

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 1 (3), 2022
Issue 1 (3), 2022**



Видавничий дім
«Гельветика»
2022

*Рекомендовано до друку Вченою радою
Міжрегіональної Академії управління персоналом
(протокол № 2 від 31 березня 2022 року)*

Інформаційні технології та суспільство / [головний редактор О. Попов]. – Київ: Міжрегіональна Академія управління персоналом, 2022. – Випуск 1 (3). – 62 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, с.н.с., заступник директора з науково-організаційної роботи, Державна установа «Інститут геохімії навколишнього середовища 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; **Юдін О. К.** – д-р техн. наук, проф., директор центру кібербезпеки Навчально-наукового інституту інформаційної безпеки та стратегічних комунікацій, Національна академія Служби безпеки України; **Гопєєнко 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).

*Свідоцтво 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. 2 dated 31 March 2022)*

Information Technology and Society / [chief editor Oleksandr Popov]. – Kyiv: Interregional Academy of Personnel Management, 2022. – Issue 1 (3). – 62 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, Senior Research Scientist, Deputy Director for Scientific-Organizational Affairs, SI “Institute of Environmental Geochemistry 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 Cholyskhina** – PhD in Engineering, Associate Professor, Director of the Institute of Computer Information Technologies and Design, Interregional Academy of Personnel Management; **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, 2022
© Copyright by the contributors, 2022

ЗМІСТ

Тетяна ВАКАЛЮК, Сергій ЛЮЩЕНКО, Юрій ЄФРЕМОВ, Олег ВЛАСЕНКО, Дмитро ЛИСОГОР ТЕОРЕТИЧНІ АСПЕКТИ РОЗРОБКИ СИСТЕМИ РОЗПІЗНАВАННЯ ЛЮДСЬКОГО ОБЛИЧЧЯ	7
Віктор БОЙКО, Микола ВАСИЛЕНКО, Валерій РАЧУК НОВЕ (ДРУГЕ) НАРОДЖЕННЯ МОВИ «ПРОЛОГ» (PROLOG) В КОНТЕКСТІ СИСТЕМ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ	16
Андрій ЖМАКІН, Віталій КОВАЛЬ, Володимир ЛЮБЧАК, Святослав ШПІЦГЛУЗ ПРОГРАМНО-ТЕХНІЧНІ РІШЕННЯ СТВОРЕННЯ БЮДЖЕТНОГО ВАРІАНТУ КОМП'ЮТЕРНИХ СИСТЕМ НАВЧАЛЬНОГО ТА ОФІСНОГО ПРИЗНАЧЕННЯ	23
Галина КИРИЧЕК, Марія ТЯГУНОВА, Анна КУРАЧ АВТОМАТИЗОВАНЕ ТЕСТУВАННЯ ВЕБ-ПЛАТФОРМ З ВИКОРИСТАННЯМ JAVA ТА SELENIUM	31
Олена КРИВОРУЧКО, Юлія КОСТЮК СТРУКТУРНО-ФУНКЦІОНАЛЬНЕ МОДЕЛЮВАННЯ ТЕХНОЛОГІЧНОГО ПРОЦЕСУ ВИРОБНИЦТВА ВЕРШКОВОГО МАСЛА	38
Іван МАРИНИЧ, Ольга СЕРДЮК ЗАСТОСУВАННЯ НЕЙРОННИХ РЕГУЛЯТОРІВ ПРИ МОДЕЛЮВАННІ КЕРУВАННЯ СТАДІЄЮ ПОДРІБНЕННЯ В УМОВАХ ГІРНИЧО-ЗБАГАЧУВАЛЬНОГО КОМБІНАТУ	45
Марина НОС ОГЛЯД СУЧАСНИХ МЕТОДОЛОГІЙ УПРАВЛІННЯ ВАРТІСТЮ ІТ-ПРОЄКТІВ.....	54

CONTENTS

Tetiana VAKALIUK, Serhiy ILIUSHCHENKO, Iurii IEFREMOV, Oleh VLASENKO, Dmytro LYSOGOR
THEORETICAL ASPECTS OF THE DEVELOPMENT OF THE HUMAN FACE RECOGNITION SYSTEM7

Viktor Boyko, Nikolai VASILENKO, Valery RACHUK, Valeria SLATVINSKAYA
THE NEW (SECOND) BIRTH OF THE “PROLOG” LANGUAGE
IN THE CONTEXT OF DECISION SUPPORT SYSTEMS16

Andrew ZHMAKIN, Vitalii KOVAL, Volodymyr LIUBCHAK, Sviatoslav SHPITSHLUZ
SOFTWARE AND TECHNICAL SOLUTIONS FOR CREATING A BUDGET OPTION
OF COMPUTER SYSTEMS FOR EDUCATIONAL AND OFFICE PURPOSES23

Halyna KYRYCHEK, Mariia TIAHUNOVA, Anna KURACH
AUTOMATED TESTING OF WEB PLATFORMS USING JAVA AND SELENIUM31

Olena KRYVORUCHKO, Yuliia KOSTIUK
STRUCTURAL-FUNCTIONAL MODELING OF TECHNOLOGICAL PROCESS OF CREAM OIL PRODUCTION38

Ivan MARYNYCH, Olga SERDIUK
USE OF NEURAL REGULATORS IN CONTROL SIMULATION OF CRUSHING STAGE
IN CONDITIONS OF MINING AND PROCESSING PLANT46

Marina NOS
OVERVIEW OF MODERN METHODOLOGIES OF IT PROJECT MANAGEMENT54

УДК 004.4

DOI <https://doi.org/10.32689/maup.it.2022.1.1>

Тетяна ВАКАЛЮК

доктор педагогічних наук, професор, професор кафедри інженерії програмного забезпечення, Державний університет «Житомирська політехніка», вул. Чуднівська, 103, Житомир, Україна, індекс 10005 (tetianavakaliuk@gmail.com)

