмара може існувати всередині або поза периметром організації. Приватна хмара також називається внутрішньою або корпоративною хмарою, яка надає полегшені гаджети заздалегідь визначеній кількості осіб за брендмауером.

3) Колективна хмара: кілька організацій разом створюють і пропонують подібну хмарну інфраструктуру, а також політики, ви-

моги, цінності та проблеми. Колективна хмара формує рівень економічної масштабованості та демократичної рівноваги. Хмарну інфраструктуру може підтримувати сторонній постачальник або одна з організацій у колективі. Хмара контролюється кількома організаціями та підтримує певний колектив, який має подібні інтриги. хмара спільноти більш безпечна, ніж публічна хмара.

4) Гібридна хмара: це група публічних, приватних і колективних хмар. Однак критична діяльність досягається приватною хмарою, тоді як некритична діяльність досягається публічною хмарою. Загальнодоступна хмара є дорожчою за приватну, тому гібридна хмара може заощадити. Моделі гібридної хмари залежать від внутрішньої ІТ-інфраструктури, тому важливо гарантувати надлишок центрів обробки даних. Наприклад, клієнт хмари може передавати хмарні послуги, що обробляють конфіденційні інформаційні дані, у приватну хмару, а інші менш чутливі хмарні послуги – у загальнодоступну хмару.

Сервісні моделі: хмара може зв'язуватися з клієнтом (клієнтом або додатком) у ряді курсів за допомогою можливостей, які називаються сервісами [3]. Моделі послуг – це функціональні моделі, на яких базуються хмарні обчислення. В Інтернеті з'явилися три основні типи послуг або моделі послуг.

1. Інфраструктура як послуга (IaaS).
2. Платформа як послуга (PaaS).
3. Програмне забезпечення як послуга (SaaS).

1) Інфраструктура як послуга (IaaS): постачальники хмарних обчислень пропонують фізичні та віртуальні комп'ютери, додаткові гаджети для організації ємності тощо. Віртуальні машини контролюються гіпервізорами, які розподіляються на пули та контролюються оперативними мережами емоційної підтримки. Обов'язком хмарних клієнтів є запровадження робочих зображень фреймворку на віртуальних машинах, а також програмування їх додатків. IaaS дозволяє хмарному провайдеру відкрито знаходити інфраструктуру в інтернеті практичним способом. Ресурси IaaS, такі як сховища, пропускна здатність, служби моніторингу, IP-адреси, брандмауери, віртуальні машини тощо, все це надається споживачеві в оренду. Споживач повинен платити залежно від часу, протягом якого споживач утримує ресурс.

Прикладами інфраструктури як послуги є: Rackspace, Windows Azure, Amazon EC2, Google Compute Engine.

2) Платформа як послуга (PaaS): це надання платформи для розробки та розгортання додатків через інтернет як послугу для розробників, які можуть використовувати платформу для легкого створення, розгортання та керування додатками SaaS. Вона також пропонує вдосконалення та розгортання пристроїв, необхідних для створення програм. Основним елементом PaaS є пристрій «наведи і зафіксуй», який дає змогу не дизайнерам створювати веб-додатки. Покупець не вимагає придбання дорогих серверів, обладнання та джерел живлення та зберігання інформації. Таким чином, зменшити або збільшити масштаб на основі запитів ресурсів додатків зовсім не важко.

Прикладами платформ як послуги є: Force.com, Google, Apache StratosApp, Engine, Windows Azure, AWS Elastic Beanstalk.

3) Програмне забезпечення як послуга (SaaS): це передача програм (наприклад, ERP або CRM) для допомоги кінцевим клієнтам через інтернет за допомогою браузерів. Клієнти хмари можуть використовувати те, що наразі встановлено та працює в хмарній інфраструктурі. Таким чином, немає необхідності встановлювати та запускати програмне забезпечення лише на ПК. А також зменшується потреба в обслуговуванні та підтримці програмного забезпечення. Деякі програми SaaS не адаптуються, наприклад, Office Suite. У будь-якому випадку SaaS надає нам інтерфейс програмування додатків (API), який дозволяє розробникам створювати індивідуальні програми.

Прикладами програмного забезпечення як послуги є: Google Apps, Microsoft Office 365.

Хмарні обчислення мають деякі з наведених нижче характеристик, щоб задовольняти вимоги клієнтів або користувачів і надавати якісні послуги.

1) Висока масштабованість: це означає надання ресурсів у величезних масштабах за запитом, не вимагаючи взаємодії людини з кожним постачальником послуг.

2) Висока доступність і надійність: доступність серверів є більш надійною та високою, що обмежує ймовірність розчарування в інфраструктурі [4].

3) Гнучкість: розподіляє ресурси між користувачами та працює дуже швидко.

4) Спільне використання кількох даних: різноманітні клієнти та програми працюють більш адекватно з меншими витратами

завдяки спільному використанню базової інфраструктури з використанням розподілених обчислень.

5) Технічне обслуговування: технічне обслуговування додатків хмарних обчислень простіше, оскільки їх не потрібно встановлювати на кожному комп'ютері, і до них також можна отримати доступ з різних місць, що зрештою зменшує вартість.

6) Низька вартість: це економічно ефективно, оскільки компанії більше не потрібно створювати власну інфраструктуру. Вона оплачує послуги відповідно до ресурсів, які споживає.

7) Послуги в режимі оплати за використання: API (інтерфейси прикладного програмування) надаються клієнтам для доступу до послуг у хмарі та оплачуються на основі використання послуги.

8) Самообслуговування на вимогу: хмарні обчислення дозволяють клієнтам використовувати служби та ресурси за запитом для взаємодії людини з постачальниками хмарних послуг. Увійти на веб-сайт можна будь-коли, коли працює пакет Office. У будь-якому випадку SaaS надає нам інтерфейс програмування додатків (API), який дозволяє розробникам створювати індивідуальні програми і використовувати їх. Обчислювальні ресурси включають віртуальні машини, обчислювальну потужність, сховище тощо.

9) Широкий доступ до мережі: доступ до таких ресурсів, як віртуальні машини, сховище, обчислювальна потужність, можна отримати через інтернет за допомогою різнорідних гаджетів, таких як мобільні телефони, ноутбуки, комп'ютери тощо. Оскільки хмарні обчислення базуються на інтернеті, доступ до них можна отримати в будь-який час. і звідки завгодно[5].

