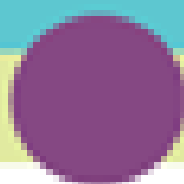


ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СУСПІЛЬСТВО

INFORMATION TECHNOLOGY AND SOCIETY



ISSN 2786-5460 (Print)
ISSN 2786-5479 (Online)

МІЖРЕГІОНАЛЬНА АКАДЕМІЯ УПРАВЛІННЯ ПЕРСОНАЛОМ
INTERREGIONAL ACADEMY OF PERSONNEL MANAGEMENT



**НАУКОВІ ПРАЦІ
МІЖРЕГІОНАЛЬНОЇ АКАДЕМІЇ
УПРАВЛІННЯ ПЕРСОНАЛОМ**

**ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ
ТА СУСПІЛЬСТВО**

**SCIENTIFIC WORKS
OF INTERREGIONAL ACADEMY
OF PERSONNEL MANAGEMENT**

**INFORMATION TECHNOLOGY
AND SOCIETY**

**Випуск 4 (10), 2023
Issue 4 (10), 2023**



**Видавничий дім
«Гельветика»
2023**

*Рекомендовано до друку Вченою радою
Міжрегіональної Академії управління персоналом
(протокол № 10 від 29 листопада 2023 року)*

Інформаційні технології та суспільство / [головний редактор О. Попов]. – Київ : Міжрегіональна Академія управління персоналом, 2023. – Випуск 4 (10). – 32 с.

Журнал «Інформаційні технології та суспільство» є науковим рецензованим виданням, в якому здійснюється публікація матеріалів науковців різних рівнів у вигляді наукових статей з метою їх поширення як серед вітчизняних дослідників, так і за кордоном.

Редакційна колегія не обов'язково поділяє позицію, висловлену авторами у статтях, та не несе відповідальності за достовірність наведених даних і посилань.

Головний редактор: Попов О. О. – член-кор. НАН України, д-р техн. наук, професор, с.н.с., в.о. директора Центру інформаційно-аналітичного та технічного забезпечення моніторингу об'єктів атомної енергетики Національної академії наук України

Редакційна колегія:

Василенко М. Д. – д-р фіз.-мат. наук, проф., професор кафедри кібербезпеки, Національний університет «Одеська юридична академія»; **Горбов І. В.** – канд. техн. наук, с.н.с., старший науковий співробітник, Інститут проблем реєстрації інформації НАН України; **Дуднік А. С.** – д-р техн. наук, доц., доцент кафедри мережевих та інтернет технологій, Київський національний університет імені Тараса Шевченка; **Євсєєв С. П.** – д-р техн. наук, професор кафедри кібербезпеки та інформаційних технологій, Харківський національний економічний університет імені Семена Кузнеця; **Зибін С. В.** – д-р техн. наук, доц., завідувач кафедри інженерії програмного забезпечення, Національний авіаційний університет; **Кавун С. В.** – д-р екон. наук, канд. техн. наук, проф., завідувач кафедри комп'ютерних інформаційних систем та технологій, Міжрегіональна Академія управління персоналом; **Комарова Л. О.** – д-р техн. наук, с.н.с., директор Навчально-наукового інституту інформаційної безпеки та стратегічних комунікацій, Національна академія Служби безпеки України; **Мілов О. В.** – д-р техн. наук, професор кафедри кібербезпеки та інформаційних технологій, Харківський національний економічний університет імені Семена Кузнеця; **Охріменко Т. О.** – канд. техн. наук, старший науковий співробітник науково-дослідної лабораторії протидії кіберзагрозам в авіаційній галузі, Національний авіаційний університет; **Рудніченко М. Д.** – канд. техн. наук, доц., доцент кафедри інформаційних технологій, Державний університет «Одеська політехніка»; **Скुरатовський Р. В.** – канд. фіз.-мат. наук, доц., доцент кафедри обчислювальної математики та комп'ютерного моделювання, Міжрегіональна Академія управління персоналом; **Супрун О. М.** – канд. фіз.-мат. наук, доц., доцент кафедри програмних систем і технологій, Київський національний університет імені Тараса Шевченка; **Табунщик Г. В.** – канд. техн. наук, проф., професор кафедри програмних засобів, Національний університет «Запорізька політехніка»; **Фомін О. О.** – д-р техн. наук, доц., професор кафедри комп'ютеризованих систем управління, професор кафедри прикладної математики та інформаційних технологій, Державний університет «Одеська політехніка»; **Хохлячова Ю. Є.** – канд. техн. наук, доц., доцент кафедри безпеки інформаційних технологій, Національний авіаційний університет; **Чолишкіна О. Г.** – канд. техн. наук, доц., директор Інституту комп'ютерно-інформаційних технологій та дизайну, Міжрегіональна Академія управління персоналом; **Чорний О. П.** – доктор технічних наук, професор, директор Навчально-наукового інституту електричної інженерії та інформаційних технологій, Кременчуцький національний університет імені Михайла Остроградського; **Юдін О. К.** – д-р техн. наук, проф., директор центру кібербезпеки Навчально-наукового інституту інформаційної безпеки та стратегічних комунікацій, Національна академія Служби безпеки України; **Гопесенко Віктор** – dr. sc. ing., проф., проректор з наукової роботи, директор навчальної програми магістратури «Комп'ютерні системи», Університет прикладних наук ISMA (Латвійська Республіка); **Leszczyna Rafal** – dr hab. inż., професор кафедри комп'ютерних наук у менеджменті, Гданський технологічний університет (Республіка Польща).

*Свідомство про державну реєстрацію друкованого засобу масової інформації
«Інформаційні технології та суспільство» Серія KB № 24815-14755P від 27.04.2021 р.*

Відповідно до Наказу МОН України № 1290 від 30 листопада 2021 року (додаток 3) журнал включено до Переліку наукових фахових видань України (категорія Б) зі спеціальностей 121 – Інженерія програмного забезпечення, 122 – Комп'ютерні науки, 123 – Комп'ютерна інженерія, 124 – Системний аналіз, 125 – Кібербезпека, 126 – Інформаційні системи та технології.

Усі електронні версії статей журналу оприлюднюються на офіційній сторінці видання
<http://journals.maup.com.ua/index.php/it>

Статті у виданні перевірені на наявність плагіату за допомогою програмного забезпечення
StrikePlagiarism.com від польської компанії Plagiat.pl.

*Recommended for publication
by Interregional Academy of Personnel Management
(Minutes No. 10 dated 29 November 2023)*

Information Technology and Society / [chief editor Oleksandr Popov]. – Kyiv : Interregional Academy of Personnel Management, 2023. – Issue 4 (10). – 32 p.

Journal «Information Technology and Society» is a peer-reviewed scientific edition, which publishes materials of scientists of various levels in the form of scientific articles for the purpose of their dissemination both among domestic researchers and abroad.

Editorial board do not necessarily reflect the position expressed by the authors of articles, and are not responsible for the accuracy of the data and references.

Chief editor: Oleksandr Popov – Corresponding Member of NAS of Ukraine, Doctor of Engineering, Professor, Senior Research Scientist, Acting Director of the Center for Information-Analytical and Technical Support of Nuclear Power Facilities Monitoring of the National Academy of Sciences of Ukraine

Editorial Board:

Mykola Vasylenko – Doctor of Physics and Mathematics, Professor, Professor at the Department of Cybersecurity, National University «Odesa Law Academy»; **Ivan Horbov** – PhD in Engineering, Senior Research Associate, Senior Research Fellow, Institute for Information Recording of NAS of Ukraine; **Andrii Dudnik** – Doctor of Engineering, Associate Professor, Senior Lecturer at the Department of Networking and Internet Technologies, Taras Shevchenko National University of Kyiv; **Serhii Yevseiev** – Doctor of Engineering, Professor at the Department of Cybersecurity and Information Technologies, Simon Kuznets Kharkiv National University of Economics; **Serhii Zybin** – Doctor of Engineering, Associate Professor, Head of the Department of Software Engineering, National Aviation University; **Serhii Kavun** – Doctor of Economics, PhD in Engineering, Professor, Head of the Department of Computer Information Systems and Technologies Interregional Academy of Personnel Management; **Larysa Komarova** – Doctor of Engineering, Senior Research Scientist, Laureate of State Prize, Director of Educational-Scientific Institute of Information Security and Strategic Communications, National Academy of the Security Service of Ukraine; **Oleksandr Milov** – Doctor of Engineering, Professor at the Department of Cybersecurity and Information Technologies, Simon Kuznets Kharkiv National University of Economics; **Tetiana Okhrimenko** – PhD in Engineering, Senior Research Scientist at the Scientific Research Laboratory for Countering Aviation Cyberthreats, National Aviation University; **Mykola Rudnichenko** – PhD in Engineering, Associate Professor, Senior Lecturer at the Department of Information Technologies, Odessa Polytechnic State University; **Ruslan Skuratovskiy** – PhD in Physics and Mathematics, Associate Professor, Senior Lecturer at the Department of Computational Mathematics and Computer Modeling, Interregional Academy of Personnel Management; **Olha Suprun** – PhD in Physics and Mathematics, Associate Professor, Senior Lecturer at the Department of Software Systems and Technologies, Taras Shevchenko National University of Kyiv; **Halyna Tabunshchyk** – PhD in Engineering, Professor, Professor at the Department of Software Tools, “Zaporizhzhia Polytechnic” National university; **Oleksandr Fomin** – Doctor of Engineering, Associate Professor, Professor at the Department of Computerized Control Systems, Professor at the Department of Applied Mathematics and Information Technologies, Odessa Polytechnic State University; **Yuliia Khokhlachova** – PhD in Engineering, Associate Professor, Senior Lecturer at the Department of Information Technology Security, National Aviation University; **Olha Cholyshkina** – PhD in Engineering, Associate Professor, Director of the Institute of Computer Information Technologies and Design, Interregional Academy of Personnel Management; **Oleksii Chornyi** – Doctor of Technical Sciences, Professor, Director of the Educational and Scientific Institute of Electrical Engineering and Information Technologies, Kremenchuk National University named after Mykhailo Ostrogradskyi; **Oleksandr Yudin** – Doctor of Engineering, Professor, Director of the Cybersecurity Center of the Educational-Scientific Institute of Information Security and Strategic Communications, National Academy of the Security Service of Ukraine; **Hopeienko Viktor** – dr. sc. ing., Professor, Vice Rector for Research, Director of the study programme “Computer systems”, ISMA University of Applied Sciences (Republic of Latvia); **Leszczyna Rafal** – dr hab. inż., Profesor, Katedra Informatyki w Zarządzaniu, Politechnika Gdańska (Republic of Poland).

*Print media registration certificate «Information Technology and Society»
series KV No. 24815-14755P dated 27.04.2021*

According to the Decree of MES No. 1290 (Annex 3) dated November 30, 2021, the journal was included in the List of scientific professional publications of Ukraine (category B) in specialties 121 – Software engineering, 122 – Computer sciences, 123 – Computer engineering, 124 – Systems analysis, 125 – Cybersecurity, 126 – Information systems and technologies.

All electronic versions of articles in the collection are available on the official website edition
<http://journals.maup.com.ua/index.php/it>

The articles were checked for plagiarism using the software
StrikePlagiarism.com developed by the Polish company Plagiat.pl.

© Interregional Academy of Personnel Management, 2023
© Copyright by the contributors, 2023

ЗМІСТ

Едуард ЖАРІКОВ, Володимир ЖНАКІН

УДОСКОНАЛЕНИЙ ГЕОМЕТРИЧНИЙ АЛГОРИТМ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

В ЗАДАЧАХ ОПТИМІЗАЦІЇ ТРАНСПОРТНИХ МАРШРУТІВ6

Nataliia KOTENKO, Tetyana ZHYROVA

ANALYSIS OF ARTIFICIAL INTELLIGENCE TECHNOLOGIES

TO OPTIMISE INCLUSIVE PRACTICES IN HIGHER EDUCATION.....15

Юлія ПАРФЕНЕНКО, Денис ЮРЧЕНКО, Дмитро ТРОЦЕНКО

РОЗРОБЛЕННЯ ВЕБДОДАТКУ ДЛЯ ОРГАНІЗАЦІЇ ЗАМОВЛЕННЯ СИСТЕМ «РОЗУМНИЙ БУДИНОК»24

CONTENTS

Eduard ZHARIKOV, Volodymyr ZHNAKIN

AN IMPROVED GEOMETRIC ALGORITHM AND SOFTWARE IN TRANSPORT ROUTE
OPTIMIZATION PROBLEMS.....6

Nataliia KOTENKO, Tetyana ZHYROVA

ANALYSIS OF ARTIFICIAL INTELLIGENCE TECHNOLOGIES
TO OPTIMISE INCLUSIVE PRACTICES IN HIGHER EDUCATION.....15

Yuliia PARFENENKO, Denys YURCHENKO, Dmytro TROTSSENKO

DEVELOPMENT OF A WEB APPLICATION FOR SMART HOME SYSTEMS ORDERING.....24

УДК 004.42

DOI <https://doi.org/10.32689/maup.it.2023.4.1>

Едуард ЖАРИКОВ

доктор технічних наук, професор, завідувач кафедри інформатики та програмної інженерії, Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", Берестейський проспект (Перемоги), 37, Київ, Україна, індекс 03056 (zharikov.edu@i.ua)

ORCID: 0000-0003-1811-9336

Володимир ЖНАКІН

магістрант кафедри інформатики та програмної інженерії, Національний технічний університет України "Київський політехнічний інститут імені Ігоря Сікорського", Берестейський проспект (Перемоги), 37, Київ, Україна, індекс 03056 (vovancom98@gmail.com)

ORCID: 0009-0003-3074-4748

Eduard ZHARIKOV

Doctor of Technical Sciences, Professor, Head of the Department of Informatics and Software Engineering of National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37, Beresteysky (Peremohy) Ave, Kyiv, Ukraine, postal code 03056 (zharikov.edu@i.ua)

Volodymyr ZHNAKIN

Master's Student of the Department of Informatics and Software Engineering of National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37, Beresteysky (Peremohy) Ave, Kyiv, Ukraine, postal code 03056 (vovancom98@gmail.com)

Бібліографічний опис статті: Жаріков, Е., Жнакін, В. (2023). Удосконалений геометричний алгоритм та програмне забезпечення в задачах оптимізації транспортних маршрутів. *Інформаційні технології та суспільство*, DOI: <https://doi.org/10.32689/maup.it.2023.4.1>

Bibliographic description of the article: Zharikov, E., Zhnakin, V. (2023). Udoskonalenyi heometrychnyi alhorytm ta proqramne zabezpechennia v zadachakh optymizatsii transportnykh marshrutiv [Improved geometric algorithm and software in transport route optimization problems]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, DOI: <https://doi.org/10.32689/maup.it.2023.4.1>

УДОСКОНАЛЕНИЙ ГЕОМЕТРИЧНИЙ АЛГОРИТМ ТА ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ В ЗАДАЧАХ ОПТИМІЗАЦІЇ ТРАНСПОРТНИХ МАРШРУТІВ

Анотація. Сучасні системи керування транспортними маршрутами потребують розроблення програмного забезпечення, що реалізує більш точні та швидкі алгоритми розв'язання задачі комівояжера, які забезпечують для великої кількості точок пошук найкращого маршруту з порівняно невеликою похибкою за короткий час. Основними недоліками існуючих реалізацій є стохастичність, обмежена адаптивність до параметрів задачі та велика чутливість до початкових умов, що призводить до невірних рішень та непотрібних витрат ресурсів. У статті наведена постановка задачі пошуку найменшого можливого циклічного маршруту, що проходить через заданий набір міст, починаючи і закінчуючи в тому самому місті. У результаті, алгоритм повинен знайти послідовність відвідування міст, щоб загальна довжина шляху між ними була мінімальною, і шлях проходив через кожне місто рівно один раз. Проведено експериментальні дослідження та порівняно ефективність удосконаленого геометричного алгоритму для 10, 50, 100 та 1000 міст. Результати роботи удосконаленого геометричного алгоритму для 10 міст порівняно з базовим геометричним алгоритмом та алгоритмом повного перебору, щоб переконатися, що є покращення не тільки часу пошуку маршруту, але й довжини знайденого шляху. Для 50, 100 1000 міст результати роботи удосконаленого геометричного алгоритму порівнюємо з базовим геометричним алгоритмом, генетичним алгоритмом і алгоритмом оптимізації мурашиної колонії, щоб переконатися, що удосконалений геометричний алгоритм працює не гірше і швидше ніж інші алгоритми. У випадках, коли удосконалений геометричний алгоритм працює з похибкою відносно інших алгоритмів, він дозволяє отримати суттєвий вииграш за часом виконання. Експерименти показали, що запропонований у роботі удосконалений геометричний алгоритм дозволяє детерміновано знаходити найкращий маршрут з похибкою 6,82% для 1000 міст. При цьому він знаходить рішення у 2,8 рази швидше, ніж алгоритм мурашиних колоній, та у 265 раз швидше, ніж генетичний алгоритм. Наведено приклад реалізації удосконаленого геометричного алгоритму мовою програмування swift для використання на платформі iOS.