ORCID: 0000-0001-6825-4697

Сергій ІЛЮЩЕНКО

здобувач, Державний університет «Житомирська політехніка», вул. Чуднівська, 103, Житомир, Україна, індекс 10005 (silliuschenko@gmail.com)

ORCID: 0000-0002-1326-7711

Юрій ЄФРЕМОВ

кандидат технічних наук, доцент, доцент кафедри інженерії програмного забезпечення, Державний університет «Житомирська політехніка», вул. Чуднівська, 103, Житомир, Україна, індекс 10005

ORCID: 0000-0002-1249-5560

Олег ВЛАСЕНКО

старший викладач кафедри інженерії програмного забезпечення, Державний університет «Житомирська політехніка», вул. Чуднівська, 103, Житомир, Україна, індекс 10005 (oleg@ztu.edu.ua)

ORCID: 0000-0001-6697-2150

Дмитро ЛИСОГОР

аспірант, старший викладач кафедри інженерії програмного забезпечення, Державний університет «Житомирська політехніка», вул. Чуднівська, 103, Житомир, Україна, індекс 10005 (lysogor@ukr.net)

ORCID: 0000-0001-5980-1708

Tetiana VAKALIUK

Doctor of Pedagogical Sciences, Professor, Professor at the Department of Software Engineering, Zhytomyr Polytechnic State University, 103 Chudnivska str., Zhytomyr, Ukraine, postal code 10005 (tetianavakaliuk@gmail.com)

Serhiy ILIUSHCHENKO

Applicant, Zhytomyr Polytechnic State University, 103 Chudnivska str., Zhytomyr, Ukraine, postal code 10005 (silliuschenko@gmail.com)

Iurii IEFREMOV

Candidate of Technical Sciences, Associate Professor, Associate Professor at the Department of Software Engineering, Zhytomyr Polytechnic State University, 103 Chudnivska str., Zhytomyr, Ukraine, postal code 10005

Oleh VLASENKO

Senior Lecturer at the Department of Software Engineering, Zhytomyr Polytechnic State University, 103 Chudnivska str., Zhytomyr, Ukraine, postal code 10005 (oleg@ztu.edu.ua)

Dmytro LYSOGOR

Postgraduate Student, Senior Lecturer at the Department of Software Engineering, Zhytomyr Polytechnic State University, 103 Chudnivska str., Zhytomyr, Ukraine, postal code 10005 (lysogor@ukr.net)

Бібліографічний опис статті: Вакалюк Т., Ілющенко С., Єфремов Ю., Власенко О., Лисогор Д. Теоретичні аспекти розробки системи розпізнавання людського обличчя. *Інформаційні технології та суспільство*. 2020. Вип. 1 (3). С. 6–15. DOI: <https://doi.org/10.32689/maup.it.2022.1.1>

Bibliographic description of the article: Vakaliuk, T., Iliushchenko, S., Iefremov, Yu., Vlasenko, O., Lysogor, D. (2022). Teoretychni aspekty rozrobky systemy rozpoznavannia liudskoho oblychchia [Theoretical aspects of the development of the human face recognition system]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, 1 (3), 6–15. DOI: <https://doi.org/10.32689/maup.it.2022.1.1>

ТЕОРЕТИЧНІ АСПЕКТИ РОЗРОБКИ СИСТЕМИ РОЗПІЗНАВАННЯ ЛЮДСЬКОГО ОБЛИЧЧЯ

Анотація. Система розпізнавання облич – це технологія, здатна зіставляти людське обличчя з цифровим зображенням або відеокадром з базою даних осіб, зазвичай використовується для автентифікації користувачів за допомогою служб перевірки особистості, працює шляхом точного визначення і вимірювання рис обличчя по даному зображенню. Системи розпізнавання обличчя використовуються сьогодні в усьому світі урядами та приватними компаніями, їх ефективність різна, і деякі системи раніше були списані через їх неефективність. Отже, створення програми для розпізнавання людського обличчя є актуальною темою. Метою статті є дослідження теоретичних аспектів розробки системи розпізнавання людського обличчя.

Процедура розпізнавання обличчя просто вимагає, щоб будь-який пристрій, оснащений цифровою фотографічною технологією, генерував і отримував зображення та дані, необхідні для створення та запису біометричного малюнка обличчя людини, якого необхідно ідентифікувати.

Розглянуто основні алгоритми розпізнавання людського обличчя: розпізнавання обличчя з використанням різних поверхонь облич, метод обличчя Фішера, метод аналізу головних компонентів та машина опорних векторів, метод каскадів Хаара. Наведено їх переваги та недоліки. Наведено застосування згорткової нейронної мережі до розпізнавання обличчя. Запропоновано реалізацію алгоритму роботи системи розпізнавання обличчя.

В даній роботі проаналізовано наявні алгоритми та системи виявлення та розпізнавання обличчя, зважені їх переваги та недоліки. Розглянуто використання згорткової нейронної системи з метою розпізнавання обличчя. Проаналізовано на практиці відсоток точності розпізнавання людського обличчя та продуктивність, враховуючи такі фактори як освітлення, якість зображення, кількість обличчя на зображенні використовуючи бібліотеку з відкритим вихідним кодом Face recognition із сімейства бібліотек DLib в основі якої лежить згорткова нейронна мережа.

Ключові слова: розпізнавання, нейронні мережі, методи розпізнавання.

THEORETICAL ASPECTS OF THE DEVELOPMENT OF THE HUMAN FACE RECOGNITION SYSTEM

Abstract. Face Detection System is a technology that compares a human face with a digital image or video frame to a database of individuals, commonly used to authenticate users through identity verification services, and works by accurately identifying and measuring facial features in a given image. Facial recognition systems are used today by governments and private companies around the world, their effectiveness varies, and some systems have previously been written off because of their ineffectiveness. Thus, the creation of a program for human face recognition is a topical issue. The aim of the article is to study the theoretical aspects of developing a human face recognition system.

The face recognition procedure simply requires that any device equipped with digital photographic technology generate and obtain the images and data necessary to create and record a biometric image of the person to be identified.