10) Об'єднання ресурсів: хмарні обчислення дозволяють кільком користувачам спільно використовувати пул ресурсів. Можна спільно використовувати єдиний фізичний екземпляр бази даних, обладнання та базової інфраструктури. Наприклад, фізичний сервер може розміщувати кілька віртуальних машин, що належать різним користувачам [6].

11) Швидка еластичність: дуже легко збільшити або зменшити ресурси в будь-який час. Ресурси, які використовуються клієнтами або наразі призначені клієнтам, автоматично відстежуються та ресурси. Це робить це можливим.

12) Вимірювана послуга: у вимірюваній службі хмарний провайдер контролює та контролює кожен частину хмарної служби.



це залежить від планування потужності, виставлення рахунків за ресурси, оптимізації тощо.

Розглянемо технології хмарних обчислень. Існують різні інновації, які працюють за платформою хмарних обчислень, щоб зробити її надійною, адаптованою та зручною для використання, і це:

- A. Віртуалізація
- B. Сервісно-орієнтована архітектура (SOA)
- C. Грід-обчислення
- D. Допоміжні обчислення
- A. Віртуалізація

Віртуалізація — це процедура, яка ліцензує спільне використання фізичного екземпляра ресурсу або програми між різними клієнтами або організацією. Це здійснюється шляхом присвоєння логічного імені фізичному ресурсу та надання, за необхідності, покажчика на цей фізичний ресурсно [7]. Основне використання цієї технології полягає в тому, щоб надати додаткам стандартну версію для своїх хмарних клієнтів. Наприклад, якщо випущено оновлену версію програми, постачальник хмарних послуг повинен надати оновлену версію своїм клієнтам. Наприклад, VMware і Хер пропонують віртуалізовані ІТ-платформи за запитом. Віртуальна система постійно прогресує, наприклад, віртуальна приватна мережа (VPN), підтримує клієнтів за допомогою модифікованого мережевого середовища для отримання хмарних ресурсів. Методи віртуалізації є основою хмарних обчислень, оскільки вони надають масштабовані та гнучкі апаратні послуги. Архітектура Multitenant пропонує віртуальне розмежування між різними орендарями, і таким чином організації можуть використовувати та налаштовувати програму так само, як кожна з них має свій окремий екземпляр.

Існують такі типи віртуалізації:

- апаратна віртуалізація;
- віртуалізація операційної системи з постачальниками хмарних послуг. Увійти на веб-сайт можна будь-коли, коли працює пакет Office. У будь-якому випадку SaaS надає користувачеві інтерфейс програмування додатків (API), який дозволяє розробникам створювати індивідуальні програми;
- віртуалізація сервера;
- віртуалізація сховищ.

Аналіз ринку найбільших постачальників послуг у сфері хмарних технологій наведено в таблиці 1.

**Таблиця 1. Перелік найбільших постачальників хмарних технологій у світі на 1 січня 2023 року**

№ з/п	Назва компанії	Місце знаходження головного офісу	Частка світового ринку, %	Загальна характеристика компанії
1	Amazon Web Services	Сіетл, Вашингтон, США	34	Це одна з найбезпечніших платформ для хмарних сервісів. Вона пропонує набори інструментів на основі хмари, які є унікальними та відрізняються від інших постачальників у галузі. AWS надає понад 200 повнофункціональних послуг, включаючи зберігання, базу даних і обчислення. Використовуючи послуги AWS, можна легко розробляти гнучкі, масштабовані та надійні програми. Хмара AWS охоплює 99 зон доступності в 31 географічному регіоні по всьому світу
2	Microsoft Azure	Редмонд, Вашингтон, Сполучені Штати Америки.	21	Компанія також є однією з відомих хмарних обчислювальних компаній у світі. Вона є однією із провідних конкурентів Amazon Web Services. Компанія надає користувачам можливості штучного інтелекту та винаходу гібридну хмару. Компанії з дорогим обладнанням і великими центрами обробки даних можуть отримати значну користь від послуг Microsoft Azure. Послуги Microsoft Azure допомагають розробляти, розгортати та керувати програмами у всесвітній мережі. Найкраще те, що вона пропонує пробну версію на 30 днів
3	Google Cloud Platform	Маунтін-В'ю, Каліфорнія, США	11	Компанія займає третє місце серед найбільших хмарних обчислювальних компаній у світі і є світовим лідером у сфері штучного інтелекту, машинного навчання та аналізу даних. Вона надає корпоративні хмарні сервіси та дозволяє розробникам розробляти, розгортати та тестувати програми на його масштабованій інфраструктурі

Продовження таблиці 1

№ з/п	Назва компанії	Місце знаходження головного офісу	Частка світового ринку, %	Загальна характеристика компанії
				GCP служить інтегрованим сховищем, яке використовується підприємствами та розробниками для живих даних. Окрім безкоштовної пробної версії, послуги Google Cloud Platform доступні користувачам за гнучкими планами оплати
4	Alibaba Cloud	Ханчжоу, Китай	5	Компанія відома як четверта за величиною платформа хмарних провайдерів у світі. Сінгапурська компанія може похвалитися надійною хмарною інфраструктурою та надає користувачам широкий спектр хмарних послуг. Деякі відомі хмарні послуги, які пропонує компанія, включають зберігання баз даних, великомасштабні обчислення, візуалізацію мережі, еластичні обчислення, аналітику великих даних і послуги управління
5	IBM Cloud	Нью-Йорк, США	3	Компанія надає глибокі корпоративні рішення для технологічного сектора. Вона пропонує PaaS, SaaS і IaaS за допомогою різних моделей хмарної доставки. Завдяки IBM Cloud користувачі мають свободу вибору бажаних моделей даних, інструментів і моделей доставки для створення програм наступного покоління. Компанія зосереджується на наданні найкращих послуг, щоб додати цінність бізнесу користувача
6	Salesforce	Сан-Франциско, Каліфорнія, США	3	Компанія також є однією з найпопулярніших компаній у сфері хмарних обчислень у всьому світі. Вона надає компаніям усі додатки, необхідні для безперебійної роботи. Від мобільних додатків до обслуговування клієнтів, CRM і ERP, компанія пропонує все на одній платформі. Найпопулярніші хмарні послуги, які надає компанія, включають Service Cloud, Marketing Cloud і Sales Cloud. Хоча Service Cloud допомагає надавати кращі послуги клієнтам, Sales Cloud дозволяє ефективно керувати інформацією про клієнтів