Ключові слова: комбінаторна оптимізація, задача комівояжера, оптимальний маршрут, час виконання, точність.

AN IMPROVED GEOMETRIC ALGORITHM AND SOFTWARE IN TRANSPORT ROUTE OPTIMIZATION PROBLEMS

Abstract. Modern transport route management systems require the development of software that implements more accurate and faster algorithms for solving the traveling salesman problem, which provide for a large number of points the search for the best route with a relatively small error in a shorter time.

The main disadvantages of the existing implementations are stochasticity, limited adaptability to the parameters of the problem and high sensitivity to the initial conditions, which leads to incorrect decisions and loss of resources. The article presents the formulation of the problem of finding the smallest possible circular route passing through a given set of cities, starting and ending in the same city. As a result, the algorithm must find a sequence of visiting cities so that the total length of the path between them is minimal, and the path passes through each city exactly once. Experimental studies were conducted, and the efficiency of the improved geometric algorithm was compared for 10, 50, 100 and 1000 cities. The performance of the improved geometric algorithm for 10 cities is compared to the basic geometric algorithm and the brute force algorithm to verify that there is an improvement not only in the route search time but also in the length of the path found. For 50, 100, 1000 cities, the results of the improved geometric algorithm will be compared with the basic geometric algorithm, the genetic algorithm, and the ant colony optimization algorithm to make sure that the improved geometric algorithm works no worse and faster than other algorithms. In cases where the improved geometric algorithm works with an error relative to other algorithms, it allows to obtain a significant gain in execution time. Experiments show that the improved geometric algorithm proposed in the article allows deterministically finding the best route with an error of 6.82% for 1000 cities. At the same time, it finds solutions 2.8 times faster than the ant colony optimization algorithm and 265 times faster than the genetic algorithm. The example of implementation of the improved geometric algorithm in the swift programming language for use on the iOS platform is given.

Key words: combinatorial optimization, traveling salesman problem, optimal route, execution time, accuracy.

Вступ

У сучасному світі, де ефективне управління маршрутами та оптимізація шляхів мають ключове значення для багатьох галузей, задача комівояжера [1] залишається однією з найважливіших. Необхідність знаходження оптимального маршруту для відвідування набору заданих точок є особливо актуальною в логістиці.

Проте, існуючі реалізації програмного забезпечення для розв'язання задачі комівояжера [2] часто стикаються з рядом проблем, які обмежують їхню ефективність та точність. Однією з основних проблем є недостатня точність наявних алгоритмів, що використовуються для розрахунку маршруту через велику кількість точок. Це може призводити до невірних рішень та непотрібних витрат ресурсів.

Запропоноване розв'язання задачі оптимізації пошуку транспортного маршруту спрямоване на усунення недоліків існуючих підходів у спосіб розробки програмного забезпечення на основі удосконаленого геометричного алгоритму. Основними недоліками існуючих реалізацій є обмежена адаптивність до різноманітних параметрів та велика чутливість до початкових умов [3]. Запропонований авторами геометричний алгоритм здатний ефективно пристосовуватися до різних умов та забезпечує високу точність у визначенні оптимального за певним критерієм маршруту.

Хоча запропоновано багато методів та алгоритмів розв'язання задачі комівояжера [4], її складність та важливість залишаються актуальними викликами для дослідників та розробників відповідного програмного забезпечення. Враховуючи ці виклики, запропоноване програмне забезпечення на основі нового геометричного алгоритму має на меті подолати недоліки та обмеження, що існують у існуючих підходах. За рахунок використання вдосконалених геометричних методів та покращених стратегій оптимізації [5], у цій роботі досягнуто значного покращення у точності та швидкості розв'язання задачі комівояжера, що відкриває нові перспективи для практичного використання запропонованого удосконаленого геометричного алгоритму в реальних умовах.

1. Постановка задачі

Задача комівояжера (англ. Traveling Salesman Problem, TSP) являє собою задачу комбінаторної оптимізації, в якій необхідно знайти найменший можливий циклічний маршрут, що проходить через заданий набір міст (точок), починаючи і закінчуючи в тому самому місті. Метою розв'язку задачі є пошук такої послідовності відвідування міст, щоб загальна довжина шляху між ними була мінімальною, і шлях проходив через кожне місто рівно один раз.

Визначення задачі комівояжера включає:

- **набір міст (точок):** вхідними даними є множина міст, кожне з яких представлене з координатами або відстанями від інших міст. Міста можуть мати різні відстані між собою [6];
- **цільову функцію:** метою є знаходження такої послідовності обходу міст, щоб сума відстаней між цими містами була мінімальною [7];
- **обмеження:** кожне місто має бути відвідане рівно один раз, і маршрут повинен починатися і закінчуватися в тому самому місті [8];
- **симетричність:** відстань від міста А до В дорівнює відстані від В до А [9].

Не зважаючи, що задача комівояжера залишається однією з найвідоміших і досліджених у галузі комбінаторної оптимізації [10], її розв'язання для великої кількості міст все ще має велике практичне значення у різних галузях, від планування маршрутів вантажівок та обслуговування клієнтів, до оптимізації виробничих процесів. У зв'язку з цим розробка ефективних методів розв'язання задачі комівояжера залишається актуальним завданням і є об'єктом постійного дослідження та розробки покращених алгоритмів.

2. Існуючі методи розв'язання задачі комівояжера

Існує декілька традиційних алгоритмів для знаходження оптимального маршруту у задачі комівояжера. Найвідоміші з них – це метод найближчого сусіда [11], метод гілок і меж [12], алгоритм мурашиної колонії [13-15] та генетичні алгоритми [16-18]. Алгоритм, який точно знаходить найкоротший маршрут відвідування всіх міст – це алгоритм повного перебору [19], однак він потребує значного часу роботи при кількості міст більше п'ятнадцяти, що ставить під сумнів його практичне застосування у сучасному програмному забезпеченні.

Прості алгоритми знаходження оптимального рішення, такі як метод найближчого сусіда та метод гілок і меж мають масу окремих випадків, де їх застосування покаже один з найгірших варіантів для побудови найкоротшого маршруту. І лише генетичні і мурашині алгоритми дозволяють отримати порівняно хороші результати як у довжині найкращого маршруту, так і за часом пошуку цього маршруту.

Тому показники роботи запропонованого геометричного алгоритму розв'язання задачі комівояжера будемо порівнювати з:

1) алгоритмом повного перебору, щоб визначити, чи може геометричний алгоритм знаходити найкоротші шляхи обходу для кількості міст не більше 15 і робити це значно швидше;

2) генетичним і мурашиним алгоритмами, щоб визначити, чи може геометричний алгоритм знаходити оптимальні маршрути не гірше за маршрути, отримані цими алгоритмами і робити це швидше, або знаходити маршрути достатньо близькі до оптимальних, але робити це суттєво швидше.

3. Принцип роботи базового геометричного алгоритму

Геометричний алгоритм полягає у пошуку найкоротшої відстані між точками, вирішуючи завдання обчислення периметра n -кутника, використовуючи такі принципи:

1) необхідно уникати можливих перетинів маршруту [20];

2) необхідно рухатися від однієї точки до іншої, використовуючи найкоротшу відстань від обраної наступної точки до прямої, яка вже побудована через дві точки, що передували їй;

3) кути, які виходять шляхом додавання кожної нової точки повинні бути якнайменш гострими і якнайменш розгорнутими, тобто всіма силами прагнути до кута 180 градусів.

Таким чином, кількість можливих варіантів побудови маршруту k можна обчислити за формулою

$$k = \frac{1}{6} (3^{n-1} + 3),$$

де n – кількість точок або міст, які потрібно відвідати, тобто алгоритмічна та часова складність запропонованого геометричного алгоритму експоненційно зростатиме. На рисунку 1 представлена прогресія можливих варіантів маршрутів. У такому стані геометричний алгоритм хоч і буде показувати себе значно краще, ніж алгоритм повного перебору, проте він буде виконуватися дуже довго для кількості точок більше 50, і вже не зможе конкурувати з генетичним або мурашиним алгоритмами. Тому виникає необхідність оптимізувати геометричний алгоритм для роботи з великою кількістю міст.

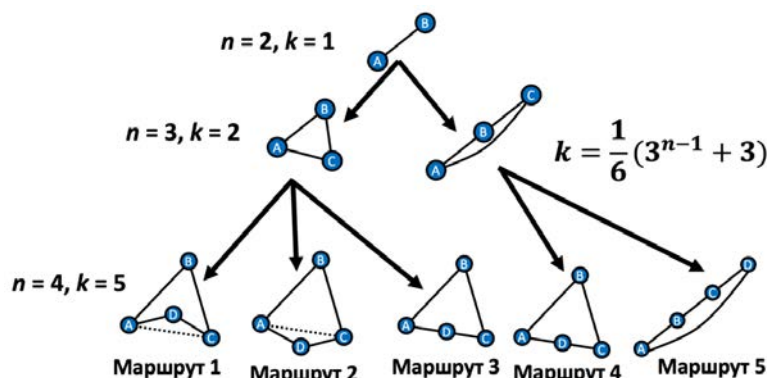


Рис. 1. Прогресія варіантів фігур в залежності від кількості міст

4. Удосконалення геометричного алгоритму

Як бачимо з рисунка 1, для $n = 4$ є п'ять варіантів можливих побудов маршрутів, при чому маршрути №3 та №4 є цілком ідентичними за визначенням. І це маршрути, в яких четверта точка збігається з однією з існуючих прямих. Тоді розпишемо загальну закономірність, як відбувається додавання кожної наступної точки до вже побудованих фігур. Як бачимо з рисунка 1, для двох точок є один варіант обходу – і це пряма. Для трьох точок існує два варіанти обходу, залежно від їхнього взаємного розташування: якщо всі три точки лежать на одній прямій, то ця фігура – це пряма; а якщо одна з точок не лежить на прямій, де знаходяться дві інші – це трикутник. Отже, **першою умовою** для перевірки буде – лежить (або не лежить) наступна точка на існуючій прямій.

Для чотирьох точок розглянемо можливі одержані фігури детально. Якщо до цього три точки лежали на одній прямій, то четверта точка може також лежати на цій прямій, або бути поза нею. А якщо до цього три точки не лежали на одній прямій, то четверта точка може або належати одній з отриманих прямих, або лежати поза отриманим трикутником, або лежати всередині отриманого трикутника. Таким чином, точка може лежати на отриманій фігурі, бути всередині неї або лежати за межами отриманої фігури – і це буде **другою умовою** перевірки для удосконалення геометричного алгоритму.

Нехай 0 – позначення маршруту, що являє собою пряму лінію, 1 – позначення маршруту, який є трикутником. Тоді для позначення нової фігури після додавання нової точки введемо такі позначення: + (плюс) – точка лежить за межами отриманої раніше фігури; – (мінус) – точка лежить усередині отриманої фігури; для позначення фігури, де нова додана точка лежить на одній із вже побудованих прямих не додаватимемо ні + (плюс) ні – (мінус) до позначення нової фігури. Тоді нова прогресія варіантів маршрутів буде виглядати так, як показано на рисунку 2.

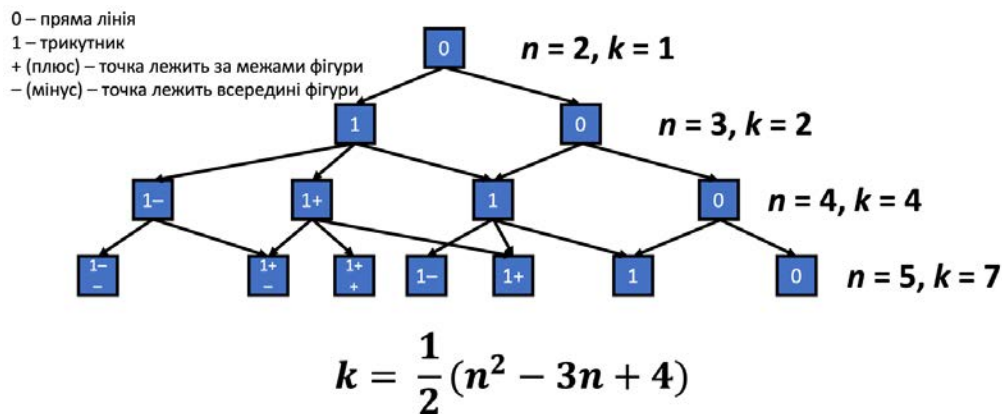


Рис. 2. Прогресія варіантів фігур у залежності від кількості міст для удосконаленого геометричного алгоритму

Отже, кількість варіантів маршрутів для удосконаленого геометричного алгоритму можна визначити за формулою

$$k = \frac{1}{2}(n^2 - 3n + 4),$$

де n – кількість точок або міст, які необхідно відвідати. Отже, алгоритмічна і часова складність удосконаленого алгоритму буде слідувати квадратичній функції, що є значним поліпшенням у порівнянні з базовою версією геометричного алгоритму, оскільки тепер алгоритм зможе виконати обчислення найкращого маршруту для великої кількості міст, наприклад, $n > 1000$.