The main algorithms of human face recognition are considered: face recognition using different face surfaces, Fisher's face method, principal components analysis method and machine of reference vectors, Haar cascade method. Their advantages and disadvantages are given. The application of convolutional neural network to face recognition is presented. The implementation of the algorithm of the face recognition system is proposed.

This paper analyzes the existing algorithms and systems for face detection and recognition, weighing their advantages and disadvantages. The use of convolutional neural system for facial recognition is considered. The percentage of human face recognition accuracy and performance were analyzed in practice, taking into account factors such as lighting, image quality, number of faces in the image using the face recognition library Face recognition from the DLib family of libraries based on convolutional neural network.

Key words: recognition, neural networks, recognition methods.

Постановка проблеми. Система розпізнавання облич – це технологія, здатна зіставляти людське обличчя з цифровим зображенням або відеокадром з базою даних осіб, зазвичай використовується для автентифікації користувачів за допомогою служб перевірки особистості, працює шляхом точного визначення і вимірювання рис обличчя по даному зображенню.

Розробка подібних систем почалася в 1960-х роках, почавшись як форма комп'ютерного додатка. З моменту свого створення системи розпізнавання осіб останнім часом стали ширше використовуватися на смартфонах і в інших технологіях, таких як Робототехніка. Оскільки комп'ютеризоване розпізнавання обличчя включає в себе вимірювання фізіологічних характеристик людини, системи розпізнавання обличчя класифікуються як біометричні. Хоча точність систем розпізнавання осіб як біометричної технології нижче, ніж розпізнавання райдужної оболонки ока і відбитків пальців, вона широко застосовується завдяки безконтактному процесу. Системи розпізнавання обличчя використовуються сьогодні в усьому світі урядами та приватними компаніями, їх ефективність різна, і деякі системи раніше були списані через їх неефективність. Отже, створення програми для розпізнавання людського обличчя є актуальною темою.

Аналіз останніх досліджень і публікацій. Методи розпізнавання людського обличчя розглядали багато учених, зокрема вітчизняних: Антошук С.Г., Бурак Т.О., Даниленко О.І., Ділай В.І., Загородня Д.І., Крилов В.М., Ляшенко Г.Є., Орленко С.П., Палій І.О., Саченко А.О., Тимошин Ю.А., та зарубіжних: M.Ş. Bayhan, W.C. Cheng, G. Geng, M. Gökmen, A. Houacine, A.K. Jain, S. Kherchaoui, J. Kovac, C. Lin, X. Liu, P. Peer, F. Solina, C.C. Tsai, J.S. Taur, C.W. Tao, M. Taayab, X. Wang, W. Yun-qiong, M.F. Zafar, L. Zhi-fang, Y. Zhi-sheng, та ін.

Зокрема, група авторів (Палій І.О., Саченко А.О., Антошук С.Г., Бурак Т.О.) розглядали нейромережний підхід до комп'ютерного розпізнавання обличч [9]. Автори здійснили представлення підходу до розпізнавання обличч на основі комбінованого каскаду нейромережних класифікаторів, а також методу головних компонент і згорткової нейронної мережі [9]. Тимошин Ю.А. та Орленко С.П. запропонували алгоритм розпізнавання обличчя людей на базі згорткової нейронної мережі [10]. Окрім того, автори ще й розглянули інші методи детектування і розпізнавання обличч, а також проаналізовано можливість їх комбінацій [10]. Ляшенко Г.Є. та Даниленко О.І. досліджували різні методи розпізнавання обличч [11], аналізували методи біометричної автентифікації [11]. Загородня Д.І., Палій І.О. та Крилов В.М. запропонували метод розпізнавання обличч за характерними точками контуру [12]. Ділай В.І. провів огляд наявних методів розпізнавання обличч для конкретної області – для використання в системах контролю і управління доступом [13].

Мета статті – дослідження теоретичних аспектів розробки системи розпізнавання людського обличчя.

Виклад основного матеріалу. Процедура розпізнавання обличчя просто вимагає, щоб будь-який пристрій, оснащений цифровою фотографічною технологією, генерував і отримувал зображення та дані, необхідні для створення та запису біометричного малюнка обличчя людини, якого необхідно ідентифікувати.

На відміну від інших рішень для ідентифікації, таких як паролі, перевірка по електронній пошті, Селфі або зображення або ідентифікація відбитків пальців, біометричне розпізнавання осіб використовує унікальні математичні і динамічні шаблони, які роблять цю систему однією з найбезпечніших і ефективних. Мета розпізнавання обличч полягає в тому, щоб по вхідному зображенню знайти серію даних одного і того ж особи в наборі навчальних зображень в базі даних. Велика складність полягає в забезпеченні того, щоб цей процес здійснювався в режимі реального часу, що недоступно всім постачальникам програмного забезпечення для біометричного розпізнавання осіб. Процес розпізнавання обличч може виконуватися в двох варіантах залежно від того, коли він виконується:

– Той, в якому вперше система розпізнавання осіб звертається до особи, щоб зареєструвати його і зв'язати з особистістю таким чином, щоб воно було записано в системі. Цей процес також відомий як цифрова адаптація з розпізнаванням обличчя.

– Варіант, в якому користувач проходить аутентифікацію до реєстрації. У цьому процесі вхідні дані з камери перетинаються з існуючими даними в базі даних. Якщо особа збігається з уже зареєстрованим посвідченням особи, користувачеві надається доступ до системи з його обліковими даними.

Розглянемо основні алгоритми розпізнавання людського обличчя.

Розпізнавання обличчя з використанням різних поверхонь обличч. Цей підхід до розпізнавання осіб використовує концепції зменшення розмірності і лінійної алгебри для розпізнавання осіб. Це метод виявлення і розпізнавання осіб, який визначає дисперсію осіб в наборах даних зображень. Він використовує ці відхилення для кодування та декодування осіб за допомогою машинного навчання [1]. Набір власних осіб являє собою набір «стандартизованих компонентів особи», що визначаються шляхом статистичного аналізу великої кількості зображень осіб рисам обличчя присвоюються математичні значення, так як цей метод використовує не цифрові зображення, а статистичні бази даних (рис. 1). Будь-яке людське обличчя являє собою комбінацію цих значень з різним відсотком.