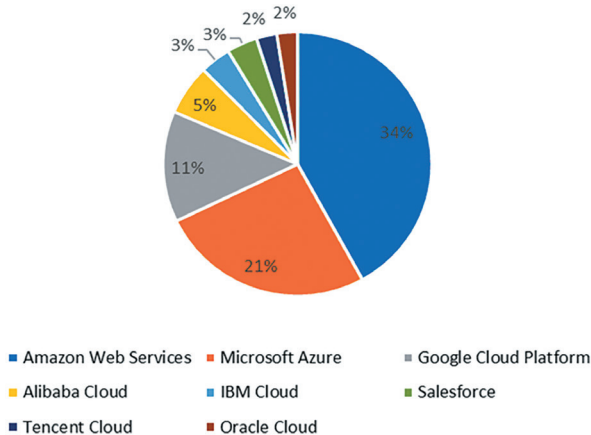
Продовження таблиці 1

№ з/п	Назва компанії	Місце знаходження головного офісу	Частка світового ринку, %	Загальна характеристика компанії
7	Oracle Cloud	Остін, Техас, США	2	Компанія також є однією з провідних хмарних обчислювальних компаній у світі. Вона була заснована пізніше, але з роками набула великої популярності. Основні пропозиції хмарних сервісів компанії включають Oracle Cloud Infrastructure (OCI) і Oracle Cloud Software-as-a-Service (SaaS). Oracle Cloud надає користувачам послуги зберігання, мережевих і обчислювальних послуг. Компанія постійно зростає на конкурентному ринку та отримує хороші відгуки про свою хмарну платформу
8	Tencent Cloud	Шеньчжень, Китай	2	Компанія є одним із провідних постачальників хмарних технологій у світі, який зосереджується на допомозі глобальним підприємствам для досягнення успіху в Китаї. Tencent Cloud пропонує потужне та надійне хмарне рішення, яке спеціально розроблено для вирішення унікальних проблем, з якими стикаються підприємства, коли вони виходять на Китай. Маючи понад один мільярд користувачів, Tencent надає послуги сотням мільйонів людей за допомогою своїх флагманських продуктів, таких як QQ і WeChat
9	Huawei Cloud	Шеньчжень, Китай	2	Компанія охоплює багато зон доступності по всьому світу, забезпечуючи повне підключення, високошвидкісні та стабільні мережі та послуги. За даними China Internet Watch, у 2021 році Huawei Cloud займала 18% ринку витрат на послуги хмарної інфраструктури Китаю. Минулого року компанія запустила свої плани «Go Cloud, Go Global», оголосивши про наміри охопити 170 країн і регіонів у всьому світі

Закінчення таблиці 1

№ з/п	Назва компанії	Місце знаходження головного офісу	Частка світового ринку, %	Загальна характеристика компанії
10	Dell Technologies	Раунд-Рок, Техас, США	< 2	Компанія Dell Technologies Cloud на базі VMware забезпечує узгоджене керування приватною, загальнодоступною та межевою хмарною інфраструктурою, дозволяючи клієнтам вибирати найкраще середовище для кожної програми та випадку використання. Її платформа підтримує понад 4200 основних хмарних провайдерів, включаючи AWS, Microsoft Azure і Google Cloud Platform. Його рішення APEX використовує досвід Dell Technologies, щоб забезпечити підприємствам простий і безпечний шлях до впровадження гібридної хмарної інфраструктури

На рис. 1 представлено діаграму розподілу світового ринку хмарних технологій між найбільшими постачальниками послуг.



**Рис. 1. Розподіл світового ринку хмарних технологій між найбільшими постачальниками послуг**

За оцінками Synergy Research Group, ринкова частка Amazon на світовому ринку хмарної інфраструктури становила 34 відсотки в третьому кварталі 2022 року, все ще перевищуючи сукупну ринкову частку двох її найбільших конкурентів, Microsoft Azure і Google Cloud.

У третьому кварталі 2022 року глобальні витрати на послуги хмарної інфраструктури зросли до 57 мільярдів доларів, у результаті чого загальний обсяг галузі за останні дванадцять місяців склав 217 мільярдів доларів. Як показано на вище наведеній діаграмі, Amazon, Microsoft і Google за три місяці отримали дві третини доходів від хмарної інфраструктури, при цьому вісім найбільших провайдерів контролюють понад 80 відсотків ринку. Серед них понад 70 відсотків становлять послуги хмарної інфраструктури американських постачальників, а близько 10 відсотків становлять послуги хмарної інфраструктури китайських постачальників.

В таблиці 2 наведено кількість регіонів і зон доступності, які охоплюють найбільші постачальники хмарних послуг у всьому світі.

**Таблиця 2. Регіони і зони доступності охоплених найбільшими постачальниками хмарних послуг**

<b>№ за/п</b>	<b>Назва компанії</b>	<b>Кількість регіонів</b>	<b>Кількість зон доступності</b>
1	Amazon Web Services (AWS)	26	84
2	Microsoft Azure	60	116
3	Google Cloud Platform (GCP)	34	103
4	Alibaba Cloud	27	84
5	Oracle Cloud	38	46
6	IBM Cloud (Kyndryl)	11	29
7	Tencent Cloud	21	65
8	OVHcloud	13	33
9	DigitalOcean	8	14
10	Linode (Akamai)	11	11

До військового вторгнення на територію України російські постачальники послуг пропонували широкий спектр онлайн продуктів, починаючи від послуг таксі та доставки їжі до роздрібних

сайтів і ринків усіх видів. Урядові установи, хоч і не настільки просунуті, як приватний сектор, експериментували з цифровим збором податків і різними онлайн-сервісами, такими як заявки на отримання водійських прав і онлайн-голосування. У той час як західні засоби масової інформації, як правило, зосереджувалися – і це справедливо – на темних сторонах російської хмари, таких як кіберзлочинність і придушення політичного інакомислення, швидке поширення та дедалі вдосконалення Інтернету змінили російську економіку та населення. Наприкінці 2021 року Інтернет-революція все ще набирала обертів, оскільки Росія перейшла в еру «великих даних» і цифровізації, по суті, йдучи тим самим шляхом, що й її міжнародні колеги, за винятком приблизно десятилітнього відставання.