На рисунку 3 наведено код мовою програмування swift для реалізації удосконаленого геометричного алгоритму. Вибір мови програмування обумовлений необхідністю використання платформи iOS для реалізації програмного забезпечення пошуку найкращого маршруту серед заданих точок.

```
// Функція для перевірки, чи точка лежить на прямій
func pointOnLine(point: Point, line: Line) -> Bool {
    return line.A * point.x + line.B * point.y + line.C == 0
}
// Структура для представлення прямокутника
struct Rectangle {
    var left: Double
    var right: Double
    var top: Double
    var bottom: Double
}
// Функція для перевірки, чи точка знаходиться всередині, на межі або поза межами прямокутника
func pointInRectangle(point: Point, rectangle: Rectangle) -> String {
    if point.x >= rectangle.left && point.x <= rectangle.right &&
        point.y >= rectangle.top && point.y <= rectangle.bottom {
        return "Точка знаходиться всередині прямокутника"
    } else if point.x == rectangle.left || point.x == rectangle.right ||
        point.y == rectangle.top || point.y == rectangle.bottom {
        return "Точка лежить на межі прямокутника"
    } else {
        return "Точка знаходиться поза межами прямокутника"
    }
}
// Приклад використання функцій
let point = Point(x: 2.0, y: 3.0)
let line = Line(A: 1.0, B: -1.0, C: -1.0)
let rectangle = Rectangle(left: 1.0, right: 5.0, top: 2.0, bottom: 4.0)

let isPointOnLine = pointOnLine(point: point, line: line)
let pointStatusInRectangle = pointInRectangle(point: point, rectangle: rectangle)

print("Перевірка прямої: \(isPointOnLine ? "Точка лежить на прямій" : "Точка не лежить на прямій")")
print("Перевірка прямокутника: \(pointStatusInRectangle)")
```

Рис. 3. Код swift реалізації удосконаленого геометричного алгоритму

5. Оцінка ефективності удосконаленого геометричного алгоритму та експериментальні дослідження. Проводити експериментальні дослідження та порівнювати ефективність удосконаленого геометричного алгоритму будемо за такою схемою постановки експерименту.

Для $n=10$ порівняємо результати роботи удосконаленого геометричного алгоритму з базовим геометричним алгоритмом та алгоритмом повного перебору, щоб переконатися, що є покращення не тільки часу пошуку маршруту, але й довжини знайденого шляху.

Для $n=50$, $n=100$ та $n=1000$ порівняємо результати роботи удосконаленого геометричного алгоритму з базовим геометричним алгоритмом, генетичним алгоритмом і мурашиним алгоритмом, щоб переконатися, що удосконалений геометричний алгоритм працює не гірше і швидше ніж інші алгоритми, або новий геометричний алгоритм працює з невеликою похибкою відносно інших алгоритмів, але дає суттєвий виграш за часом виконання. Так як генетичний і мурашиний алгоритм є випадковими алгоритмами, то для отримання показників їх роботи будемо виконувати 15 запусків і обчислювати середнє значення часу виконання та довжини маршруту.

Похибку будемо обчислювати відносно найкращого середнього значення. Координати точок, що є вхідними даними для усіх чотирьох алгоритмів, будемо генерувати випадково. Відстань між отриманими точками маршруту будемо вимірювати в умовних одиницях, а час виконання алгоритму в секундах.

На рисунку 4 показано результати для порівняння роботи алгоритму повного перебору, базового геометричного алгоритму і удосконаленого геометричного алгоритму.

Кількість міст: 10	
Алгоритм повного перебору	
Отриманий шлях = 1897.1	
Час виконання = 0.36288 секунд	
Базовий геометричний алгоритм	
Отриманий шлях = 1897.1	
Час виконання = 0.003281 секунд	
Покращений геометричний алгоритм	
Отриманий шлях = 1897.1	
Час виконання = 0.000037 секунд	
Найкращий шлях = 1897.1	

Рис. 4. Результати роботи алгоритму повного перебору, базового геометричного алгоритму і удосконаленого геометричного алгоритму

На рисунку 5 показано результати роботи програмного забезпечення для порівняння удосконаленого геометричного алгоритму, генетичного алгоритму і мурашиного алгоритму для $n=50$. Для генетичного алгоритму у цьому та подальших експериментах взято стартову популяцію в 1000 осіб і кількість поколінь 100.

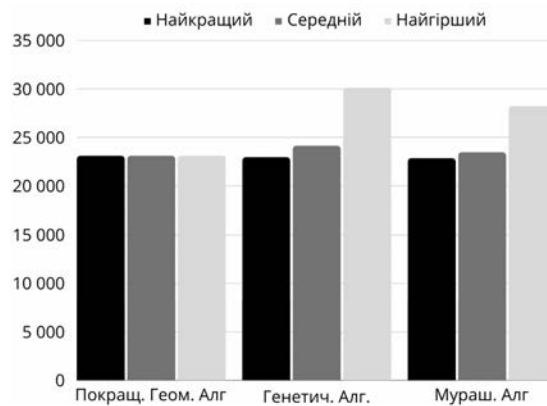


Рис. 5. Довжина отриманого маршруту (в умовних одиницях) для удосконаленого геометричного алгоритму, генетичного алгоритму і мурашиного алгоритму для $n=50$

На рисунку 6 показано результати роботи програмного забезпечення для порівняння удосконаленого геометричного алгоритму, генетичного алгоритму і мурашиного алгоритму для $n=100$.

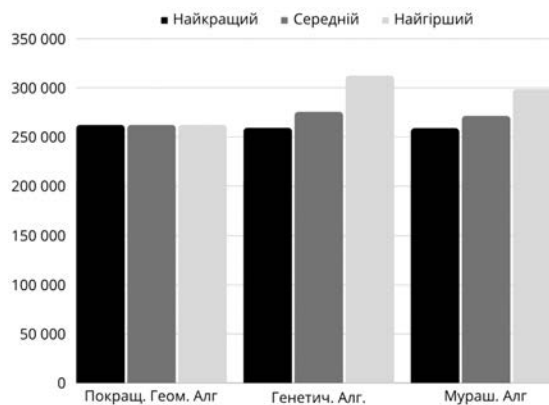


Рис. 6. Довжина отриманого маршруту (в умовних одиницях) для удосконаленого геометричного алгоритму, генетичного алгоритму і мурашиного алгоритму для $n=100$

На рисунку 7 показано результати роботи програмного забезпечення для порівняння удосконаленого геометричного алгоритму, генетичного алгоритму і мурашиного алгоритму для $n=1000$.

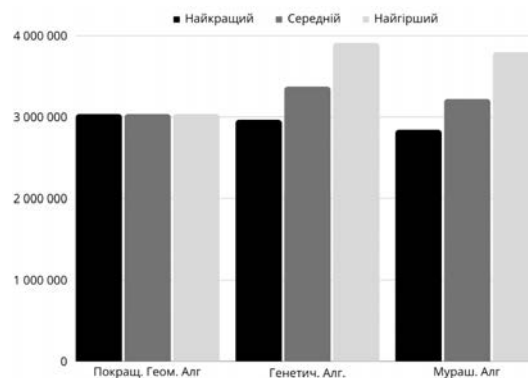


Рис. 7. Довжина отриманого маршруту (в умовних одиницях) для удосконаленого геометричного алгоритму, генетичного алгоритму і мурашиного алгоритму для $n=1000$

Порівняльні результати роботи досліджуваних алгоритмів показані на рисунку 8.

	Алгоритм повного перебору			Базовий геометричний алгоритм			Покращений геометричний алгоритм		
	час виконання, с.	отриманий шлях	похибка	час виконання, с.	отриманий шлях	похибка	час виконання, с.	отриманий шлях	похибка
n = 10	0,36288	1897,1	0%	0,003281	1897,1	0%	0,000037	1897,1	0%
Найкращий шлях = 1897,1									
	Генетичний алгоритм			Мурашиний алгоритм			Покращений геометричний алгоритм		
	час виконання, с.	отриманий шлях	похибка	час виконання, с.	отриманий шлях	похибка	час виконання, с.	отриманий шлях	похибка
n = 50	5,283	24137,4	5,57%	0,002501	23477,2	2,68%	0,001177	23107,5	1,06%
Найкращий шлях = 22864,8									
	Генетичний алгоритм			Мурашиний алгоритм			Покращений геометричний алгоритм		
	час виконання, с.	отриманий шлях	похибка	час виконання, с.	отриманий шлях	похибка	час виконання, с.	отриманий шлях	похибка
n = 100	11,297	275662,8	6,45%	0,012378	271538,2	4,85%	0,004852	262310,1	1,29%
Найкращий шлях = 258 965,7									
	Генетичний алгоритм			Мурашиний алгоритм			Покращений геометричний алгоритм		
	час виконання, с.	отриманий шлях	похибка	час виконання, с.	отриманий шлях	похибка	час виконання, с.	отриманий шлях	похибка
n = 1000	132,763	3375128,3	18,69%	1,408139	3222648,7	13,33%	0,498502	3037615,3	6,82%
Найкращий шлях = 2843625,4									

Рис. 8. Порівняння часу виконання, довжини отриманих маршрутів і похибок для досліджуваних алгоритмів (середні значення п'ятнадцяти запусків)

На рисунку 9 зображено графік залежності часу виконання програми у залежності від кількості міст для базового геометричного алгоритму, удосконаленого геометричного алгоритму, генетичного алгоритму і мурашиного алгоритму.

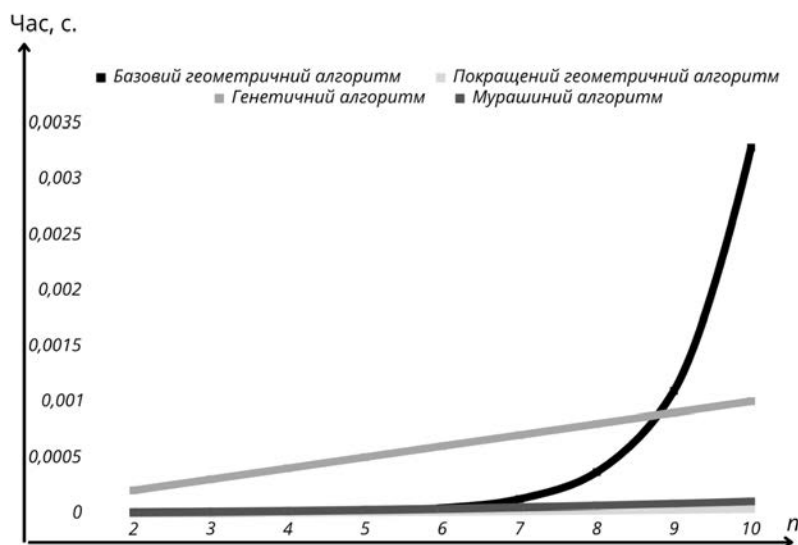


Рис. 9. Графік залежності часу виконання програми у залежності від кількості міст

Як бачимо з рисунків, наведених вище, удосконалений геометричний алгоритм працює значно швидше базового геометричного алгоритму, при чому зберігає і його точність обчислень оптимального шляху. Базовий геометричний алгоритм мав експоненційну часову складність, хоч і меншу, ніж алгоритм повного перебору. Натомість в удосконаленому геометричному алгоритмі часова складність має квадратичну складність, що дозволяє обчислювати маршрути для великої кількості міст. Також, як бачимо з рисунка 9, удосконалений геометричний алгоритм має витрати часу на обчислення нижче ніж одні з кращих варіантів генетичних і мурашиних алгоритмів за досить низької похибки обчислень.

Висновки

У статті запропонований удосконалений геометричний алгоритм розв'язання задачі комівояжера та програмне забезпечення його реалізації на платформі iOS. Удосконалений геометричний алгоритм дозволяє отримати кращі показники ефективності роботи порівняно з класичними методами розв'язання

задачі комівояжера. Отримані результати дозволяють стверджувати, що якісні показники роботи удосконаленого геометричного алгоритму мають тенденцію до поліпшення зі збільшенням кількості міст у порівнянні з мурашиним та генетичним алгоритмами. Він дозволяє детерміновано знаходити найкращий маршрут з порівняно невеликою похибкою (6,82% для 1000 міст) за короткий час, ніж алгоритм мурашиних колоній (у 2,8 рази швидший) та генетичний алгоритм (у 265 раз швидший), що робить його швидким і невибагливим до ресурсів інструментом для пошуку оптимальних маршрутних планів.

Програмне забезпечення, розроблене з урахуванням даного алгоритму, може застосовуватися у різних галузях, де оптимізація маршрутів має критичне значення. Запропонований алгоритм не тільки покращить продуктивність та ефективність логістичних процесів, а й відкриє нові можливості для практичного застосування задачі комівояжера.

Список використаних джерел:

1. Lawler, E. L., et al. The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization. Wiley. 1985. P. 25-31.
2. Monnot, J., & Toulouse, S. The traveling salesman problem and its variations. Paradigms of combinatorial optimization: problems and new approaches. 2014. P. 173-214.
3. Chandra, A., & Naro, A. A comparative study of metaheuristics methods for solving traveling salesman problem. International Journal of Information Science and Technology. 2022. №6(2), P.1-7.
4. Zhang, C., & Sun, P. Heuristic Methods for Solving the Traveling Salesman Problem (TSP): A Comparative Study. In 2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC). 2023. P. 1-6. IEEE.
5. Reinelt, G. The Traveling Salesman: Computational Solutions for TSP Applications. Springer. 1994. P. 18-21.
6. Toth, P., et al. The Vehicle Routing Problem. SIAM Monographs on Discrete Mathematics and Applications. SIAM. 2002. P. 39-41.
7. Nagata, Y. Recent developments in combinatorial optimization for the traveling salesman problem: A survey. European Journal of Operational Research. 2011. №213(1). P. 1-10.
8. Vansteenwegen, P., et al. The Traveling Salesman Problem: A Case Study in Local Optimization. Operations Research Perspectives. 2011. P. 65-77.
9. Gutin, G., Punnen, A. The traveling salesman problem and its variations. Springer Science & Business Media, 2006.
10. Jiang, C., Wan, Z., & Peng, Z. A new efficient hybrid algorithm for large scale multiple traveling salesman problems. Expert Systems with Applications. 2020. №139. P. 112867.
11. Cheikhrouhou, O., & Khoufi, I. A comprehensive survey on the Multiple Traveling Salesman Problem: Applications, approaches and taxonomy. Computer Science Review. 2021. №40. P. 100369.
12. Lawler, E. L., et al. The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization. Wiley. 1985. P. 49-52.
13. Dorigo, M., et al. The Ant System: Optimization by a colony of cooperating agents. IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics. 1996. №26(1). P. 29-41.
14. Dorigo, M., et al. Ant Colony Optimization. MIT Press. 2004. P. 16-20.
15. Blum, C., et al. The hyper-cube framework for ant colony optimization. IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics. 2004. №34(2). P. 1161-1172.
16. Holland, J. H. Adaptation in Natural and Artificial Systems. University of Michigan Press. 1975. P. 17-22.
17. Goldberg, D. E. Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley Professional. 1989. P. 24-27.
18. Mitchell, M. An Introduction to Genetic Algorithms. MIT Press. 1998. P. 30-33.
19. Cormen, T. H., et al. Introduction to Algorithms. MIT Press. 2022.
20. An Improvement to the 2-Opt Heuristic Algorithm for Approximation of Optimal TSP Tour. URL: <https://www.mdpi.com/2076-3417/13/12/7339>

References:

1. Lawler, E. L., et al. (1985). The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization. Wiley. P. 25-31.
2. Monnot, J., & Toulouse, S. (2014). The traveling salesman problem and its variations. Paradigms of combinatorial optimization: problems and new approaches. P. 173-214.
3. Chandra, A., & Naro, A. (2022). A comparative study of metaheuristics methods for solving traveling salesman problem. International Journal of Information Science and Technology. №6(2), P.1-7.
4. Zhang, C., & Sun, P. (2023). Heuristic Methods for Solving the Traveling Salesman Problem (TSP): A Comparative Study. In 2023 IEEE 34th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC). P. 1-6. IEEE.
5. Reinelt, G. (1994). The Traveling Salesman: Computational Solutions for TSP Applications. Springer. P. 18-21.
6. Toth, P., et al. (2002). The Vehicle Routing Problem. SIAM Monographs on Discrete Mathematics and Applications. SIAM. P. 39-41.
7. Nagata, Y. (2011). Recent developments in combinatorial optimization for the traveling salesman problem: A survey. European Journal of Operational Research. №213(1). P. 1-10.