Перевагою цього алгоритму є те, що власні поверхні були винайдені саме для цих цілей, що робить систему дуже ефективною. Недоліком є те, що він чутливий до умов освітлення і положення голови. Недоліки-пошук власних векторів і власних значень забирає багато часу.

Метод обличч Фішера. Обличчя є одним з найпопулярніших алгоритмів розпізнавання обличчя. Вважається, що він перевершує багато його альтернатив. В якості поліпшення алгоритму власних поверхонь його часто порівнюють з власними поверхнями і вважають більш успішним в розрізненні кластерів в процесі навчання [2]. Ключовою перевагою цього алгоритму є його здатність інтерполювати та екстраполювати зміни освітлення та виразу обличчя. Є повідомлення про 93% точності алгоритму обличч Фішера в поєднанні з методом АОК (Аналіз основних компонентів) на етапі попередньої обробки.

Метод аналізу головних компонентів – це універсальний статистичний метод, що має безліч практичних застосувань [3]. При використанні в процесі розпізнавання осіб АГК прагне зменшити розмір вихідних даних, зберігаючи при цьому найбільш актуальну інформацію. Він генерує набір зважених власних векторів (рис. 3), які, в свою чергу, створюють власні обличчя – великі набори різних зображень людського обличчя. Лінійна комбінація власних поверхонь представляє кожне зображення в навчальному наборі. АГК використовується для отримання цих власних векторів з коваріаційної матриці навчального набору зображень. Для кожного зображення розраховуються його основні компоненти (від 5 до 200). Інші компоненти кодують незначні відмінності між обличчями та шумом. Процес розпізнавання включає в себе порівняння основного компонента невідомого зображення з компонентами всіх інших зображень.



Рис. 1. Статистична база даних зображень людських облич

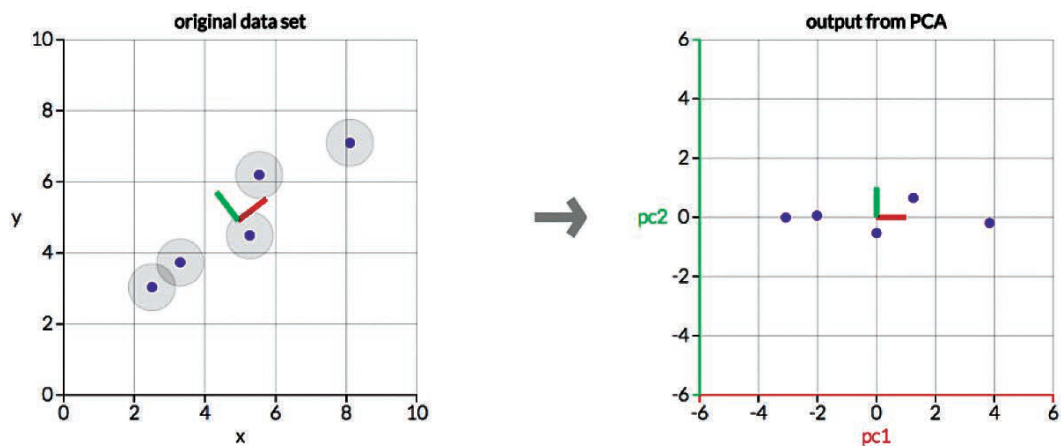


Рис. 2. Вхідні та вихідні дані аналізу головних компонентів

Наведемо основні переваги аналізу основних компонентів:

1. Видаляє корельовані об'єкти.

У реальному сценарії це дуже поширене явище, коли Ви отримуєте Тисячі об'єктів у вашому наборі даних. Ви не можете запустити свій алгоритм на всіх функціях, так як це знизить продуктивність вашого алгоритму, і буде нелегко візуалізувати таку кількість функцій в будь-якому вигляді графіка. Таким чином, ви повинні зменшити кількість об'єктів у вашому наборі даних. Вам потрібно з'ясувати кореляцію між функціями (корельованими змінними). Пошук кореляції вручну в тисячах функцій практично неможливий, розчаровує і забирає багато часу. АГК робить це для вас ефективно. Після реалізації АГК у вашому наборі даних всі основні компоненти незалежні один від одного. Між ними немає ніякої кореляції.

2. Підвищує продуктивність алгоритму.

З такою кількістю функцій продуктивність вашого алгоритму різко знизиться. АГК -це дуже поширений спосіб прискорити ваш алгоритм машинного навчання, позбувшись від корельованих змінних, які не беруть участь у прийнятті будь-яких рішень. Час навчання алгоритмів значно скорочується при меншій кількості функцій. Таким чином, якщо вхідні розміри занадто високі, то використання АГК для прискорення алгоритму є розумним вибором.

3. Зменшує перенавчання.

Переоснащення в основному відбувається, коли в наборі даних занадто багато змінних. Таким чином, АГК допомагає подолати проблему переоснащення за рахунок скорочення кількості функцій.

4. Покращує візуалізацію.

Дуже складно візуалізувати і розуміти дані у великих вимірах. АГК перетворює дані високої розмірності в дані низької розмірності (2 вимірювання), щоб їх можна було легко візуалізувати. Ми можемо використовувати 2D-графік Осипу, щоб побачити, які основні компоненти призводять до високої дисперсії і мають більший вплив в порівнянні з іншими основними компонентами. Навіть найпростіший набір даних райдужної оболонки ока є 4-мірним, що важко візуалізувати. Ми можемо використовувати АГК, щоб зменшити його до 2-х вимірювань для кращої візуалізації.

До недоліків аналізу основних компонентів можна віднести:

1. Незалежні змінні стають менш інтерпретованими. Після реалізації АГК в наборі даних ваші вихідні об'єкти перетворюються в основні компоненти. Основні компоненти-це лінійна комбінація ваших оригінальних функцій. Основні компоненти не так легко читаються і інтерпретуються, як оригінальні функції.