Але російська хмара мала одну головну потенційну слабкість – вона майже повністю базувалася на іноземних технологіях та імпорті. За даними російської Вищої школи економіки, залежність російського сектора електроніки від іноземного імпорту напередодні вторгнення була надзвичайною. Імпорт становив 70% доданої вартості технологій, послуг і програмного забезпечення. Для деяких ключових компонентів залежність була ще більшою; з майже 53 000 серверів, які працювали в комерційних центрах обробки даних у 2021 році, майже всі були іноземними. Більшість фізичних зв'язків російської хмари із зовнішнім світом також забезпечувалися оптоволоконними кабелями, встановленими в 1990-х роках у співпраці із західними компаніями.

Потім відбулося російське вторгнення та масове запровадження санкцій Заходу. Технічні компанії, реагуючи на війну в Україні та економічні санкції США, накладені на Росію, припинили продаж продукції, обмежили послуги, відмовились від доходів і перешкоджають доступу Росії до передових технологій. В таблиці 3 наведено перелік компаній у сфері хмарних технологій, які припинили продаж своєї продукції російським споживачам.

Враховуючи санкції, дефіцит поставок, втрату персоналу, кібератаки та загальну нестачу грошей для інвестицій – навіть провідні гравці були змушені скорочуватися. Яндекс, лідер галузі, вийшов зі своїх міжнародних підприємств. Sber довелося відкласти свої амбітні плани побудувати мережу регіональних центрів обробки даних, які пропонують «хмару в оренду».

**Таблиця 3. Перелік компаній  
у сфері хмарних технологій, що припинили продаж  
своєї продукції російським споживачам**

№ з/п	Назва компанії	Заходи з припинення співпраці з російськими компаніями
1	Amazon Web Services (AWS)	Припинено прийом нових клієнтів Amazon Web Services або сторонніх продавців Amazon у Росії чи Білорусі; припинено відвантаження роздрібної продукції покупцям у Росії та Білорусі; призупинено доступ до Prime Video в Росії; надано в Україні послуги з кібербезпеки
2	Microsoft Azure	Призупинено продаж нових продуктів, послуг в Росії; надано в Україні послуги з кібербезпеки
3	IBM Cloud	Припинено продаж технологій
4	Google Cloud Platform	Призупинено надходження від реклами для підтримуваних Росією державних ЗМІ, обмеження послуг у Росії, надання послуг кібербезпеки Україні
5	Oracle Cloud	Призупинено продаж товарів та послуг в Росії
6	Salesforce	Вихід із відносин через торговельних посередників/ інші канали з невеликою кількістю клієнтів у Росії; відсутність будь-якого матеріального бізнесу в Росії
7	VMware	Призупинено продаж/підтримку клієнтів, які потрапили під санкції; заблокована доставка ліцензії партнерам

Незважаючи на ці проблеми, санкції не приведуть до припинення роботи російської хмари найближчим часом. Дійсно, на даний момент основні постачальники послуг оприлюднюють збільшення прибутків – головним чином за рахунок підвищення цін на хмарні послуги, – що дозволило їм продовжувати розширення. Але темпи зростання сповільнюються: від щорічного зростання на 30 – 40% у попередні роки до лише 5% у 2022 році. Це повільне розширення може тривати лише до тих пір, поки доступні сервери, так як імпорт нових серверів майже припинився. Поки санкції залишаються в силі, фізична основа російської хмари поступово деградуватиме, оскільки обладнання потребуватиме заміни, а потреба у захисті від кібератак вимагає модернізації.



Велика невідомість полягає в тому, чи прийде Китай на допомогу. Але, незважаючи на обнадійливі розмови російських IT-керівників, поки що мало що свідчить про це. Наприклад, Huawei, яка лише минулого року відкрила дата-центр у Москві, скоротила свій московський офіс до основного штату та перевела більшість своїх співробітників назад до Китаю чи інших країн СНД. Інші китайські провайдери були так само обережні.

Отже, підсумовуючи, основним наслідком санкцій для Russian Cloud буде уповільнення зростання та поступова деградація, ізоляція від міжнародних тенденцій, хронічна нестача сховищ, зростаючий розрив між Москвою/Санкт-Петербургом і провінцією, а також зростання конкуренції з боку держави за наявні потужності на шкоду приватному сектору. Під загрозою вся «цифрова трансформація» економіки. «Російська хмара» була нащадком короткого відкриття глобалізації Росії. Але допоки тривають санкції, вона все більше приречена бути в повній ізоляції.

Що стосується великої трійки хмарних постачальників – Amazon, Microsoft Azure і Google Cloud, Джон Дінсдейл, головний аналітик фірми Synergy Research, яка відстежує ринок хмарних технологій, зазначив, що на Росію припадає лише частка 1% від загального бізнесу цих компаній, тому відключення російських клієнтів не має великого впливу на бізнес цих компаній.

**Висновки та пропозиції.** Хмарні обчислення можна розглядати як нове явище, яке має змінити спосіб використання Інтернету, тому слід бути обережними. Існує безліч нових досягнень, що розвиваються швидкими темпами, кожен з яких має механічний прогрес і здатність спростити життя людини. У цій роботі наведено короткий огляд хмарних обчислень, моделей розгортання та моделей обслуговування, методів хмарних обчислень та віртуалізації. Проведено дослідження глобального ринку надання послуг у сфері хмарних технологій. Визначено найбільших постачальників хмарних послуг, сумарна частка яких на ринку сягає 80%. В результаті маркетингового дослідження визначено, що основними постачальниками хмарних послуг є американські компанії та китайські компанії. Сумарна частка найбільших американських постачальників хмарних послуг становить більше 70 відсотків, в той час як сумарна частка найбільших китайських постачальників хмарних послуг становить близько 10 відсотків. Внаслідок санкцій, накладених на Росію через її вторгнення на територію України, більшість

великих постачальників хмарних послуг припинили продаж продукції клієнтам Росії, обмежили надання послуг, відмовились від доходів і перешкоджають доступу Росії до передових технологій. Це в свою чергу призвело до уповільнення зростання та поступової деградації технологій хмарних обчислень.