8. Vansteenwegen, P., et al. (2011). The Traveling Salesman Problem: A Case Study in Local Optimization. *Operations Research Perspectives*. P. 65-77.
9. Gutin, G., Punnen, A. (2006). The traveling salesman problem and its variations. Springer Science & Business Media.
10. Jiang, C., Wan, Z., & Peng, Z. (2020). A new efficient hybrid algorithm for large scale multiple traveling salesman problems. *Expert Systems with Applications*. №139. P. 112867.
11. Cheikhrouhou, O., & Khoufi, I. (2021). A comprehensive survey on the Multiple Traveling Salesman Problem: Applications, approaches and taxonomy. *Computer Science Review*. №40. P. 100369.
12. Lawler, E. L., et al. (1985). The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization. Wiley. P. 49-52.
13. Dorigo, M., et al. (1996). The Ant System: Optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*. №26(1). P. 29-41.
14. Dorigo, M., et al. (2004). Ant Colony Optimization. MIT Press. P. 16-20.
15. Blum, C., et al. (2004). The hyper-cube framework for ant colony optimization. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*. №34(2). P. 1161-1172.
16. Holland, J. H. (1975). Adaptation in Natural and Artificial Systems. University of Michigan Press. P. 17-22.
17. Goldberg, D. E. (1989). Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley Professional. P. 24-27.
18. Mitchell, M. (1998). An Introduction to Genetic Algorithms. MIT Press. P. 30-33.
19. Cormen, T. H., et al. (2022). Introduction to Algorithms. MIT Press.
20. An Improvement to the 2-Opt Heuristic Algorithm for Approximation of Optimal TSP Tour. Retrieved from <https://www.mdpi.com/2076-3417/13/12/7339>

UDC 004.8

DOI <https://doi.org/10.32689/maup.it.2023.4.2>

Nataliia KOTENKO

Candidate of Pedagogical Sciences, Associate Professor, Associate Professor at the Department of Software Engineering and Cyber Security, State University of Trade and Economics, 19, Kyoto St, Kyiv, Ukraine, postal code 02156 (kotenkono@ukr.net)

Tetyana ZHYROVA

Candidate of Pedagogical Sciences, Associate Professor, Associate Professor at the Department of Software Engineering and Cyber Security, State University of Trade and Economics, 19, Kyoto St, Kyiv, Ukraine, postal code 02156 (zhyrova@knote.edu.ua)

Наталія КОТЕНКО

кандидат педагогічних наук, доцент, доцент кафедри інженерії програмного забезпечення та кібербезпеки, Державний торговельно-економічний університет, вул. Кіото, 19, Київ, Україна, індекс 02156 (kotenkono@ukr.net)

ORCID: 0000-0002-2675-6514

Тетяна ЖИРОВА

кандидат педагогічних наук, доцент, доцент кафедри інженерії програмного забезпечення та кібербезпеки, Державний торговельно-економічний університет, вул. Кіото, 19, Київ, Україна, індекс 02156 (zhyrova@knote.edu.ua)

ORCID: 0000-0001-8321-6939

Bibliographic description of the article: Kotenko, N., Zhyrova, T. (2023). Analiz tekhnolohiy shtuchnoho intelektu dlya optymizatsiyi inklyuzyvnykh praktyk u vyshchii osviti [Analysis of artificial intelligence technologies to optimise inclusive practices in higher education]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, DOI: <https://doi.org/10.32689/maup.it.2023.4.2>

Бібліографічний опис статті: Котенко, Н., Жирова, Т., (2023). Аналіз технологій штучного інтелекту для оптимізації інклюзивних практик у вищій освіті. *Інформаційні технології та суспільство*, DOI: <https://doi.org/10.32689/maup.it.2023.4.2>

ANALYSIS OF ARTIFICIAL INTELLIGENCE TECHNOLOGIES TO OPTIMISE INCLUSIVE PRACTICES IN HIGHER EDUCATION

Abstract. The article analyzes the potential of artificial intelligence (AI) technologies to optimize inclusive practices in higher education. Artificial intelligence opens up new opportunities for adapting the educational process to the individual characteristics of each student, which is especially important for people with special educational needs. The article discusses key aspects of the use of modern AI technologies in the educational process, including adaptive learning systems, voice assistants and text analyzers, and face and gesture recognition technologies. Both global trends and Ukraine's specific experience in implementing AI to support inclusiveness in education are analyzed. In particular, the article focuses on how AI technologies can help develop an educational environment for people with visual and hearing impairments, autism spectrum disorders, ADHD, and other disabilities. In conclusion, the article emphasizes the need for further research and development of innovative approaches to the implementation of artificial intelligence technologies in the educational process to create an inclusive, accessible and effective educational environment. The article contributes to the formation of a new view of the possibilities of integrating inclusive practices and artificial intelligence in the context of higher education.

Key words: Artificial intelligence, inclusive education, higher education, adaptive learning, voice assistants, facial recognition, text analyzers, augmented reality technologies, intelligent systems.

АНАЛІЗ ТЕХНОЛОГІЙ ШТУЧНОГО ІНТЕЛЕКТУ ЩОДО ОПТИМІЗАЦІЇ ІНКЛЮЗИВНИХ ПРАКТИК У ВИЩІЙ ОСВІТІ

Анотація. Стаття присвячена аналізу потенціалу технологій штучного інтелекту (ШІ) щодо оптимізації інклюзивних практик у вищій освіті. Штучний інтелект відкриває нові можливості для адаптації навчального процесу до індивідуальних особливостей кожного студента, що є особливо важливим для осіб з особливими освітніми потребами.

У статті розглядаються ключові аспекти використання сучасних технологій ШІ в освітньому процесі, включаючи системи адаптивного навчання, голосові асистенти та текстові аналізатори, технології розпізнавання

обличчя та жестів. Проаналізовано як глобальні тенденції, так і конкретний досвід України в імплементації ШІ для підтримки інклюзивності в освіті.

Зокрема, акцентується увага на тому, як технології ШІ можуть допомогти в розвитку освітнього середовища для осіб із порушеннями зору, слуху, розладами аутистичного спектру, СДУГ та іншими особливостями.

У підсумку, стаття наголошує на необхідності подальшого дослідження та розвитку інноваційних підходів до впровадження технологій штучного інтелекту в освітній процес з метою створення інклюзивного, доступного та ефективного освітнього середовища. Стаття сприяє формуванню нового погляду на можливості інтеграції інклюзивних практик та штучного інтелекту в контексті вищої освіти.

Ключові слова: штучний інтелект, інклюзивна освіта, вища освіта, адаптивне навчання, голосові асистенти, розпізнавання обличчя, текстові аналізатори, технології доповненої реальності, інтелектуальні системи.

Introduction. In today's world, inclusiveness is not new, but it is gaining importance as it is one of the main principles of many democratic societies. This is reflected not only in the political but also in the educational field in many countries, including Ukraine. Thus, human rights in education are reflected in several legal documents of both national and international importance: Article 24 of the Constitution of Ukraine enshrines the principle of equality of all citizens before the law; Article 6 of the Law of Ukraine "On Education" (2017) defines inclusive education as one of the directions of state policy in the field of education; the Strategy of National Policy on Education Development in Ukraine until 2021 calls for the introduction of inclusive education in all educational institutions of the country; Article 26 of the Universal Declaration of Human Rights (1948) defines The Strategy for the Development of Inclusive Education envisages creating conditions for all students, regardless of their characteristics, to have full access to the educational process.

It is in the interest of society to find ways to optimize inclusive practices to integrate a wide range of individuals into higher education institutions. Modern artificial intelligence technologies open up unlimited possibilities for realizing this task, as they can ensure that the educational process is adapted to the needs of each student.

The analysis of artificial intelligence technologies in the context of optimizing inclusive practices in higher education reveals the potential of modern innovations to solve this significant social problem. Possible areas of AI implementation include adaptive learning systems, voice assistants, face recognition technologies, etc. Artificial intelligence technologies help to solve the issues of accessibility of education for people with disabilities, improving the quality of education and promoting their social inclusion.

Thus, the analysis of modern technologies and the study of their potential for optimizing inclusive practices in higher education is a promising area of research that can contribute to the development of effective strategies for integrating all categories of students into the educational process.

Purpose. The goal of this study is to analyze modern artificial intelligence technologies and identify ways to use them effectively to optimize inclusive practices in higher education, to improve the accessibility and quality of the educational process for students with special educational needs.

The subject of the study is a set of artificial intelligence technologies (such as adaptive learning systems, voice assistants, face recognition systems, etc.) and their potential to solve the problems of optimizing inclusive education in higher education institutions.

The object of the study is the process of inclusive education in higher education institutions, including educational programs, teaching methods, tools and technologies used to ensure inclusiveness.

Literature review. A number of domestic and foreign scholars study the issue of inclusive education in general and in higher education institutions in particular: Gokul Yenduri [1], Vasiliki Ioannidi [2], Junaidi, Ahsan [3], Palamarchuk E.A. [4], Bortun K.O. and others. Many scientists study the digitalization of the educational process, in particular, focusing on the use of artificial intelligence in education: Ushakova O.I. [5], Michurin M. [6], Sharov S.V. [7], Barna O.V. [8], Somenko D.V. [9], Mupaikwa E. [10] and others. It should be noted that the issue of using artificial intelligence to organize inclusive education in higher education institutions is not sufficiently covered.

Inclusive education and artificial intelligence are fairly new areas of research. Given that this field of study has just begun to develop quite dynamically, many scientists and researchers are still formulating basic concepts and theories. The analysis of information sources confirmed the hypothesis that the use of artificial intelligence to organize inclusive education in higher education institutions is not sufficiently described.

Results. Inclusive education has its roots in the global civil rights movement, which emphasized equal rights for all individuals, regardless of their physical, mental or social differences. Starting in the 1990s, inclusive education was gradually introduced into the curricula of many countries around the world, with special attention paid to individualizing the learning process. Today, inclusive education continues to develop and adapt to modern realities. In Ukraine, as well as in other countries, there are numerous programs and initiatives aimed at implementing inclusive practices in the educational process. In this context, artificial intelligence opens up new perspectives, allowing for the creation of adaptive learning systems that can take into account the individual characteristics of each student.

The global context is characterized by a trend towards the democratization of education, using artificial intelligence technologies to expand access to educational resources and create more personalized learning experiences.

Inclusive education has its roots in the global civil rights movement, which emphasized equal rights for all individuals, regardless of their physical, mental or social differences. Starting in the 1990s, inclusive education was gradually introduced into the curricula of many countries around the globe, with particular attention paid to individualizing the learning process. Today, inclusive education continues to develop and adapt to modern realities. In Ukraine, as in other countries, many programs and initiatives exist to implement inclusive practices in the educational process.

The global context is characterized by a trend towards democratization of education, using artificial intelligence technologies to expand access to educational resources and create more personalized learning experiences.

Inclusive education has its own peculiarities that should be taken into account to enhance the processes of inclusion of all citizens in social relations, regardless of their physical, intellectual, cultural, linguistic, national, and other characteristics. The principle of inclusion is based on the realization of the value of human diversity and differences between people. It excludes any discrimination and reflects one of the main features of a democratic society. It applies to all citizens, regardless of age. However, it should be borne in mind that most people studying in educational institutions belong to the age category of children and youth. Unfortunately, it is traditionally believed that the concept of inclusion is associated only with people who have a "disability", a measure of health loss due to illness, injury (its consequences), or congenital defects, which, in turn, in interaction with the external environment, can lead to restrictions on a person's life. Mainstream education applies to all citizens with special educational needs (SEN) [4].

According to the UNESCO classification [11], persons with SEN include those who have the following developmental disabilities: emotional and behavioural; physical/neuromotor; speech and communication; learning difficulties; hearing vision; delayed/limited intellectual development.

This list is supplemented by children who grew up in unfavourable conditions, children from ethnic minorities, street children and children with HIV/AIDS.

According to the source [12], the structure of inclusive perfection has five dimensions: intrapersonal awareness; interpersonal awareness; transformation of the curriculum; inclusive pedagogy; and inclusive learning environments.

The difficulties of forming an electronic inclusive educational environment lie in the need to identify electronic tools for people with disabilities and their integration into the learning management system. Some tools require the implementation of projects for their development and implementation. In this study, we will use the general requirements for electronic resources in accordance with the features of inclusion, which are described in Table 1 of source [4].

The use of artificial intelligence in education is growing rapidly, and with it, the potential to create a more accessible and inclusive learning environment. AI-based technologies can be used to customize learning experiences for students with different needs, abilities, and backgrounds [13].

The term 'artificial intelligence' is relatively new, so it requires additional justification. Artificial intelligence is a toolkit of a system or service that can be used to collect and adapt user data (or data posted in open repositories) and generate new solutions or conclusions based on them, according to the user's request [14].

Artificial Intelligence can be divided in various types, there are mainly two types of main categorization which are based on capabilities and based on functionality of AI. Following is flow diagram which explain the types of AI (see Fig. 1) [15].

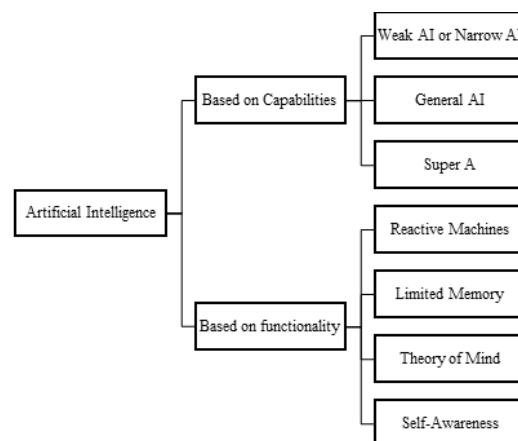


Fig. 1. Types of Artificial Intelligence

Based on Capabilities:

1. Weak AI or Narrow AI:

- Narrow AI is a type of AI which is able to perform a dedicated task with intelligence. The most common and currently available AI is Narrow AI in the world of Artificial Intelligence.
- Narrow AI cannot perform beyond its field or limitations, as it is only trained for one specific task. Hence it is also termed as weak AI. Narrow AI can fail in unpredictable ways if it goes beyond its limits.
- Apple Siri is a good example of Narrow AI, but it operates with a limited pre-defined range of functions.
- IBM's Watson supercomputer also comes under Narrow AI, as it uses an Expert system approach combined with Machine learning and natural language processing.
- Some Examples of Narrow AI are playing chess, purchasing suggestions on e-commerce site, self-driving cars, speech recognition, and image recognition.