2. Стандартизація даних необхідна перед АГК. Ви повинні стандартизувати свої дані перед впровадженням АГК, в іншому випадку АГК не зможе знайти оптимальні основні компоненти. Наприклад, якщо набір об'єктів містить дані, виражені в одиницях кілограмів, світлових років або мільйонів, масштаб дисперсії в навчальному наборі величезний. Якщо АГК застосовується до такого набору функцій, результуючі навантаження для об'єктів з високою дисперсією також будуть великими. Отже, основні компоненти будуть зміщені в бік функцій з високою дисперсією, що призведе до помилкових результатів. Крім того, для стандартизації всі категоріальні характеристики повинні бути перетворені в числові характеристики, перш ніж можна буде застосувати АГК. На АГК впливає масштаб, тому вам необхідно масштабувати об'єкти у ваших даних перед застосуванням АГК.

3. Втрата інформації. Хоча основні компоненти намагаються охопити максимальну відмінність між об'єктами в наборі даних, якщо ми не будемо ретельно вибирати кількість основних компонентів, в ньому може бути відсутня деяка інформація в порівнянні з вихідним списком об'єктів.

Метод каскадів Хаара – це метод виявлення об'єктів, який використовується для визначення місця розташування об'єктів на зображеннях (рис. 3) [10]. Алгоритм навчається на великій кількості позитивних і негативних зразків-перший містить об'єкт, що цікавить, а другий містить все, що завгодно, крім об'єкта, який ви шукаєте. Після навчання класифікатор може знайти цікавить об'єкт на нових зображеннях. Цей метод використовувався при ідентифікації злочинців у поєднанні з локальним алгоритмом бінарного шаблону для розпізнавання осіб. Каскадний класифікатор Хаара використовує 200 (з 6000) функцій, що забезпечує швидкість розпізнавання 85-95 % навіть при різних виразах.

Згорткові нейронні мережі схожі на звичайні нейронні мережі, але з явним припущенням, що вхідними даними є зображення, що дозволяє дизайнерам кодувати певні властивості в архітектуру [7]. Архітектура ЗНМ складається з послідовності рівнів з найпростішою архітектурою [INPUT-CONV-RELU-POOL-FC].

Вхідний шар містить необроблені значення пікселів зображень, шар CONV складається з ядра або фільтра фіксованого розміру, який ковзає у вигляді вікна для виконання операції згортки на віконному зображенні для вилучення об'єктів. Застосовується прокладка на розмір вхідного зображення, щоб подолати нерівномірне зіставлення з розміром фільтра. RELU розшифровується як випрямлені лінійні одиниці, що являє собою елементарну функцію активації, яка присвоює нульове значення прихованим одиницям. ПУЛ позначає шар об'єднання, який відповідає за зменшення вибірки і зменшення розмірності, що, в свою чергу, зменшує обчислювальну потужність, необхідну для обробки даних.

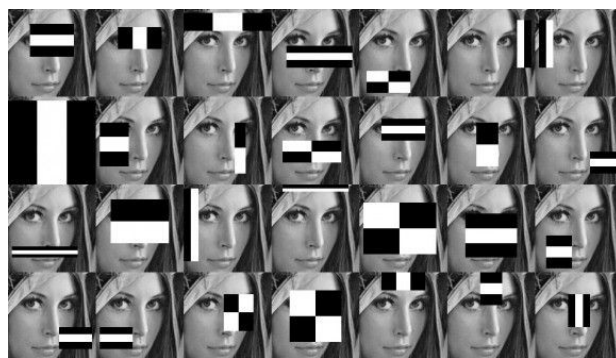


Рис. 3. Загальне уявлення навчання класифікатору Хаара

Шар об'єднання також має ядро або функцію, яка ковзає, як вікно, на вхід, щоб витягти домінуючі функції, які є обертальними і позиційно-інваріантними. Максимальне об'єднання і середнє об'єднання є використовуються дві загальні функції. FC-це повністю підключений шар, де кожен нейрон на вході підключений до кожного

нейрону на виході, і цей шар відповідає за обчислення оцінки певного класу, в результаті чого виходить N виходів, де n позначає кількість класів / категорій, що підлягають класифікації. Клас з максимальним балом визначається як прогнозований клас архітектури ЗНМ. Шар FC також називають щільним шаром.

Можна відзначити, що архітектура ЗНМ може бути змінена в залежності від вимог до дизайну і продуктивності системи. Деякі з інших шари, які використовуються в архітектурі ЗНМ, включають відсів і вирівнювання. Рівень відсіву - це метод регуляризації для запобігання надмірній підгонки ЗНМ, при якому частина вхідних даних (звана коефіцієнтом відсіву) відсівається, встановлюючи їх значення рівними 0 при кожному оновленні під час навчання. Значення збережених вхідних даних масштабуються, так що їх сума залишається незмінною під час навчання. Шари згладжування вводяться перед шаром FC для перетворення двовимірних об'єктів в один вимір.

Переваги згорткової нейронної мережі над звичайною нейронною мережею та її використання для розпізнавання людського обличчя.

Згорткові нейронні мережі повсюдно застосовуються для вирішення різних завдань навчання. Вони досить ефективні для вирішення завдань класифікації зображень. Ми побачимо, що відрізняє згорткові нейронні мережі або ЗНМ від повністю пов'язаних нейронних мереж і чому згорткові нейронні мережі так добре справляються з завданнями класифікації зображень.

Спочатку давайте подивимося на подібності. Як нейронні мережі згортки, так і нейронні мережі мають навчальні ваги і зміщення. В обох мережах нейрони отримують певний ввід, виконують точковий твір і слідує за ним за допомогою нелінійної функції, такої як ReLU (випрямлена лінійна одиниця).