© **Лемешко А.В., Антоненко А.В., Слюсар В.О., Бахуринський Д.В.,  
Куценко М.О., 2023**

## ЛІТЕРАТУРА

1. Abid A., Manzoor M.F., Farooq M.S., Farooq U. & Hussain M. (2020). Challenges and issues of resource. allocation techniques in cloud computing. // *KSII Transactions on Internet and Information Systems (TIIS)*, 14(7), 2815 – 2839.
2. Al Hadwer A., Tavana M., Gillis D. & Rezaia D. (2021). A systematic review of organizational factors impacting cloud-based technology adoption using Technology-organization-environment framework. *Internet of Things*, 15. 100407.
3. Ali O., & Osmanaj V. (2020). The role of government regulations in the adoption of cloud computing: A case study of local government. *Computer Law & Security Review*, 36. 105396.
4. Alkhater N., Walters R., & Wills, G. (2018). An empirical study of factors influencing cloud adoption among private sector organisations // *Telematics and Informatics*. 35(1). 38 – 54.
7. Твердохліб А.О., Коротін Д.С. Ефективність функціонування комп'ютерних систем при використанні технології блокчейн і баз даних // *Таврійський науковий вісник*. Серія: Технічні науки. 2022. (6).
8. Цвик О.С. Аналіз і особливості програмного забезпечення для контролю трафіку // *Вісник Хмельницького національного університету*. Серія: Технічні науки, 2023. (1).
9. Новіченко Є.О. Актуальні засади створення алгоритмів обробки інформації для логістичних центрів // *Таврійський науковий вісник*. Серія: Технічні науки, 2023. (1).
10. Зайцев Є.О. Smart засоби визначення аварійних станів у розподільних електричних мережах міст // *Таврійський науковий вісник*. Серія: Технічні науки, 2022. (5).
9. Attaran M., & Woods J. (2019). Cloud computing technology: improving small business performance using the Internet // *Journal of Small Business & Entrepreneurship*, 31(6). 495–519.

10. Buyya R., Yeo C. S., Venugopal S., Broberg J. & Brandic I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility. *Future Generation Computer Systems*, 25(6). 599 – 616.

11. Danilin I.V. (2021). The U.S.- China Technological War. *Russia in Global Affairs*. 19(4). 78 – 96.

12. Law M. (2023). Top 10 biggest cloud providers in the world in 2023-Dgtl Infra. <https://technologymagazine.com/top10/top-10-biggest-cloud-providers-in-the-world-in-20236>.

13. Gustafson T. Clouding Over: The Impact of Sanctions on the Russian Cloud <https://thanegustafson.substack.com/p/clouding-over-the-impact-of-sanctions>.

## REFERENCES

1. Abid A., Manzoor M.F., Farooq M.S., Farooq U. & Hussain M. (2020). Challenges and issues of resource. allocation techniques in cloud computing // *KSII Transactions on Internet and Information Systems (TIIS)*. 14(7). 2815 – 2839.

2. Al Hadwer A., Tavana M., Gillis D. & Rezania D. (2021). A systematic review of organizational factors impacting cloud-based technology adoption using Technology-organization-environment framework. *Internet of Things*, 15, 100407.

3. Ali O., & Osmanaj V. (2020). The role of government regulations in the adoption of cloud computing: A case study of local government. *Computer Law & Security Review*, 36, 105396.

4. Alkhater N., Walters R., & Wills, G. (2018). An empirical study of factors influencing cloud adoption among private sector organisations // *Telematics and Informatics*, 35(1). 38 – 54.

5. Tverdokhlib A.O., Korotin D.S. Efektyvnist funktsionuvannia kompiuternykh system pry vykorystanni tekhnolohii blokchein i baz dannykh // *Tavriiskyi naukovyi visnyk*. Serii: Tekhnichni nauky. 2022. (6) [in Ukrainian].

6. Tsvyk O.S. Analiz i osoblyvosti prohramnoho zabezpechennia dlia kontroliu trafiku // *Visnyk Khmelnytskoho natsionalnoho universytetu*. Serii: Tekhnichni nauky. 2023. (1) [in Ukrainian].

7. Novichenko Ye.O. Aktualni zasady stvorennia alhorytmiv obrobky informatsii dlia lohistychnykh tsentriv // *Tavriiskyi naukovyi visnyk*. Serii: Tekhnichni nauky. 2023 (1) [in Ukrainian].

8. Zaitsev Ye.O. Smart zasoby vyznachennia avariinykh staniv u rozpodilnykh elektrychnykh merezhakh mist // *Tavriiskyi naukovyi visnyk*. Seria: Tekhnichni nauky. 2022. (5) [in Ukrainian].

9. Attaran M., & Woods J. (2019). Cloud computing technology: improving small business performance using the Internet // *Journal of Small Business & Entrepreneurship*. 31(6). 495 – 519.

10. Buyya R., Yeo C. S., Venugopal S., Broberg J. & Brandic I. (2009). Cloud computing and emerging IT platforms: Vision, hype, and reality for delivering computing as the 5th utility // *Future Generation Computer Systems*. 25(6).599 – 616.

11. Danilin I.V. (2021). The U.S.- China Technological War // *Russia in Global Affairs*, 19(4). 78 – 96.

12. Law M. (2023). Top 10 biggest cloud providers in the world in 2023-Dgtl Infra. <https://technologymagazine.com/top10/top-10-biggest-cloud-providers-in-the-world-in-20236>.

13. Gustafson T. Clouding Over: The Impact of Sanctions on the Russian Cloud <https://thanegustafson.substack.com/p/clouding-over-the-impact-of-sanctions>.

**СТАТТЯ НАДІЙШЛА ДО РЕДАКЦІЇ 15.05.2023**

УДК 378.176

DOI: <https://doi.org/10.53920/ITS-2023-1-7>

**Олександр Анатолійович КОСТИКОВ,**

кандидат фізико-математичних наук, доцент,  
ТОВ «ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
«МЕТІНВЕСТ ПОЛІТЕХНІКА»  
ORCID ID: 0000-0003-3503-4836

**Тетяна Юріївна СОЛОМКО,**

кандидат технічних наук, доцент, проректор,  
Донбаський інститут техніки та менеджменту Закладу вищої освіти  
«Міжнародний науково-технічний університет  
імені академіка Юрія Бугая»  
ORCID ID: 0000-0002-3029-7920

## **ОЦІНКА ЯКОСТІ ТЕСТОВИХ ЗАВДАНЬ МЕТОДАМИ СУЧАСНОЇ ТЕОРІЇ ТЕСТУВАННЯ**

***У статті проведено аналіз якості тестових завдань для контролю знань студентів на основі однопараметричної моделі Раша з використанням інформаційних функцій. Аналіз ґрунтувався на досягненнях сучасної теорії тестування IRT (Item Response Theory). Метою дослідження було виявлення недоліків тесту та визначення засобів їх усунення. Для досягнення цієї мети було наведено алгоритм побудови інформаційних функцій тестових завдань та тесту в цілому для однопараметричної моделі Раша. На основі дослідження інформаційних функцій запропоновано методику аналізу і підвищення якості тестових завдань. Продемонстровано застосування запропонованої методики на прикладі комп'ютерного тестування з дисципліни «Паралельні та розподілені обчислення», яке було проведено для студентів 4-го курсу ЗВО «Міжнародний науково-технічний університет імені академіка Юрія Бугая».***