2. General AI:

- General AI is a type of intelligence which could perform any intellectual task with efficiency like a human.
- The idea behind the general AI to make such a system which could be smarter and think like a human by its own.
- Currently, there is no such system exist which could come under general AI and can perform any task as perfect as a human.
- The worldwide researchers are now focused on developing machines with General AI.
- As systems with general AI are still under research, and it will take lots of efforts and time to develop such systems.

3. Super AI:

- Super AI is a level of Intelligence of Systems at which machines could surpass human intelligence, and can perform any task better than human with cognitive properties. It is an outcome of general AI.
- Some key characteristics of strong AI include capability include the ability to think, to reason, solve the puzzle, make judgments, plan, learn, and communicate by its own.
- Super AI is still a hypothetical concept of Artificial Intelligence. Development of such systems in real is still world changing task.

Based on functionality:

1. Reactive Machines:

- Purely reactive machines are the most basic types of Artificial Intelligence.
- Such AI systems do not store memories or past experiences for future actions.
- These machines only focus on current scenarios and react on it as per possible best action.
- IBM's Deep Blue system is an example of reactive machines.
- Google's AlphaGo is also an example of reactive machines.

2. Limited Memory:

- Limited memory machines can store past experiences or some data for a short period of time.
- These machines can use stored data for a limited time period only.
- Self-driving cars are one of the best examples of Limited Memory systems. These cars can store recent speed of nearby cars, the distance of other cars, speed limit, and other information to navigate the road.

3. Theory of Mind:

- Theory of Mind AI should understand the human emotions, people, beliefs, and be able to interact socially like humans.
- This type of AI machines are still not developed, but researchers are making lots of efforts and improvement for developing such AI machines.

4. Self-Awareness:

- Self-awareness AI is the future of Artificial Intelligence. These machines will be super intelligent, and will have their own consciousness, sentiments, and self-awareness.
- These machines will be smarter than human mind.
- Self-Awareness AI does not exist in reality still and it is a hypothetical concept [15].

In the 1970s, AIED has occurred as a specialist area to cover new technology to teaching & learning, specifically for higher education. The main aim of AIED is to facilitate the learners with flexible, personalized, and engaging learning along with the basic automated task. Some popular trends in AIED include Intelligent tutor systems, smart classroom technologies, adaptive learning, and pedagogical agents. Below diagram shows the relationship between all these trends.

In the 1970s, a specialized discipline known as the Automated Intelligent Educational Environment (AIED) emerged in higher education, aimed at introducing the latest technologies into the learning and teaching processes. The main goal of AIED is to create conditions for flexible, individualized learning, which is complemented by the automation of basic tasks.

Among the significant trends in AIED are the creation of intelligent tutoring systems, technologies for smart classrooms, adaptive learning, and pedagogical agents. The figure below demonstrates the relationship between all these areas (see Fig. 2).

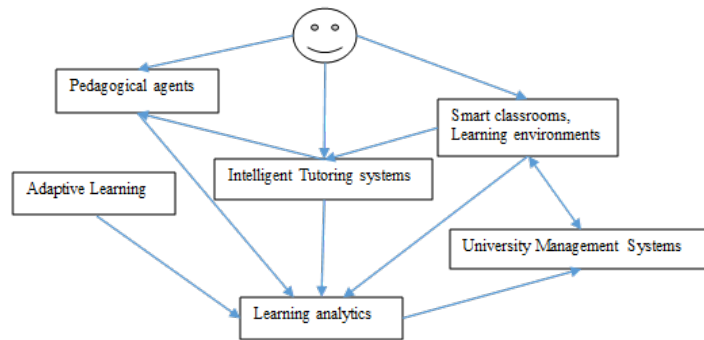


Fig. 2. Interconnection of artificial intelligence elements in higher education

The following ways of applying artificial intelligence in higher education institutions can be identified:

1. Automation of the same type of actions to assess students' knowledge.
2. Use of tutor programs to study basic material.
3. Provide instant feedback.
4. Provide personalized learning.
5. Create smart content using artificial intelligence.
6. Providing access to education for students with special needs:

- Adaptive learning systems use artificial intelligence algorithms to create individualized learning plans and monitor student progress. This helps optimize the learning process, making it more efficient and helping students with special needs find their way to success. Technologies analyze the dynamics of student learning and automatically adapt materials to their current level of knowledge and skills.

- Voice assistants and text analyzers can be extremely useful tools in inclusive education. They can help organize the learning process, convert text to speech, and support students with special needs through voice feedback and textual information analysis. This makes it possible to create a more accessible and inclusive learning environment where each student can work with the material in the most suitable format for them.

- Face and gesture recognition technologies open up new horizons for inclusive education. They can help analyze the emotional state and reaction of students to learning material, as well as adapt the learning process to the peculiarities of each student's perception.

To ensure an inclusive educational environment for students with SEN, it is advisable to use appropriate artificial intelligence systems:

1. Adaptive learning systems:

- Squirrel AI: is an adaptive learning system that uses artificial intelligence algorithms to personalize each student, analyzing their strengths and weaknesses and offering the most optimal learning path.

- DreamBox Learning: a math learning platform that adapts tasks and instructions in real time to the student's learning pace.

- ALEKS (Assessment and Learning in Knowledge Spaces) is an adaptive learning platform that provides individualized learning materials in math, chemistry, statistics, etc.

- Knewton is a system that uses big data to adapt learning materials to the needs of each student.

- Maple Learn is an interactive math learning tool that adapts the material to each student's level of proficiency.

2. Voice assistants and text analyzers:

- Google Assistant and Amazon Alexa: these assistants can help students organize their learning process, provide quick answers to questions, and help them solve various tasks.

- Grammarly: This tool uses AI to analyze text and detect grammatical, stylistic, and spelling errors, helping students improve their writing skills.

- Twinword Text Analysis: An API for text analysis that can detect sentiment, categorize text, and perform other tasks related to natural language processing.

3. Face and gesture recognition to adapt learning material:

- BrainCo: The company has developed a physical headband that can measure students' attention levels through brainwave analysis, allowing teachers to adapt their approach on the fly.

- Microsoft's Immersive Reader: This tool uses artificial intelligence technology to make reading and comprehension easier for students with special needs, including features such as reading text aloud, breaking words into syllables, and helping to translate text.

- Control other smart devices (lights, locks, thermostats, vacuum cleaners, switches) [16].

- Kinect (Microsoft) - Although initially developed for gaming, Kinect can be used for education for developing applications that respond to the user's physical actions.

Thus, we can summarize the correlation between the type of inclusion and the artificial intelligence system [4, 17-20], which should be used (Table 1).

Table 1

Available artificial intelligence systems depending on student inclusion

Disorders in Students with SEN	Features	Artificial intelligence systems	Examples
Partial visual impairment	Partial visual impairment, ability to work at a computer for some time	Text and image enlargement systems. Adaptive lighting and contrast.	Windows Magnifier: A magnification tool built into Windows operating systems. ZoomText: An application that provides powerful screen magnification and reading tools. Adaptive lighting and contrast: f.lux: a program that automatically adjusts the colour of your computer screen according to the time of day.
Visual impairment is complete	Lack of vision	Screen readers: Programs such as JAWS or NVDA can read the text displayed on the screen out loud to help the visually impaired. Voice Assistants: Voice assistants such as Google Assistant, Siri, or Alexa can help students find information or complete tasks without a screen. Text-to-Speech (TTS): TTS technologies can turn text documents into audio files, which can help visually impaired people listen to text instead of reading. Optical Character Recognition (OCR): Programs that use OCR technology can convert scanned documents or photos of text into editable text files, allowing students to use other technologies, such as TTS, to access the material.	JAWS (Job Access With Speech): one of the most popular screen reader programs. NVDA (NonVisual Desktop Access): A free and open-source screen reader. Voice assistants: Google Assistant, Siri, Alexa. Text-to-Speech (TTS): NaturalReader: a program that allows you to convert text to audio; Kurzweil 3000: an educational tool that provides various tools to support learning, including TTS. Optical Character Recognition (OCR): Adobe Acrobat: can use OCR technology to convert scanned documents into editable text files; ABBYY FineReader: a program that uses OCR technology to recognize text in scanned documents. Specialized keyboards and mice.
Hearing impairment	It is possible to listen by amplifying the sound level Ability to hear only with the help of special devices Non-perception of auditory information	For people with partial hearing loss, specialized artificial intelligence systems can be used to facilitate communication and learning. Here are some examples of such systems: Automatic speech recognition (ASR) and sign language translation systems. Systems for transcribing sound into text. Video and audio analyzers. Edutainment platforms with integrated subtitles and sign languages. Specialized software for teaching sign language. Adaptive learning platforms.	AVA: A mobile app that converts speech to text in real-time, helping deaf and hard-of-hearing people communicate more effectively. Google Live Transcribe: provides real-time transcription, helping people with hearing impairments to "read" what is being said around them. Kapwing: an online tool for automatically creating video subtitles. Coursera, edX: many online courses on these platforms have integrated subtitles and the ability to watch videos at a slower pace to make it easier to understand the material. Blackboard: a learning platform that includes tools for creating accessible content, including audio descriptions and subtitles.

Disorders in Students with SEN	Features	Artificial intelligence systems	Examples
Diseases of the musculoskeletal system	Cerebral palsy and other diseases	For people with musculoskeletal disorders, artificial intelligence systems can help create a more comfortable and accessible learning environment. Distance learning tools. Robotics for assistance.	Microsoft Teams: Allows students with physical disabilities to study remotely, avoiding the hassle of physical travel. Robot assistants, such as Robear or Toyota's Human Support Robot (HSR): help people with physical disabilities in everyday life and education.
ASD (autism spectrum disorder)	One of the disorders of psychological development. Symptoms manifested in the following disorders: understanding and expressiveness of speech; development of attachment; social contacts; functional or symbolic play. ASD may be accompanied by intellectual disabilities, problems with concentration, sensory over/under sensitivity, and reduced motivation to interact.	Adaptive learning systems. Social robots and assistants. Voice assistants. Technologies to improve concentration and focus. Visual and audio tools. Apps to improve communication skills. Specialized training software.	DreamBox Learning, ALEKS: These systems adapt to the individual needs of the student, allowing them to learn at a comfortable pace. Robots like Milo: specially designed to interact with children with ASD, helping them learn to recognize emotions and improve socialization skills. Amazon Alexa, Google Assistant: can facilitate learning by answering questions and helping to organize the learning process. Apps like Brili: help children with ASD structure their day by maintaining consistency and routine, which is often necessary for children with ASD. Visual schedules, Audio books: can facilitate the learning process by providing visual or audio cues. Proloquo2Go, Avaz: help improve communication skills by using icons and other visual tools to communicate.

Conclusions. Thus, based on the study of the role of artificial intelligence technologies in optimizing inclusive practices in higher education, we can conclude that AI plays a key role in the modern educational process. Adaptive learning systems, which allow individualized learning for a particular student, have proven to be particularly effective in this context.

Additionally, the use of voice assistants and text analyzers opens up new perspectives for people with visual and hearing impairments, making it easier for them to access educational materials and providing them with the opportunity to immerse themselves in the learning process.

In addition, facial and gesture recognition technologies can serve as tools to create a more personalized and adaptive learning experience, taking into account the individual characteristics of each student.

Thus, further study and implementation of artificial intelligence technologies in higher education is an important step towards creating a more inclusive and accessible educational environment. This will ensure quality education for all students, regardless of their characteristics and needs.

Список використаних джерел:

1. From Assistive Technologies to Metaverse – Technologies in Inclusive Higher Education for Students with Specific Learning Difficulties: A Review / G. Yenduri et al. IEEE Access. 2023. P. 1. URL: <https://doi.org/10.1109/access.2023.3289496> (date of access: 18.12.2023).
2. Ioannidi V. Research on Special and Inclusive Education in the Context of Higher Education - Teachers' Views About Labeling. International Journal of Learning and Development. 2023. Vol. 13, no. 3. P. 65. URL: <https://doi.org/10.5296/ijld.v13i3.21211> (date of access: 18.12.2023).
3. Inclusive Education in Higher Education: Baseline Study at Universitas Negeri Malang / A. R. Junaidi et al. Journal of ICSAR. 2022. Vol. 6, no. 2. P. 196. URL: <https://doi.org/10.17977/um005v6i22022p196> (date of access: 18.12.2023).
4. Паламарчук Є.А. Напрями розвитку засобів інклюзивної освіти в електронних навчальних системах / Є.А. Паламарчук, О.О. Коваленко, О.І. Черешнюк // ISSN 1997-9266. Вісник Вінницького політехнічного інституту. 2023. No 3 URL: <https://doi.org/10.31649/1997-9266-2023-168-3-100-105>

5. Ушакова І.О. Особливості використання штучного інтелекту в освіті / І. О. Ушакова, О. А. Педан // Матеріали Міжнародної науково-практичної конференції "Інформаційні технології та системи": тези доповідей, 9-10 квітня 2020 р. Х.: ХНЕУ імені Семена Кузнеця, 2020. 60 с.
6. Мічурін І. Застосування систем штучного інтелекту в освіті // Інформаційна безпека та Інформаційні технології: збірник тез доповідей IV Всеукраїнської науково-практичної конференції молодих учених, студентів і курсантів, м. Львів, 27 листопада 2020 року. Львів, ЛДУ БЖД, 2020, 249 с.
7. Шаров С.В. Сучасний стан розвитку штучного інтелекту та напрямки його використання / Шаров С.В. // Українські студії в європейському контексті. №6. 2023, с.136
8. Барна О.В. Переваги штучного інтелекту в освіті / О.В. Барна // Збірник тез доповідей V Міжнародної науково-практичної конференції «Підготовка майбутніх учителів фізики, хімії, біології та природничих наук в контексті вимог Нової української школи», 18-19 травня 2023 року, с.22
9. Садовий М.І. Штучний інтелект та неймережі в освітньому процесі: переваги та недоліки / Соменко Д.В., Трифонова О.М., Садовий М.І. // VII Всеукраїнська науково-практична інтернет-конференція «Актуальні проблеми та перспективи технологічної і професійної освіти», 2023, с. 78
10. Mupaiqua, Elisha. The Use of Artificial Intelligence The Use of Artificial Intelligence in Education in Education: Applications, Challenges, and the Way Forward. 2023. P. 269
11. ЮНЕСКО URL: <https://mep.gov.ua/content/yunesco-unesco.html> (дата звернення: 18.12.2023).
12. del Carmen Salazar M., Norton A. S., Tuitt F. A. 12: Weaving Promising Practices for Inclusive Excellence into the Higher Education Classroom. To Improve the Academy. 2010. Vol. 28, no. 20201217. URL: <https://doi.org/10.3998/tia.17063888.0028.016> (date of access: 18.12.2023).
13. Штучний інтелект та цифрова інклюзія в освіті: забезпечення доступу та рівності за допомогою інтелектуальних машин. TS2 SPACE. URL: <https://ts2.space/uk/штучний-інтелект-та-цифрова-інклюзія/#gsc.tab=0> (дата звернення: 18.12.2023).
14. Мар'єнко М., Коваленко В. Штучний інтелект та відкрита наука в освіті. Physical and Mathematical Education. 2023. Т. 38, № 1. С. 48–53. URL: <https://doi.org/10.31110/2413-1571-2023-038-1-007> (дата звернення: 18.12.2023)
15. Types of Artificial Intelligence. URL: <https://www.javatpoint.com/types-of-artificial-intelligence> (date of access: 18.12.2023).
16. Terzopoulos G., Satratzemi M. Voice Assistants and Smart Speakers in Everyday Life and in Education. Informatics in Education. 2020. P. 473–490. URL: <https://doi.org/10.15388/infedu.2020.21> (date of access: 18.12.2023).
17. On Inclusion: Video Analysis of Older Adult Interactions with a Multi-Modal Voice Assistant in a Public Setting. ACM Digital Library. URL: <https://dl.acm.org/doi/fullHtml/10.1145/3572334.3572371> (date of access: 18.12.2023).
18. Digital Technologies and Artificial Intelligence Technologies in Education. European Journal of Contemporary Education. 2021. Vol. 10, no. 2. URL: <https://doi.org/10.13187/ejced.2021.2.285> (date of access: 27.11.2023).
19. Huawei Technologies Co., Ltd. Artificial Intelligence Technology. Singapore: Springer Nature Singapore, 2023. URL: <https://doi.org/10.1007/978-981-19-2879-6> (date of access: 27.11.2023).
20. Kotlyarova I. O. Artificial Intelligence Technologies in Education. Bulletin of the South Ural State University series "Education. Educational Sciences". 2022. Vol. 14, no. 3. P. 69–82. URL: <https://doi.org/10.14529/ped220307> (date of access: 27.12.2023).