Основна проблема з повністю підключеним шаром:

Коли справа доходить до класифікації зображень - скажімо, розміром $64 \times 64 \times 3$ - повністю підключеним шаром потрібно 12288 ваг в першому прихованому шарі! Кількість ваг буде ще більшою для зображень розміром $225 \times 225 \times 3 = 151875$. Мережі з великою кількістю параметрів стикаються з низкою проблем, наприклад, з більш повільним часом навчання, ймовірністю перенавчання і т. д.

Основна функціональна відмінність нейронної мережі згортки полягає в тому, що основна матриця зображення зводиться до матриці меншої розмірності в найпершому шарі за допомогою операції, званої згорткою. Наприклад, зображення розміром $64 \times 64 \times 3$ може бути зменшено до $1 \times 1 \times 10$. Після чого виконуються наступні операції.

ЗНМ зазвичай складається з наступних компонентів:

- Вхідний шар - в якості вхідного задається одне необроблене зображення. Для зображення RGB його розмір буде $A \times B \times 3$, де 3 представляє кольори червоний, зелений і синій.
- Шар згортки являє собою матрицю розмірності, меншою, ніж вхідна матриця. Він виконує операцію згортки з невеликою частиною вхідної матриці, що має той же розмір. Сума добутків відповідних елементів є результатом цього шару.

ReLU або випрямлена лінійна одиниця - ReLU математично виражається як $\max(0, x)$. Це означає, що будь-яке число менше 0 перетворюється в 0, в той час як будь-яке додатне число може передаватися як є (рис. 4).

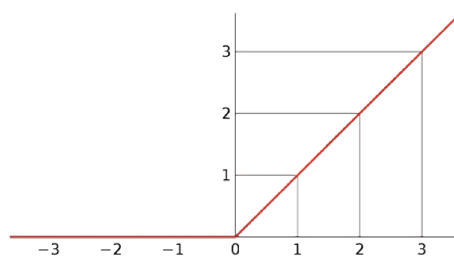


Рис. 4. Функція ReLU

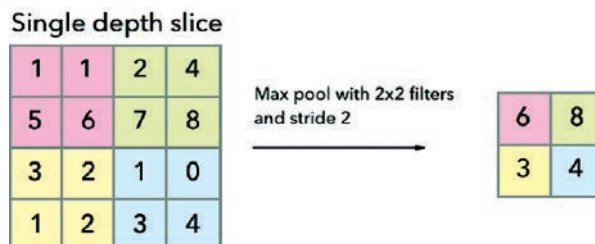


Рис. 5. Функція максимальний пул

Максимальний пул-передає максимальне значення з невеликої колекції елементів вхідної матриці на вихід. Зазвичай це квадратна матриця (рис. 5).

Повністю підключений шар – кінцевий вихідний шар являє собою звичайний повністю підключений шар нейронної мережі, який видає вихідні дані (рис. 6).

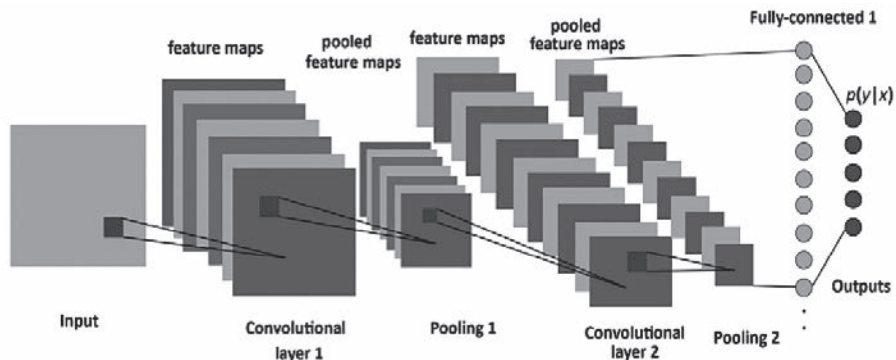


Рис. 6. Згорткова нейронна мережа

Наведемо алгоритм роботи системи розпізнавання обличчя (рис. 7), який включає:

- Захват фрейму(ів) обличчя з відеопотоку.
- Передача фрейму(ів) до згорткової нейронної системи.
- Порівняння обробленого фрейму(ів) обличчя з наявними в базі даних.

Вивід ім'я в рамці навколо обличчя розпізнаної особи чи «Невідомо» навколо обличчя нерозпізнаної особи.

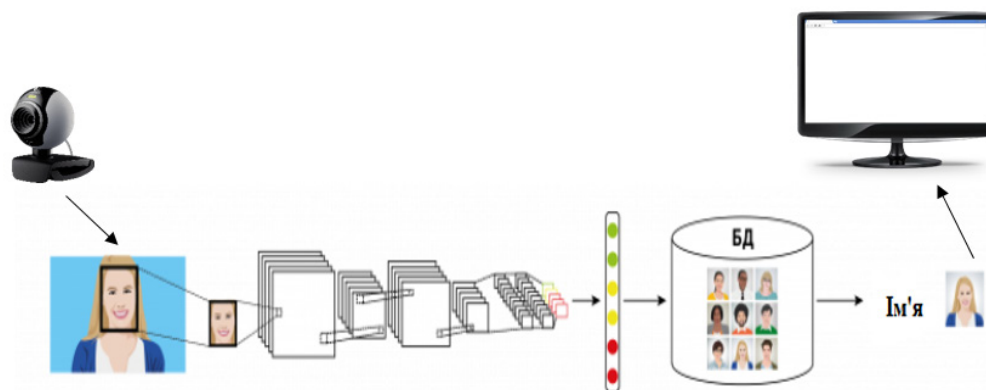


Рис. 7. Алгоритм роботи системи розпізнавання обличчя

Для розробки використовувалася Microsoft Visual Studio 2019. Для розробки внутрішньої логіки було використано мову програмування C# та .Net framework 4.6. Для захвату фрейму відеопотоку для збереження обличчя особи в базу даних була використана бібліотека EmguCV. Бібліотека EmguCV повністю написано на C#. Він може бути запущений на будь-якій платформі, що підтримується. Net, включаючи iOS, Android, Mac OS X, Linux і Windows. Було витрачено багато зусиль на створення чистої реалізації C#, так як заголовки повинні бути перенесені, в порівнянні з керованою реалізацією C++. Вся функціональність і простота використання збережена порівнюючи з бібліотекою OpenCV на C++.