**Ключові слова:** модель Раша, латентний параметр, інформаційна функція, сучасна теорія тестування IRT.

**Alexander KOSTIKOV**

PhD in Phys. and Math., associative Professor  
associative Professor, LLC «TECHNICAL UNIVERSITY  
«METINVEST POLYTECHNIC»

**Tatyana SOLOMKO**

PhD in Engineering, associative Professor,  
Prorector, IHE «Academician Yuri Bugay  
International science and technical university»  
ORCID ID 0000-0002-3029-7920

## **ASSESMENT OF THE TEST ITEM QUALITY BY THE METODS OF ITEM RESPONSE THEORY**

***In this article the analysis of the test quality based on one-parameter Rush's model using information functions is performed. The analysis was based on advances in Item Response Theory. The purpose of the study was to identify the shortcomings of the test and determine the means to eliminate them. To achieve this goal, an algorithm for constructing information functions of test items and the test for the one-parameter Rush's model was given. Based on the study of information functions, a methodology for analyzing and improving the quality of test items is proposed. The application of the proposed methodology on the example of computer testing in the discipline «Parallel and distributed computing», which was conducted for 4th year students of the IHE «Academician Yuri Bugay International science and technical university», is demonstrated. As result of this demonstration, the non-informative test items were found. Ways to improve the quality of test items were proposed.***

**Keywords:** *Rash's model, latent parameter, information function, modern testing theory IRT.*

**Постановка проблеми.** Нині, у зв'язку з розвитком дистанційного навчання, для оцінювання знань широко застосовується тестовий підхід. Для реалізації тестового підходу в e-Learning дуже активно використовуються комп'ютерні технології. Так, у Міжнародному науково-технічному університеті імені академіка Юрія Бугая для дистанційного навчання використовується Google Classroom. Використання цієї платформи вимагає від викладачів створення електронного курсу і розробки тестових завдань для проведення комп'ютерного тестування. При цьому виникає проблема адекватності оцінювання знань на основі тестових технологій. Використання методів сучасної теорії тестування IRT (Item Response Theory) дозволяє оцінити якість розроблених тестових

завдань і їх придатність для ефективного оцінювання знань студентів.

**Аналіз останніх досліджень і публікацій.** Нині для оцінювання результатів тестування використовуються два підходи: математичний апарат класичної (Classical Test Theory) та сучасної теорії тестування (Item response Theory – IRT).

Відносно переваг та недоліків кожного підходу серед дослідників немає єдиного підходу [1, 2]. Класична теорія тестів від сучасної відрізняється засобом оцінювання учнів, результатом якого є підсумковий бал по тестовому завданню з урахуванням похибки.

У випадку методів IRT підсумковий бал є результатом сукупної взаємодії латентних параметрів, а саме справжнього рівня підготовки учнів та складності запропонованих завдань [1]. Від цих параметрів залежить ймовірність вірної відповіді на тестове завдання.

Для знаходження залежності ймовірності правильної відповіді від вищезазначених латентних параметрів в IRT використовуються різні математичні моделі. Серед них найбільш відомими є однопараметрична модель Раша [3], двохпараметрична та трьохпараметрична моделі Бірнбаума [4].

Однією із складових аналізу на основі IRT є дослідження інформаційної функції як тестового завдання, так і тесту в цілому. На основі інформаційної функції можна зробити висновки про адекватність тестових завдань для оцінки рівня підготовленості студентів.

Поняття інформаційної функції було введено у 1968 році в роботі [4] А. Бірнбаума. В цій роботі була виведена загальна формула інформаційної функції для дихотомічних завдань, тобто для завдань, відповідь на які має вигляд «так» чи «ні». Дослідженню цієї функції присвячено роботи багатьох авторів, зокрема, роботи Ф. Бейкера [5]. Для політомічних завдань (для яких відповіді мають декілька градацій) формула для інформаційної функції була виведена Ф. Сімеємою [6].

Аналіз тестових завдань на основі моделі Раша з використанням інформаційної функції було проведено в роботах авторів [7, 8].

**Мета статті** – провести аналіз розроблених тестових завдань для дистанційного навчання та надати рекомендації по підвищенню їх якості для адекватного оцінювання рівня знань студентів.

**Виклад основного матеріалу дослідження.** Аналіз якості тестових завдань проводився на основі моделі Раша [3], яка виражає залежність між ймовірністю правильної відповіді учня на тестове завдання і латентними параметрами за допомогою формули:

$$P_{ni} = \frac{\exp(\theta_n - \beta_i)}{1 + \exp(\theta_n - \beta_i)}$$

де  $P_{ni}$  – ймовірність того, що учасник  $n$ ,  $n = 1, \dots, N$  з рівнем підготовки  $\theta_n$  вірно виконає завдання  $i$ ,  $i = 1, \dots, l$ , з рівнем складності  $\beta_i$ .

В якості вихідних даних для аналізу було взято результати комп'ютерного тестування з курсу «Паралельні та розподілені обчислення» кафедри комп'ютерних наук та інженерії програмного забезпечення Міжнародного науково-технічний університету імені академіка Юрія Бугая.

В табл. 1 наведено статистичні характеристики відібраних для аналізу десяти тестових завдань для демонстрації запропонованої технології.

**Таблиця 1. Статистичні характеристики тестових завдань**

№ пит.	Успішність	Станд. відхилення	Призначена вага	Ефективна вага	Розрізнення	Ефективність розрізнення
1	33.33%	40.32%	5.0%	0,00%	-32.64%	-35.00%
2	76.00%	41.83%	6.0%	10.57%	34.34%	87.76%
3	66.67%	51.64%	6.0%	11.19%	73.29%	100.00%
4	76.00%	41.83%	6.0%	10.57%	34.34%	87.76%
5	33.33%	40.82%	6.0%	10.78%	92.20%	100.00%
6	100.00%	0.00%	5.0%	0.00%	0,00%	0,00%
7	83.33%	40.82%	5.0%	10.78%	92.20%	100.00%
8	100.00%	0.00%	5.0%	0.00%	0,00%	0,00%
9	50.00%	54.77%	5,0%	10.78%	60.19%	100.00%
10	100.00%	0.00%	5.0%	0.00%	0,00%	0,00%

Джерело: Розроблено авторами

Ці дані використовувались для аналізу якості тестових завдань.