References:

1. From Assistive Technologies to Metaverse – Technologies in Inclusive Higher Education for Students with Specific Learning Difficulties (2023) A Review IEEE Access. Retrieved from: <https://doi.org/10.1109/access.2023.3289496>.
2. Ioannidi V. (2023) Research on Special and Inclusive Education in the Context of Higher Education - Teachers' Views About Labeling. International Journal of Learning and Development. Vol. 13, no. 3. P. 65. Retrieved from: <https://doi.org/10.5296/ijld.v13i3.21211>.
3. Inclusive Education in Higher Education: Baseline Study at Universitas Negeri Malang (2022) Journal of ICSAR. Vol. 6, no. 2. P. 196. Retrieved from: <https://doi.org/10.17977/um005v6i22022p196>.
4. Palamarchuk, Je.A., Kovalenko, O.O., Cheresnjuk, O.I. (2023) Naprjamy rozvytku zasobiv inkluzyvnoji osvity v elektronnykh navchalnykh systemakh [Development directions of means of inclusive education in electronic educational systems]. Visnyk Vinnycjkogho politekhnichnogho instytutu. No 3 Retrieved from: <https://doi.org/10.31649/1997-9266-2023-168-3-100-105> [in Ukrainian].
5. Ushakova, I.O., Pedan, O. A. (2022) Osoblyvosti vykorystannja shtuchnogho intelektu v osviti [Peculiarities of using artificial intelligence in education] Materialy Mizhnarodnoji naukovo-praktychnoji konferenciji "Informacijni tekhnologhiji ta systemy": tezy dopovidej. [in Ukrainian].
6. Michurin, I. (2020) Zastosuvannja system shtuchnogho intelektu v osviti [Application of artificial intelligence systems in education] Informacijna bezpeka ta Informacijni tekhnologhiji: zbirnyk tez dopovidej IV Vseukrajinskoji naukovo-praktychnoji konferenciji molodykh uchenykh, studentiv i kursantiv, [in Ukrainian].
7. Sharov, S.V. (2023) Suchasnyj stan rozvytku shtuchnogho intelektu ta naprjamky jogho vykorystannja. [The current state of artificial intelligence development and directions of its use]. Ukrajinsjki studiji v jevropejsjkomu konteksti, 6, 136 [in Ukrainian].
8. Barna, O.V. (2023) Perevaghy shtuchnogho intelektu v osviti. [Advantages of artificial intelligence in education]. Zbirnyk tez dopovidej V Mizhnarodnoji naukovo-praktychnoji konferenciji «Pidghotovka majbutnikh uchyteliv fizyky, khimiji, biologhiji ta pryrodnych nauk v konteksti vymogh Novoji ukrajinskoji shkoly», 22. [in Ukrainian].
9. Sadovyj, M.I., Tryfonova, O.M., Somenko, D.V. (2023) Shtuchnyj intelekt ta nejromerezhi v osvitnjomu procesi: perevaghy ta nedoliky. [Artificial intelligence and neural networks in the educational process: advantages and

disadvantages]. VII Vseukrajinsjka naukovo-praktychna internet-konferencija «Aktualjni problemy ta perspektyvy tekhnologichnoji i profesijnoji osvity», 78. [in Ukrainian].

10. Mupaikwa, E. (2023) The Use of Artificial Intelligence in Education: Applications, Challenges, and the Way Forward.

11. Sait Junesko Retrieved from: <https://mepr.gov.ua/content/yunesko-unesco.html>. [in Ukrainian].

12. del Carmen Salazar, M., Norton, A. S., Tuitt, F. A. (2010) 12: Weaving Promising Practices for Inclusive Excellence into the Higher Education Classroom. To Improve the Academy. Vol. 28, no. 20201217. Retrieved from: <https://doi.org/10.3998/tia.17063888.0028.016>

13. Shtuchnyj intelekt ta cyfrova inkluzija v osviti: zabezpechennja dostupu ta rivnosti za dopomoghoju intelektualnykh mashyn. (2023) TS2 SPACE. Retrieved from: <https://ts2.space/uk/shtuchnyj-intelekt-ta-cyfrova-inkluzija/#gsc.tab=0> [in Ukrainian].

14. Mar'jenko, M., Kovalenko, V. (2023) Shtuchnyj intelekt ta vidkryta nauka v osviti. [Artificial intelligence and open science in education]. Physical and Mathematical Education. T. 38, Vol. 1. pp. 48–53. Retrieved from: <https://doi.org/10.31110/2413-1571-2023-038-1-007>. [in Ukrainian].

15. Types of Artificial Intelligence. Retrieved from: <https://www.javatpoint.com/types-of-artificial-intelligence>.

16. Terzopoulos, G., Satratzemi, M. (2020) Voice Assistants and Smart Speakers in Everyday Life and in Education. Informatics in Education. pp. 473–490. Retrieved from: <https://doi.org/10.15388/infedu.2020.21>.

17. On Inclusion: Video Analysis of Older Adult Interactions with a Multi-Modal Voice Assistant in a Public Setting. ACM Digital Library. Retrieved from: <https://dl.acm.org/doi/fullHtml/10.1145/3572334.3572371>.

18. Digital Technologies and Artificial Intelligence Technologies in Education. (2021) European Journal of Contemporary Education. Vol. 10, no. 2. Retrieved from: <https://doi.org/10.13187/ejced.2021.2.285>

19. Huawei Technologies Co., Ltd. (2023) Artificial Intelligence Technology. Singapore: Springer Nature Singapore. Retrieved from: <https://doi.org/10.1007/978-981-19-2879-6>.

20. Kotlyarova, I. O. (2022) Artificial Intelligence Technologies in Education. Bulletin of the South Ural State University series "Education. Educational Sciences". Vol. 14, no. 3. pp. 69–82. Retrieved from: <https://doi.org/10.14529/ped220307>.

УДК 004.42:004.657

DOI <https://doi.org/10.32689/maup.it.2023.4.3>

Юлія ПАРФЕНЕНКО

кандидат технічних наук, доцент, доцент кафедри інформаційних технологій, Сумський державний університет, вул. Миколи Сумцова, 2, Суми, Україна, 40007 (yuliya_p@cs.sumdu.edu.ua)

ORCID: 0000-0003-4377-5132

Денис ЮРЧЕНКО

аспірант кафедри інформаційних технологій, Сумський державний університет, вул. Миколи Сумцова, 2, м. Суми, Україна, індекс 40007 (dengravity@gmail.com)

ORCID: 0000-0002-5814-7561

Дмитро ТРОЦЕНКО

магістрант факультету електроніки та інформаційних технологій, Сумський державний університет, вул. Миколи Сумцова, 2, м. Суми, Україна, індекс 40007 (dmytro.trotsenko@student.sumdu.edu.ua)

ORCID: 0009-0006-1159-6550

Yuliia PARFENENKO

Candidate of Technical Sciences, Associate Professor, Associate Professor at the Department of Information Technology, Sumy State University, 2, Mykoly Sumtsova St, Sumy, Ukraine, postal code 40007 (yuliya_p@cs.sumdu.edu.ua)

Denys YURCHENKO

Postgraduate student at the Department of Information Technology, Sumy State University, 2, Mykoly Sumtsova St, Sumy, Ukraine, postal code 40007 (dengravity@gmail.com)

Dmytro TROTSSENKO

Master Student at the Faculty of Electronics and Information Technology, Sumy State University, 2, Mykoly Sumtsova St, Sumy, Ukraine, 40007 (dmytro.trotsenko@student.sumdu.edu.ua)

Бібліографічний опис статті: Парфененко, Ю., Юрченко, Д., Троценко, Д. (2023). Розроблення вебдодатку для організації замовлення систем «розумний будинок». *Інформаційні технології та суспільство*, DOI: <https://doi.org/10.32689/maup.it.2023.4.3>

Bibliographic description of the article: Parfenenko, Yu., Yurchenko, D., Trotsenko D. (2023). Rozroblennya vebdodatku dlya organizaciyi zamovlennya system «rozumnyj budynok». [Development of a web application for smart home systems ordering]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, DOI: <https://doi.org/10.32689/maup.it.2023.4.3>

РОЗРОБЛЕННЯ ВЕБДОДАТКУ ДЛЯ ОРГАНІЗАЦІЇ ЗАМОВЛЕННЯ СИСТЕМ «РОЗУМНИЙ БУДИНОК»

Анотація. Метою роботи є розроблення вебдодатку для організації замовлення систем «розумний будинок» з можливістю підбору окремих елементів та різновидів систем розумний будинок для різних потреб користувачів. **Методологія.** При проектуванні та розробленні вебдодатку було використано методологію системного аналізу, структурно-функціонального моделювання інформаційних систем, UML моделювання, методологію структурного програмування та розроблення баз даних. **Результати.** Проведено огляд сучасного стану застосування інформаційних технологій при організації замовлення систем «розумний будинок». Визначено вимоги до вебдодатку організації замовлення систем «розумний будинок». Спроектровано архітектуру вебдодатку у вигляді клієнт-серверного застосунку. Проведено моделювання варіантів використання вебдодатку засобами уніфікованої мови моделювання UML. Спроектвано модель бази даних вебдодатку, для її програмної реалізації використано систему керування базами даних MySQL. Для вибору варіантів систем розумного будинку використано метод підтримки прийняття рішень з використанням дерев рішень. Розроблено інтерфейси для клієнта та кабінет адміністратора вебдодатку. **Наукова новизна** роботи полягає у тому, що на відміну від існуючих вебдодатків замовлення систем «розумний будинок», реалізовано можливість вибору варіантів систем з урахуванням потреб користувача, а не лише окремих компонентів, з використанням дерева рішень. Практична значимість роботи полягає у можливості використовувати розроблений вебдодаток для індивідуального замовлення систем «розумний будинок». **Висновок.** У роботі представлено архітектуру вебдодатку організації замовлення систем «розумний будинок», варіанти використання вебдодатку клієнтом та адміністратором, наведено проектування бази даних, опис бізнес-логіки використання вебдодатку клієнтами.

Ключові слова: система «розумний будинок», замовлення, вебдодаток, база даних, дерево рішень.

DEVELOPMENT OF A WEB APPLICATION FOR SMART HOME SYSTEMS ORDERING

Abstract. *The purpose of the work* is to develop a web application for organizing the order of smart home systems with the possibility of selecting individual elements and types of smart home systems for different user needs. **Methodology.** The methodology of system analysis, structural and functional modeling of information systems, UML modeling, the methodology of structural programming and database development were used in the design and development of the web application. **The results.** An overview of the current state of the use of information technologies in the organization of the order of "smart home" systems was carried out. The requirements for the web application of the organization of ordering "smart house" systems have been determined. The architecture of the web application was designed in the form of a client-server application. Modeling of options for using the web application was carried out using the unified modeling language UML. The database model of the web application was designed, and the MySQL database management system was used for its software implementation. The decision support method using decision trees was used to select options for smart home systems. Interfaces for the client and the web application administrator's office have been developed. **The scientific novelty** of the work lies in the fact that, unlike existing web applications for ordering "smart home" systems, it is possible to choose system variants taking into account the needs of the user, and not just individual components, using a decision tree. The practical significance of the work lies in the possibility of using the developed web application for individual ordering of "smart house" systems. **Conclusion.** The work presents the architecture of the web application for the organization of ordering "smart house" systems, the options for using the web application by the client and the administrator, the design of the database, and the description of the business logic of using the web application by clients.

Key words: smart home system, order, web application, database, decision tree.

Постановка проблеми. На сьогоднішній день системи «розумний будинок» стають все більш популярними з огляду на потреби у забезпеченні комфорту та екологічності. Система «розумний будинок» є складовою Інтернет Речей, що базується на взаємодії різних приладів з можливістю віддаленого автоматизованого управління без втручання особи, що приймає рішення. Прилади, як правило, оснащені датчиками, які вимірюють параметри зовнішнього середовища та показники функціонування системи. Система «розумний будинок» приймає рішення відповідно до даних, отриманих з її оточення [1]. Кількість домогосподарств, які встановлюють системи «Розумний будинок», з року в рік зростає, зокрема через потребу в енергозабезпеченні, яке підлаштовується під реальні потреби споживачів і враховує профіль енергоспоживання [2]. Дослідження, присвячені аналізу перспектив поширення технології Інтернет Речей в Україні [3], свідчать про можливість підвищення економічних показників у випадку широкого розповсюдження даної технології для використання у повсякденному житті, прикладом чого є впровадження систем «розумний будинок», але сам процес змін потребує відповідної інформаційної підтримки [4].

Системи «розумний будинок» можуть бути спроектовані з різноманіття комплектуючих від різних виробників, наявних на ринку, з урахуванням запитів клієнта. Розроблення вебдодатку організації замовлення систем «розумний будинок» є актуальною задачею, оскільки дозволяє забезпечити клієнта необхідною інформацією для вибору системи «розумний будинок», а також спрощує сам процес замовлення систем «розумний будинок» та її окремих компонентів.

Аналіз останніх досліджень і публікацій.

Вебдодатки використовуються для різних аспектів впровадження та використання систем «розумний будинок». У роботі [5] представлено вебдодаток для тестування клієнтом можливостей системи, щоб зробити роботу системи прозорішою та зрозумілішою, забезпечивши довіру до користувача. В роботі [6] запропоновано систему замовлення складових «розумного будинку» із алгоритмом підбору оптимальних параметрів.

Було проаналізовано підходи до розроблення вебсистем для організації замовлень і встановлено, що поширеними є реалізації з використанням стеку технологій MERN, який включає в себе MongoDB та Node.js, такий підхід запропоновано в роботах [7, 8].