Для сховища даних, а саме для збереження інформації про особу використовується база даних SQLite. SQLite-це бібліотека на мові Сі, яка реалізує невеликий, швидкий, автономний, високонадійний, повнофункціональний механізм бази даних SQL. SQLite – найбільш часто використовуваний движок баз даних у світі. SQLite вбудований у всі мобільні телефони та більшість комп'ютерів і поставляється в комплекті з незліченними іншими додатками, якими люди користуються щодня.

Для розпізнавання людського обличчя у системі використовується Dlib. Dlib – це сучасний інструментарій C++, що містить алгоритми машинного навчання та інструменти для створення складного програмного забезпечення INC++ для вирішення реальних завдань. Він використовується як у промисловості, так і в наукових колах у широкому діапазоні областей, включаючи робототехніку, вбудовані

пристрої, мобільні телефони та великі високопродуктивні обчислювальні середовища. Ліцензування Dlib з відкритим вихідним кодом дозволяє вам безкоштовно використовувати його в будь-якому додатку. Саме для нашої системи із Dlib буде використовуватися інструмент розпізнавання обличчя.

В основі цього інструментарію лежить згортова нейронна система. При використанні dlib отримує точність 99,38% за стандартним тестом розпізнавання обличчя LFW порівняно з іншими сучасними методами розпізнавання осіб станом на Лютий 2021 року. Ця точність означає, що при наявності пари лицьових зображення, інструмент правильно визначить, чи належить пара одному і тому ж людина або від різних людей в 99,38% випадків. При цьому при використанні були ураховані такі показники як освітлення, якість зображення, дистанція до обличчя та кількість облич у фреймі. Також досягнута велика продуктивність роботи системи (рис. 8) в зв'язку з можливістю використання паралельних обчислень з використанням всіх ядер процесора або використання можливостями графічної карти, що демонструє ще кращу продуктивність.

Головне меню системи складається з форми для входу адміністратора, з кнопки входу користувача і кнопки виходу з системи.



Рис. 8. Багато облич в фреймі при розпізнаванні

Користувацьке меню (рис. 9) складається з трьох кнопок для додавання фото в базу даних, видалення фото з бази даних та безпосередньо запуск системи визначення та розпізнавання обличчя з використанням веб камери комп'ютера. Слід зазначити, що кнопки додати та видалити фото увімкнені тільки для адміністратора, для користувача вони ввімкнені.

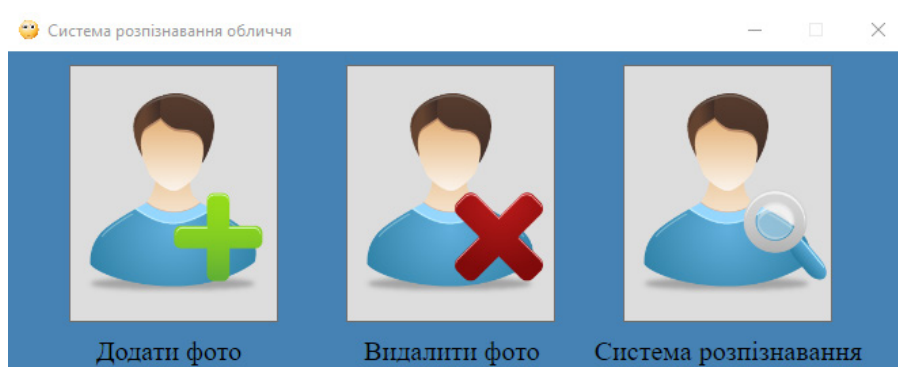


Рис. 9. Користувацьке меню системи розпізнавання обличчя

В вікні додавання фото (рис. 10) знаходиться форма для внесення даних про особу, фото якої буде додано в базу даних системи, також є кнопки «Зберегти», «Оновити», «Завантажити» та «Назад», що відповідно зберігають особу до бази даних, оновлюють фрейм для збереження, завантаження фото з будь-якого джерела даних та вихід в попереднє вікно.

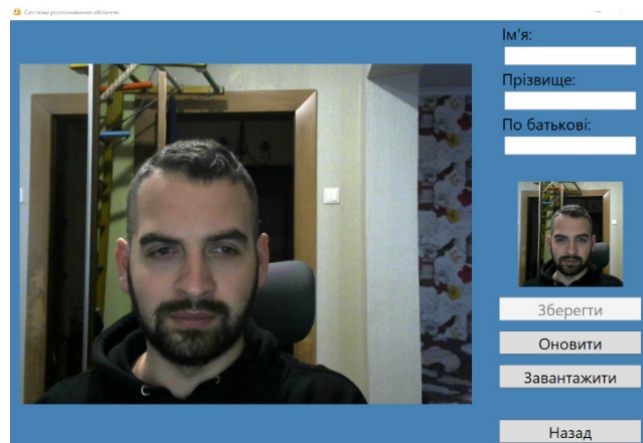


Рис. 10. Вікно додавання особи до бази даних

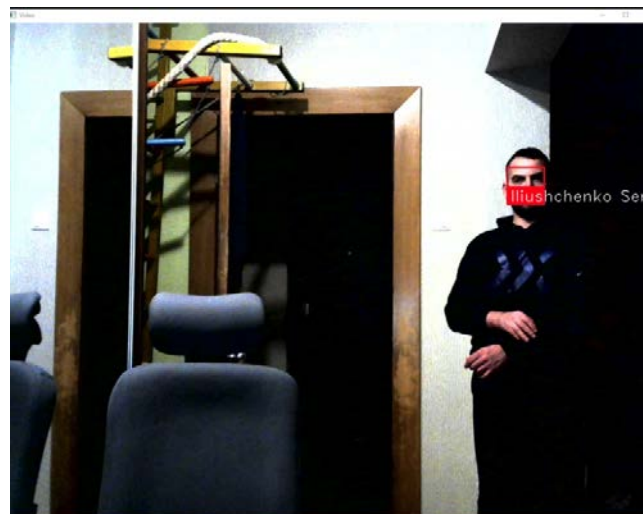


Рис. 11. Вікно розпізнавання обличчя

У вікні видалення фото знаходиться список збережених осіб та кнопки «Видалити» та «Назад», що відповідно означають видалити фото вибраної особи з бази даних та повернення в попереднє меню. У вікні система розпізнавання саме відбувається розпізнавання обличчя і виведення на дисплей даних про знайдену особу в виді рамки навколо обличчя з відеопотоку чи слова «Unkown», якщо особу не було знайдено в існуючій базі даних.