На першому етапі проводилася оцінка рівня підготовки  $i$ -го студента (в логітах) за формулою:



$$\theta_i^0 = \ln\left(\frac{p_i}{q_i}\right), \quad i = 1, 2 \dots N,$$

де  $N$  – кількість учасників тестування,  $p_i$  – доля правильних відповідей  $i$ -го учасника на всі завдання,  $q_i$  – доля неправильних відповідей ( $q_i = 1 - p_i$ ).

Далі обчислювався рівень складності тестових завдань  $\beta_j^0$  в логітах за формулою:

$$\beta_j^0 = \ln\left(\frac{q_j}{p_j}\right), \quad j = 1, 2 \dots M,$$

де  $M$  – кількість тестових завдань,  $p_j$  – доля правильних відповідей всіх учасників на  $j$ -те завдання тесту,  $q_j$  – доля неправильних відповідей.

Далі виконувалося зведення рівня підготовки студентів  $\theta_i^0$  і рівня складності тестового завдання  $\beta_j^0$  до єдиної інтервальної шкали для зниження впливу складності завдань на оцінки учасників тестування.

Попередньо обчисливши середнє значення початкових логітів рівня знань студентів

$$\bar{\theta} = \frac{\sum_{i=1}^N \theta_i^0}{N}$$

та стандартне відхилення  $V$  розподілу початкових значень параметра  $\theta$

$$V^2 = \frac{\sum_{i=1}^N (\theta_i^0 - \bar{\theta})^2}{N - 1},$$

отримаємо формулу для обчислення логіта складності  $j$ -ого завдання

$$\beta_j = \bar{\theta} + Y \cdot \beta_j^0, \quad j = \overline{1, M},$$

$$\text{де } Y = \left(1 + \frac{V^2}{2.89}\right)^{\frac{1}{2}}$$

Аналогічно, обчисливши  $\bar{\beta} = \frac{\sum_{j=1}^M \beta_j^0}{M}$ ,  $W = \sqrt{\frac{\sum_{j=1}^M (\beta_j^0 - \bar{\beta})^2}{M-1}}$ , отримаємо формулу для обчислення логіта рівня знань  $i$ -го студента:

$$\theta_i = \bar{\beta} + X \cdot \theta_i^0, \quad i = \overline{1, N},$$

За допомогою отриманих значень можна співставити рівень знань здобувачів із рівнем складності завдань. Від'ємне і велике за модулем значення різниці  $\theta_i - \beta_j$  означає, що завдання  $\beta_j$  є дуже важким для учасника тестування і не може використовуватися для виміру його рівня знань. У випадку додатної і великої за модулем різниці завдання вважається занадто легким і також воно не придатне для оцінювання рівня знань цього студента. Якщо  $\theta_i = \beta_j$ , то ймовірність того, що студент вірно виконає завдання, дорівнює 0,5.

На наступному етапі знаходяться інформаційні функції для тестового завдання та тесту в цілому.

Для обчислення інформаційної функції  $i$ -го завдання використовується формула:

$$I_i(\theta) = P_i(\theta) \cdot Q_i(\theta)$$

Інформаційна функція тесту обчислюється як сума інформаційних функцій тестових завдань:

$$I(\theta) = D^2 \cdot \sum_{j=1}^M I_j(\theta)$$

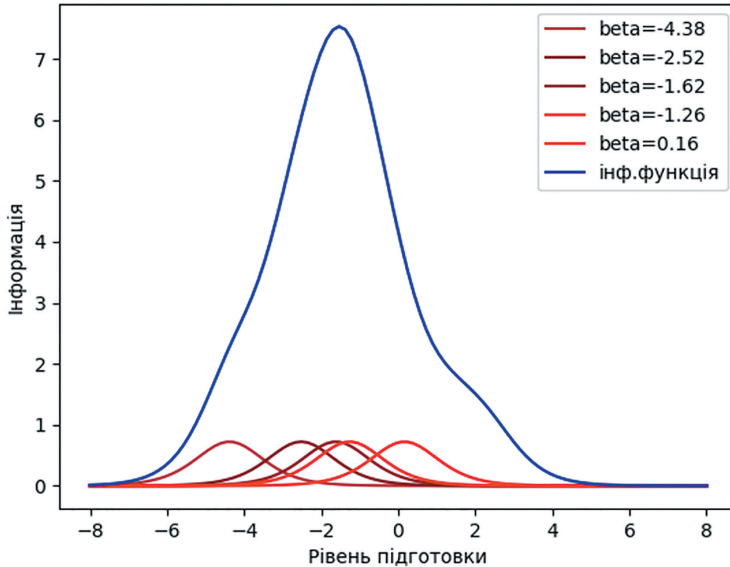
де  $D$  – поправний коефіцієнт ( $D=1.7$ ), необхідний для наближення розподілу логістичної ймовірності до закону нормального розподілу.

На основі інформаційної функції обчислюється похибка вимірювання за формулою:

$$SE(\theta) = \frac{1}{\sqrt{I(\theta)}}$$

Вона використовується для завершення комп'ютерного тестування в адаптивних алгоритмах оцінювання знань з використанням IRT.

Графік інформаційних функцій тестового завдання та тесту в цілому для використовуваних вихідних даних наведено на рис. 1.



**Рис. 1. Інформаційна функція тесту та тестових завдань**

Джерело: Розроблено авторами

Аналізуючи рис.1, можна помітити, що інформаційна функція має один максимум, що означає збалансованість тесту. Також тест має багато легких завдань, про що свідчить розташування інформаційних функцій багатьох тестових завдань в інтервалі (-3,1). Тест містить мало завдань підвищеної складності. Інформаційні функції тестових завдань розташовані нерівномірно, є задачі з близькою складністю, які можна видалити. В той же час необхідно додати до тесту задачі підвищеної складності.

**Висновки та пропозиції.** Аналіз якості тестових завдань на основі IRT виявив такі проблеми:

1) у тесті багато легких завдань та мало складних завдань, що приводить до завищення оцінки рівня навчальних досягнень здобувача освіти;

2) має місце нерівномірність заповнення інтервалів рівнів підготовки інформаційними кривими, що свідчить про недостатню структурованість тестових завдань за рівнем складності, а це, у свою чергу, не дозволяє здійснити об'єктивну диференціацію студентів за рівнем сформованості компетентностей.