В роботі [9] представлено розроблення систем онлайн-замовлення на основі стеку технологій LNPM/LAPM, що включає в себе операційну систему Linux, HTTP-сервери Nginx/Apache, мову програмування PHP та систему керування базами даних MySQL.

В роботі [10] розглядаються особливості побудови вебдодатку для онлайн-замовлення товарів. Наводяться приклади інтерфейса користувача для пошуку товарів, формування списку замовлень, управління замовленнями.

Постановка завдання.

Метою роботи є розроблення вебдодатку організації замовлення систем «розумний будинок». Реалізація системи замовлення у вигляді вебдодатку спрямована на те, щоб зробити дану технологію доступною для широкого кола користувачів та надасть необхідну інформацію для обґрунтованого вибору варіантів системи «розумний будинок» під власні потреби. Користувачами вебдодатку будуть усі зацікавлені особи, які розглядають можливість впровадження системи «розумний будинок» у власно-

му домогосподарстві, або вже експлуатують таку систему і мають потребу в її модернізації чи заміні комплектуючих.

Виклад основного матеріалу.

Проектування архітектури вебдодатку. Архітектура вебдодатку спроектована у вигляді тривірневої архітектури побудови веб-застосунків. Вона описує його структуру та взаємодію між основними компонентами. Також архітектура визначає принципи та підходи до розроблення вебдодатку. Архітектура вебдодатку організації замовлення системи «розумний будинок у вигляді UML діаграми компонентів представлена на рис. 1.

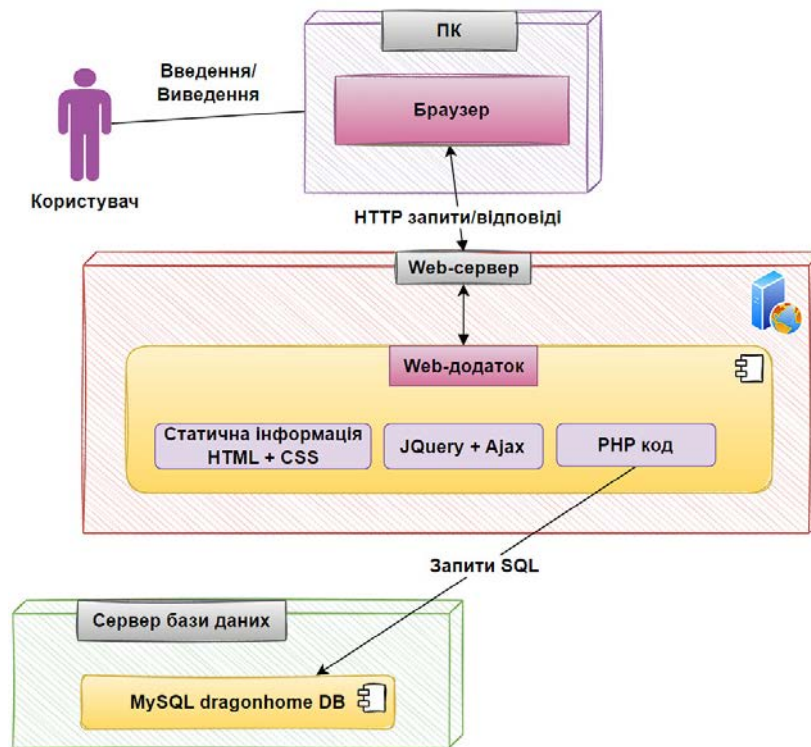


Рис. 1. Архітектура вебдодатку організації замовлення систем «розумний будинок»

Архітектура виділяє клієнтський рівень, рівні сервера додатку та сервера бази даних, між якими налагоджена двостороння взаємодія. Клієнт взаємодіє з вебдодатком через програму – браузер із будь-якого комп'ютера чи мобільного пристрою, що має вихід в мережу Інтернет. Обробка запитів від клієнта виконується веб-сервером, відображаються відповідні .html файли зі статичним контентом. Для додавання динамічного контенту на стороні клієнта використовуються JavaScript бібліотеки, зокрема jQuery. Для взаємодії із сервером та отримання від нього відповідей без перезавантаження сторінок використано технологію Ajax. Уся інформація про компоненти систем «розумного будинку», категорії користувачів зберігається у базі даних. Web-сервер виконує взаємодію із сервером бази даних, через SQL запити, отримуючи необхідні для відображення на вебсторінках дані. Наприклад, із бази даних виводиться каталог компонентів системи «розумний будинок», інформація про замовлені системи заносяться у базу даних після підтвердження замовлення адміністратором.

Моделювання бізнес-логіки використання вебдодатку. Проведено моделювання бізнес-логіки вебдодатку та визначено його основних стейкхолдерів – клієнта та адміністратора. Побудовано UML діаграми варіантів використання вебдодатку, представлені на рис. 2 та 3.

Клієнтом є особа, що використовує вебдодаток для оптимізації процесу замовлення компонентів системи «розумний будинок» чи підбору системи залежно від своїх потреб.

Нижче викладена бізнес-логіка використання вебдодатку клієнтом.

1. Клієнтом може бути фізична або юридична особа, яка відвідала сторінку сайту через браузер.
2. Клієнт може переглядати товари в каталозі.
3. Для оформлення замовлення клієнт має зареєструватися і увійти в систему як авторизований користувач.

4. Авторизований користувач може додавати товари з каталогу до кошика, оформлювати замовлення, відслідковувати історію власних замовлень.

5. Клієнт може виконувати підбір системи «розумний будинок», надавши інформацію про свої потреби шляхом опитування у веб-інтерфейсі.

На рис. 2 на діаграмі варіантів використання показано двох акторів. Зліва актором виступає клієнт, тобто особа, що зацікавлена у спрощенні замовлення та вибору систем «розумний будинок». Для забезпечення власних потреб клієнт = буде використовувати розроблений вебдодаток, який в свою чергу повинен реалізовувати бізнес-логіку клієнта. Вебдодаток повинен дозволяти клієнту переглядати інформацію про товари та системи «розумний будинок», здійснювати підбір систем «розумний будинок» за вказаними потребами та надавати можливість оформлення замовлення продукції. Іншим актором є база даних, яка взаємодіє з клієнтом через прецеденти, надаючи дані за запитом користувача.

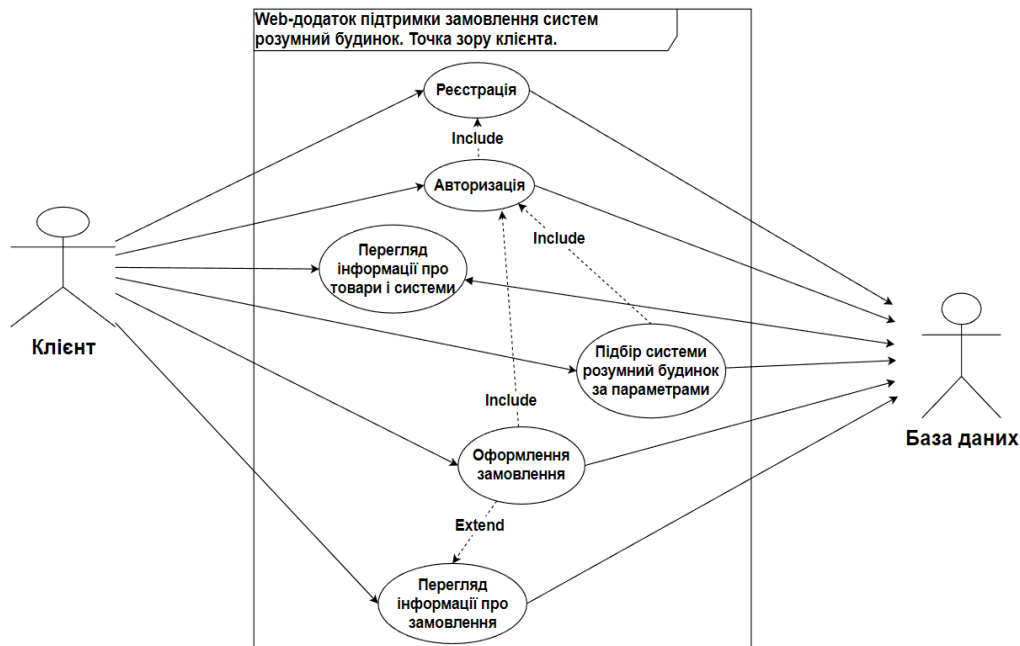


Рис. 2. Діаграма варіантів використання вебдодатку клієнтом

Бізнес-логіка адміністратора вебдодатку наведена нижче.

1. Адміністратор має повний доступ до усіх функцій вебдодатку, у тому числі клієнтських.
2. Адміністратор виконує управління даними в базі даних – редагування, додавання, видалення даних про товари.
3. Адміністратор підтверджує реєстрацію клієнтів.
4. Адміністратор виконує оброблення замовлень від клієнта, перевіряє достовірність наданої клієнтом інформації, за потреби редагує замовлення клієнта або видаляє його.
5. Адміністрування бази даних для оптимізації часу виконання запитів та захисту даних від несанкціонованого доступу.

Адміністратор виконує налаштування вебдодатку на хостингу, є відповідальним за створення резервних копій та безпеку вебдодатку.

На UML діаграмі (рис. 3) актор адміністратор вебдодатку зображений зліва. Після авторизації йому доступні усі функції, що й клієнту, тобто за потреби він також може зробити замовлення, але його основною ціллю є підтримка роботи вебдодатку. Будь-яка дія адміністратора в системі супроводжується запитом до бази даних, яка показана справа як один із акторів.

Побудовані діаграми варіантів використання вебдодатку організації замовлення систем «розумний будинок» відображають бізнес-логіку його використання основними акторами – клієнтом та адміністратором та відповідають визначеним функціональним вимогам.

Проектування бази даних вебдодатку.

При проектуванні бази даних вебдодатку організації замовлення системи «розумний будинок» було розроблено концептуальну модель, яка відображає зв'язки між основними сутностями й атрибутами, та логічну модель.

Фізична модель бази даних є розширенням та доповненням концептуальної моделі, що є більш деталізованою. Вона відображає специфікації для кожного із атрибутів, зокрема довжину полів, тип даних. Ця модель призначається для розробників бази даних. Також на фізичній моделі даних показуються первинні ключі, обмеження, які накладаються на дані, процедури. Фізична модель бази даних вебдодатку організації замовлення систем «розумний будинок» зображена на рис. 4.

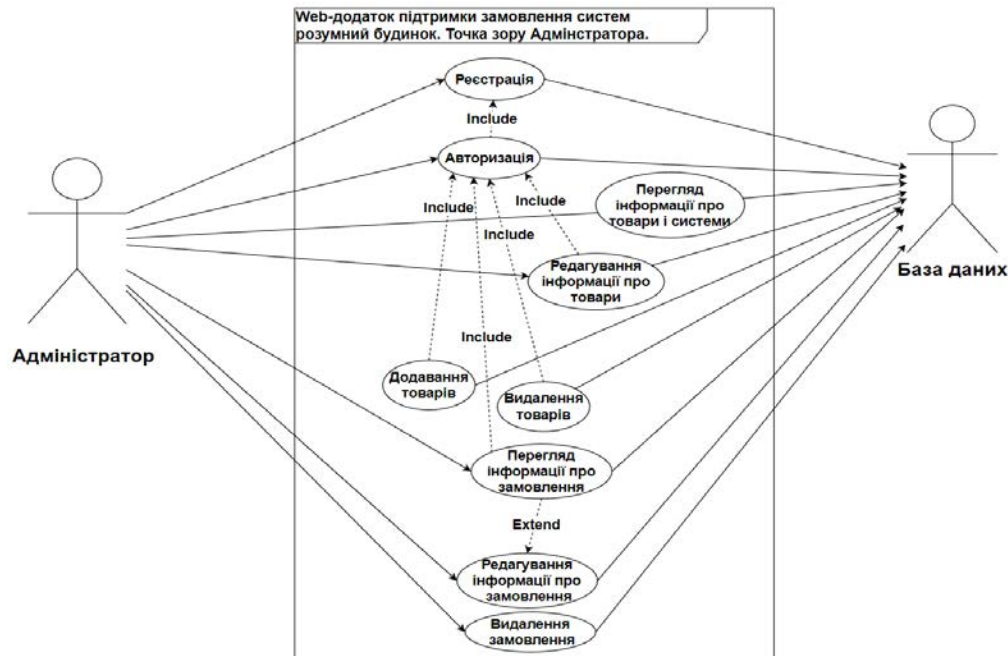


Рис. 3. Діаграма варіантів використання вебдодатку адміністратором

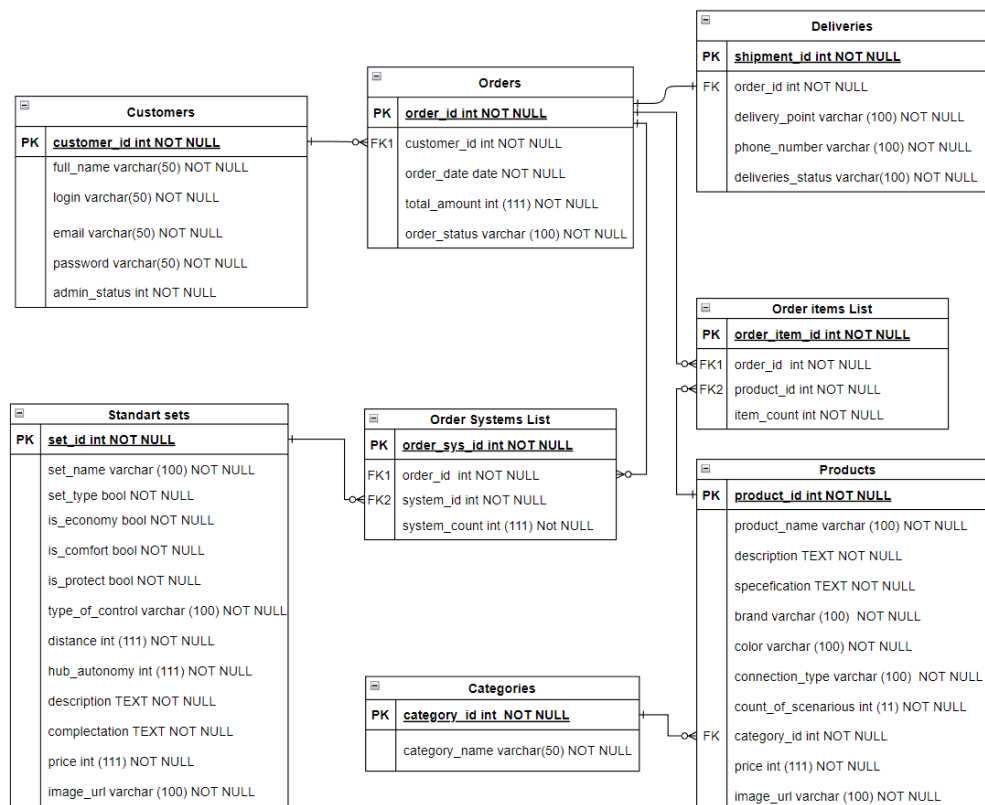


Рис. 4. Фізична модель бази даних вебдодатку організації замовлення систем «розумний будинок»

База даних складається із восьми взаємопов'язаних таблиць. Таблиця Categories зберігає дані про категорії товарів. Таблиця Products містить дані про окремі товари. Вона пов'язана з таблицею категорій Categories зовнішнім ключем category_id. Цей зв'язок визначає належність товарів до певної категорії компонентів системи. Таблиця Orders зберігає дані про усі сформовані замовлення клієнтів. Вона містить дані загальної суми товарів у кошику, дату замовлення, його поточний статус та ідентифікатор клієнта. Таблиця Customers містить дані користувача, що використовуються при авторизації та зворотньому зв'язку з користувачем.

Таблиця Deliveries містить дані про доставку конкретного замовлення. Таблиці Order Systems List та Order Items List містять списки компонентів та систем розумний будинок, що містяться у конкретному замовленні клієнта. Дані у таблицях формуються із полів, що є ідентифікаторами замовленої системи чи компонента system_id та product_id, а також їх кількості system_count_id та item_count_id.

Таблиця Standart sets містить дані про варіанти системи «розумний будинок», які будуть запропоновані клієнту за результатами опитування. Поле complectation зберігає перелік компонентів, що складають набір, is_comfort, is_economy, is_protect, set_type – відповідають потребі клієнта із вибором моносистеми чи мультисистеми із забезпеченням комфорту, економічності чи екологічності.

Розроблення дерева рішень для індивідуального замовлення системи «розумний будинок». Особливістю вебдодатку організації замовлення системи «розумний будинок» є можливість підбору варіанту системи серед запропонованих через визначення уподобань користувача шляхом опитування. Для забезпечення підтримки прийняття рішень щодо підбору системи «розумний будинок» було побудоване та програмно реалізоване дерево рішень, схематичне зображення якого наведено на рис.5.

Структура, показана на рис. 5, є орієнтовною, і за потреби може бути розширена для деталізованого вибору. Дерево рішень схематично відображає зв'язок між запитаннями, які ставляться клієнту, та варіантами підібраних систем.

На кожному етапі вибору користувачу потрібно здійснити вибір, висловивши свої можливості та уподобання, обираючи один із запропонованих варіантів. Наприклад, обрати один із трьох сценаріїв використання системи «розумний будинок» при виборі моносистеми або з двох варіантів при виборі типу системи.

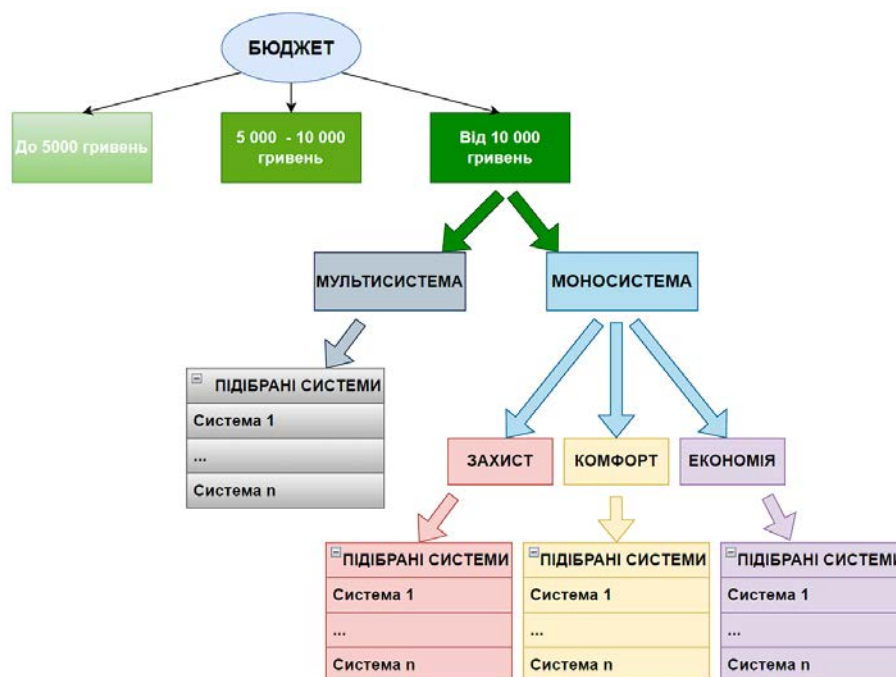


Рис. 5. Дерево рішень для підбору варіантів системи «розумний будинок»

Як результат послідовної відповіді на поставлені запитання клієнту буде запропоновано перелік варіантів систем «розумний будинок», що відповідають його запитам.

Використання вебдодатку для замовлення систем «Розумний будинок» клієнтом. Робота із починається із головної сторінки, зображеної на рис. 6. Неавторизований користувач має можливість перегляду товарів, що належать до відповідної категорії, наприклад, Датчики.

Зліва знаходиться панель фільтрів для виведення усіх товарів відповідної категорії, діапазону цін та кількості сценаріїв. Фільтр буде застосований після натискання на кнопку «Пошук». Реалізовано можливість сортування товарів за ціною та назвою. Також наявна форма пошуку товарів за назвою. Каталог товарів відображається у центральній частині головної сторінки вебдодатку. Для кожного товару відображається зображення товару, назва та ціна. Сторінка з детальною інформацією про товар доступна зареєстрованим користувачам після натискання на кнопку «Переглянути».

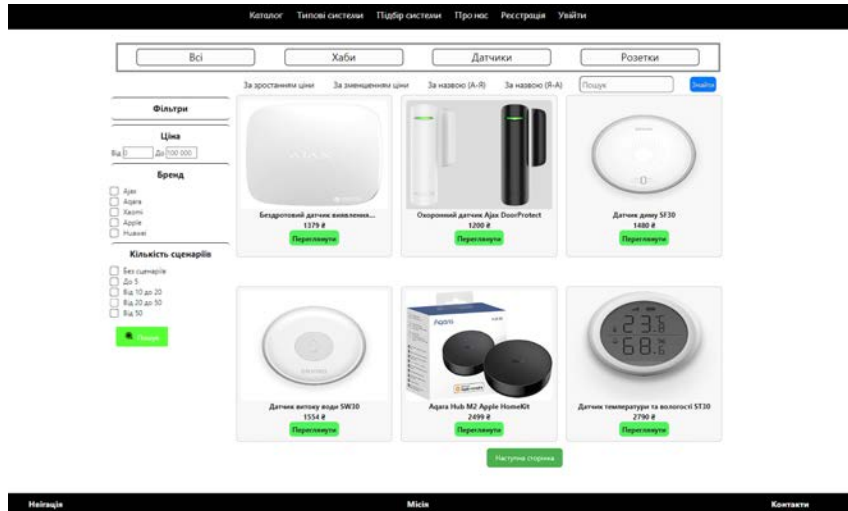


Рис. 6. Інтерфейс вебдодатку. Сторінка каталогу

Для того, щоб переглянути детальну інформацію та придати товар, клієнту необхідно зареєструватися та авторизуватися.

За необхідності індивідуального підбору системи «розумний будинок» клієнту потрібно обрати пункт «Підбір системи» на головному меню. Після цього у веб-інтерфейсі системи буде виведена анкета, що містить варіанти вибору із детальним поясненням, що собою представляє кожен критерій вибору системи. Спершу користувач обирає грошовий діапазон, вказуючи, на яку суму розраховує при придбанні системи «розумний будинок». На наступному кроці при підборі системи «розумний будинок» потрібно зазначити тип системи: моносистеми чи мультисистеми. У випадку вибору моносистеми клієнту треба визначитися із її типом залежно від сценарію використання. Пропонується здійснити вибір за трьома варіантами: комфорт, економія та захист довкілля. Приклад інтерфейсу вибору сценарію використання системи «розумний будинок» показано на рис 7.

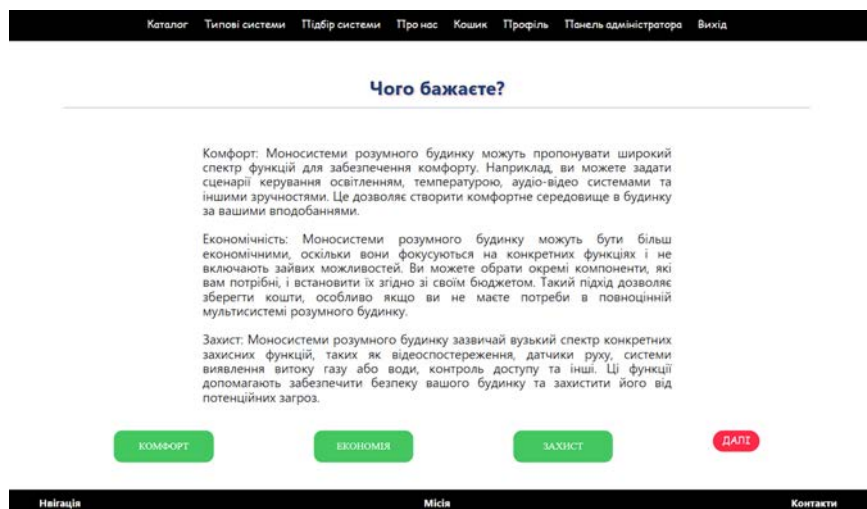


Рис. 7. Інтерфейс вебдодатку. Вибір характеристики системи «розумний будинок»

Якщо варіанти систем, які відповідають запитам користувача, не будуть знайдені, він отримає відповідне повідомлення у інтерфейсі вебдодатку.

Після того, як компоненти системи чи самі системи «розумного будинку» додано до кошику, відбувається перехід до сторінки оформлення замовлення, де клієнт може переглянути загальну суму покупки та підтвердити замовлення, вказавши контактні дані. Таким чином відбувається підтримка організації замовлення систем «розумний будинок».

Окремо реалізовано панель адміністратора, який має повні права на управління замовленнями, а також редагування, видалення, додавання інформації про системи та їх компоненти.

Висновки. В роботі описано проектування та реалізацію вебдодатку замовлення системи «розумний будинок» із можливістю підбору варіанту системи з урахуванням потреб користувача. Підбір здійснюється на основі побудованого дерева рішень, яке реалізоване у вигляді опитувальника, який пропонується клієнту. Після зазначення власних потреб клієнту надається можливість переглянути у веб-інтерфейсі можливі варіанти та обрати найбільш підходящий. Такий підхід до підбору систем «розумний будинок» дозволяє забезпечити клієнтоорієнтованість і зручність користування вебдодатком.

Список використаних джерел:

1. Сосновська, О. О., Вакофян, В. Г. Індустрія 4.0: сутність і тенденції розвитку. *Бізнес Інформ*. 2023. Т. 14, № 30. С. 137–144.
2. Sovacool, B. K., Furszyfer Del Rio, D. D. Smart home technologies in Europe: a critical review of concepts, benefits, risks and policies. *Renewable and Sustainable Energy Reviews*. 2020. Vol. 120. P. 109663.
3. Ноджак Л., Парашич М. Розвиток 4.0 індустрії в Україні: проблеми, перспективи. *Економіка та суспільство*. 2022. № 45.
4. Valencia-Arias, A. et al. Smart home adoption factors: A systematic literature review and research agenda. *PLOS ONE*. 2023. vol.18.
5. Đorđević, S., Jevremović, S., Tošić, J., Stojanovic, N. Smart house web application: design and implementation using Java EE, MVC framework and Arduino microcontroller. *Annals of Spiru Haret University Economic Series*. 2022. Vol.12(21). 245.
6. K. -s. Hong, H. J. Kim and C. Lee. Automated Grocery Ordering Systems for Smart Home. *Future Generation Communication and Networking (FGCN 2007)*, Jeju, Korea (South), 2007. P 87-92.
7. Santosh, K. S. et al. Application using MERN Stack. *International journal for modern trends in science and technology*. 2022. Vol. 8, no. 6. P. 102–105.
8. Malshika M. D. J. Computerized system to manage business functions of e-commerce web application using MERN stack technology. *International journal of engineering and management research*. 2022. Vol. 12. No 5. P. 472–481.
9. Gumilang, S.F., Ambarsari, N., & Nurmala, M.D. Web-Based Food Delivery Management System. *Proceedings of the 2018 International Conference on Industrial Enterprise and System Engineering (IcoIESE 2018)*. 2019.
10. Wijegunaratna, K.P.S.G.G. et al. Online Supermarket Web Application. *International Journal of Engineering and Management Research*. 2022. Vol 12(6), P. 216–223.

References:

1. Sosnovska O. O., & Vakofian V. H. (2023). Industriia 4.0: sutnist i tendentsii rozvytku [Industry 4.0: essence and development trends]. *Biznes inform – Business Inform*, 14, 30, 137–144 [in Ukrainian].
2. Sovacool B. K., & Furszyfer Del Rio, D. D. (2020). Smart home technologies in Europe: a critical review of concepts, benefits, risks and policies. *Renewable and sustainable energy reviews*, 120, 109663.
3. Nodzhak L., Parashchych M. (2022). Rozvytok 4.0 industrii v Ukraini: problemy, perspektyvy [Development of industry 4.0 in Ukraine: problems, prospects.]. *Ekonomika ta suspilstvo – Economy and society*, 45. [in Ukrainian].
4. Valencia-Arias, A., et al. (2023). Smart home adoption factors: A systematic literature review and research agenda. *PLOS ONE*, 18.
5. Đorđević, Strahinja & Jevremović, Svetlana & Tošić, Jovana & Stojanovic, Nina. (2022). Smart house web application: design and implementation using Java EE, MVC framework and Arduino microcontroller. *Annals of Spiru Haret University Economic Series*, 12(21), 245.
6. K. -s. Hong, H. J. Kim and C. Lee. (2007). Automated Grocery Ordering Systems for Smart Home, *Future Generation Communication and Networking (FGCN 2007)*, Jeju, Korea (South), 87-92.
7. Santosh, K. S. et al. (2022). Application using MERN Stack *International journal for modern trends in science and technology*, 8, 6, 102–105.
8. Malshika, M. D. J. (2022). Computerized system to manage business functions of e-commerce web application using MERN stack technology. *International journal of engineering and management research*, 12, 5, 472–481.
9. Gumilang, S.F., Ambarsari, N., & Nurmala, M.D. (2019). Web-Based Food Delivery Management System. *Proceedings of the 2018 International Conference on Industrial Enterprise and System Engineering (IcoIESE 2018)*.
10. Wijegunaratna, K.P.S.G.G et al. (2022). Online Supermarket Web Application. *International Journal of Engineering and Management Research*, 12(6), 216–223.

НАУКОВЕ ВИДАННЯ

**ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ
ТА СУСПІЛЬСТВО**

**INFORMATION TECHNOLOGY
AND SOCIETY**

**ВИПУСК 4 (10)
ISSUE 4 (10)**

2023

Коректура
Ірина Чудеснова

Комп'ютерна верстка
Наталія Кузнєцова

Формат 60x84/8. Гарнітура Cambria.
Папір офсет. Цифровий друк. Ум. друк. арк. 3,72. Замов. № 0124/088. Наклад 300 прим.

Видавництво і друкарня – Видавничий дім «Гельветика»
65101, Україна, м. Одеса, вул. Інглєзі, 6/1
Телефон +38 (095) 934 48 28, +38 (097) 723 06 08
E-mail: mailbox@helvetica.ua
Свідоцтво суб'єкта видавничої справи
ДК No 7623 від 22.06.2022 р.