Запропоновану методику можна застосовувати для створення тестів різної складності та підвищення якості тестових завдань, що може суттєво підвищити ефективність освітнього процесу з використанням платформ e-Learning.

© **Костіков О.А., Соломко Т.Ю., 2023**

## ЛІТЕРАТУРА

1. Wiberg, M. Classical test theory vs. item response theory [Electronic resource] // *EM* No50. 2004. Way of access: [http://www.edusci.umu.se/digitalAssets/59/59529\\_em-no-50.pdf](http://www.edusci.umu.se/digitalAssets/59/59529_em-no-50.pdf). – Title from the screen

2. Ronald, K. Comparison of classical test theory and item response theory and their applications to test development // *Educational Measurement: issues and practice*. 1993. Way of access: [http://www.internationalgme.org/Resources/Pubs/ITEMS\\_Module\\_16.pdf](http://www.internationalgme.org/Resources/Pubs/ITEMS_Module_16.pdf) – Title from the screen.

3. Rasch G. Probabilistic models for some intelligence and attainment tests. – MESA Press, 5835 S. Kimbark Ave., Chicago, IL 60637; e-mail: MESA@uchicago.edu; web address: [www.rasch.org](http://www.rasch.org); tele, 1993.

4. Birnbaum A. Some latent trait models and their use in inferring an examinee's ability. In F.M. Lord & M.R. Novick, *Statistical theories of mental test scores*. Reading, Mass.: Addison-Wesley. 1968. Chapters 17 – 20.

5. Baker F. *The Basics of Item Response Theory* / F. Baker. – ERIC Clearinghouse on Assessment and Evaluation. 2001. 172 p.

6. Samejima F. Estimation of Latent Ability Using a Response Pattern of Graded Scores / F. Samejima // *Psychometrika Monograph*. № 17. 1969. Pp. 1 – 100.

7. Kostikov A. A. The algorithm for knowledge assessment based on the Rasch model / A. A. Kostikov, K. V. Vlasenko, I. V. Lovianova, S. V. Volkov, E. O. Avramov // *CEUR Workshop Proceedings (9th Illia O. Teplytskyi Workshop on Computer Simulation in Education, CoSinE 2021, Kherson, 1 October 2021, Code 177072)*. 2022. Vol. 3083. pp. 28–42. Режим доступу до повного тексту статті: <http://ceur-ws.org/Vol-3083/paper268.pdf>.

8. Kostikov A, Vlasenko K, Lovianova I, Volkov S, Kovalova D, Zhuravlov M. Assessment of Test Items Quality and Adaptive Testing on the Rasch Model. *Communications in Computer and Information Science*. 2022 Nov 18;1698:252–71. doi:10.1007/978-3-031-20834-8\_12. Режим доступу до повного тексту статті на сайті видання: [https://doi.org/10.1007/978-3-031-20834-8\\_12](https://doi.org/10.1007/978-3-031-20834-8_12).

## REFERENCES

1. Wiberg, M. Classical test theory vs. item response theory [Electronic resource] // *EM* No50. 2004. Way of access: [http://www.edusci.umu.se/digitalAssets/59/59529\\_em-no-50.pdf](http://www.edusci.umu.se/digitalAssets/59/59529_em-no-50.pdf). – Title from the screen.
2. Ronald, K. Comparison of classical test theory and item response theory and their applications to test development // *Educational Measurement: issues and practice*. 1993. Way of access: [http://www.internationalgme.org/Resources/Pubs/ITEMS\\_Module\\_16.pdf](http://www.internationalgme.org/Resources/Pubs/ITEMS_Module_16.pdf) – Title from the screen.
3. Rasch G. Probabilistic models for some intelligence and attainment tests. – MESA Press, 5835 S. Kimbark Ave., Chicago, IL 60637; e-mail: MESA@uchicago.edu; web address: [www.rasch.org](http://www.rasch.org); tele, 1993.
4. Birnbaum A. Some latent trait models and their use in inferring an examinee's ability. In F.M. Lord & M.R. Novick, *Statistical theories of mental test scores*. Reading, Mass.: Addison-Wesley. 1968. Chapters 17 – 20.
5. Baker F. *The Basics of Item Response Theory* / F. Baker. – ERIC Clearinghouse on Assessment and Evaluation. 2001. 172 p.
6. Samejima F. Estimation of Latent Ability Using a Response Pattern of Graded Scores / F. Samejima // *Psychometrika Monograph*. № 17. 1969. Pp. 1 – 100.
7. Kostikov A. A. The algorithm for knowledge assessment based on the Rasch model / A. A. Kostikov, K. V. Vlasenko, I. V. Lovianova, S. V. Volkov, E. O. Avramov // *CEUR Workshop Proceedings (9th Illia O. Teplytskyi Workshop on Computer Simulation in Education, CoSinE 2021, Kherson, 1 October 2021, Code 177072)*. 2022. Vol. 3083. Pp. 28 – 42. Way of access: <http://ceur-ws.org/Vol-3083/paper268.pdf>.
8. Kostikov A, Vlasenko K, Lovianova I, Volkov S, Kovalova D, Zhuravlov M. Assessment of Test Items Quality and Adaptive Testing on the Rasch Model. *Communications in Computer and Information Science*. 2022 Nov 18;1698:252–71. doi:10.1007/978-3-031-20834-8\_12. Way of access: [https://doi.org/10.1007/978-3-031-20834-8\\_12](https://doi.org/10.1007/978-3-031-20834-8_12).

**СТАТТЯ НАДІЙШЛА ДО РЕДАКЦІЇ 07.06.2023**

НАУКОВИЙ ЖУРНАЛ

## «IT SYNERGY»

Заклад вищої освіти  
«Міжнародний науково-технічний університет  
імені академіка Юрія Бугая»  
Вип. 1 (4), Київ, 2023. – 118 с.

Відповідальний за випуск О. І. Бражнікова

Статті збірника проходять обов'язкове рецензування членами редакційної колегії, друкуються мовою оригіналу. Редакція не обов'язково поділяє думку автора і не відповідає за фактичні помилки, яких він припустився.

---

Підписано до друку 29.06.2023.  
Формат 60x90/16. Ум. друк. арк. – 6.79. Тираж 100. Зам. № 271.

---

Друк: Видавництво ТОВ «А-ЦЕНТР».  
Свідоцтво про реєстрацію Серія ДК № 599 від 14.01.2001 р.  
04112, м. Київ, вул. Івана Гонти, 3А.