Висновки. В даній роботі проаналізовано наявні алгоритми та системи виявлення та розпізнавання обличчя, зважені їх переваги та недоліки. Розглянуто використання згорткової нейронної системи з метою розпізнавання обличчя. Проаналізовано на практиці відсоток точності розпізнавання людського обличчя та продуктивність, враховуючі такі фактори як освітлення, якість зображення, кількість облич на зображенні використовуючи бібліотеку з відкритим вихідним кодом Face recognition із сімейства бібліотек DLib в основі якої лежить згорткова нейронна мережа.

Список використаних джерел:

1. L. Zhi-fang, Y. Zhi-sheng, A.K.Jain and W. Yun-qiong, 2003, "Face Detection And Facial Feature Extraction In Color Image", Proc. The Fifth International Conference on Computational Intelligence and Multimedia Applications (ICCIMA'03), pp. 126-130, Xi'an, China.
2. C. Lin, 2005, "Face Detection By Color And Multilayer Feedforward Neural Network", Proc. 2005 IEEE International Conference on Information Acquisition, pp. 518-523, Hong Kong and Macau, China.
3. S. Kherchaoui and A. Houacine, 2010, "Face Detection Based On A Model Of The Skin Color With Constraints And Template Matching", Proc. 2010 International Conference on Machine and Web Intelligence, pp. 469-472, Algiers, Algeria.
4. P. Peer, J. Kovac and F. Solina, 2003, "Robust Human Face Detection in Complicated Color Images", Proc. 2010 The 2nd IEEE International Conference on Information Management and Engineering (ICIME), pp. 218-221, Chengdu, China.
5. M. Ş. Bayhan and M. Gökmen, 2008, "Scale And Pose Invariant Real-Time Face Detection And Tracking", Proc. 23rd International Symposium on Computer and Information Sciences ISCIS '08, pp. 1-6, Istanbul, Turkey.

6. C.C. Tsai, W.C. Cheng, J.S. Taur and C.W. Tao, 2006, "Face Detection Using Eigenface And Neural Network", Proc. 2006 IEEE International Conference on Systems, Man, and Cybernetics, pp. 4343-4347, Taipei, Taiwan.
7. X. Liu, G. Geng and X. Wang, 2010, "Automatically Face Detection Based On BP Neural Network And Bayesian Decision", Proc. 2010 Sixth International Conference on Natural Computation (ICNC 2010), pp. 1590-1594, Shandong, China.
8. M. Tayyab and M. F. Zafar, 2009, "Face Detection Using 2D-Discrete Cosine Transform And Back Propagation Neural Network", Proc. 2009 International Conference on Emerging Technologies, pp. 35-39, Islamabad, Pakistan.
9. Палій І.О., Саченко А.О., Антошук С.Г., Бурак Т.О. Нейромережний підхід до комп'ютерного розпізнавання облич // Искусственный интеллект, Вип. 3, 2010. С. 378-387.
10. Тимошин Ю. А., Орленко С. П. Алгоритм розпізнавання обличчя людей на базі згорткової нейронної мережі // Міжвідомчий науково-технічний збірник «Адаптивні системи автоматичного управління», Вип. 1 (32), 2018. С. 166-173.
11. Ляшенко Г.Є., Даниленко О.І. Дослідження методів розпізнавання облич // Міжнародна науково-практична конференція High-Technologies in infocommunications 23-25 травня 2019 р., Харків – Кам'янець-Подільський, Україна. С. 85-86.
12. Загородня Д. І., Палій І. О., Крилов В. М. Метод розпізнавання облич за характерними точками контуру в комп'ютерних системах відеоспостереження // МНПК «Сучасні інформаційні та електронні технології». Одеса, 27-31 травня 2013 р. С. 49-52.
13. Ділай В.І. Огляд методів розпізнавання облич для використання в системах контролю і управління доступом // Матеріали VII Міжнародної науково-технічної конференції молодих учених та студентів. Актуальні задачі сучасних технологій – Тернопіль 28-29 листопада 2018. С. 48-49.

References:

1. L. Zhi-fang, Y. Zhi-sheng, A.K.Jain and W. Yun-qiong, 2003, "Face Detection And Facial Feature Extraction In Color Image", Proc. The Fifth International Conference on Computational Intelligence and Multimedia Applications (ICCIMA'03), pp. 126-130, Xi'an, China [in English].
2. C. Lin, 2005, "Face Detection By Color And Multilayer Feedforward Neural Network", Proc. 2005 IEEE International Conference on Information Acquisition, pp. 518-523, Hong Kong and Macau, China [in English].
3. S. Kherchaoui and A. Houacine, 2010, "Face Detection Based On A Model Of The Skin Color With Constraints And Template Matching", Proc. 2010 International Conference on Machine and Web Intelligence, pp. 469-472, Algiers, Algeria [in English].
4. P. Peer, J. Kovac and F. Solina, 2003, "Robust Human Face Detection in Complicated Color Images", Proc. 2010 The 2nd IEEE International Conference on Information Management and Engineering (ICIME), pp. 218-221, Chengdu, China [in English].
5. M. Ş. Bayhan and M. Gökmen, 2008, "Scale And Pose Invariant Real-Time Face Detection And Tracking", Proc. 23rd International Symposium on Computer and Information Sciences ISCIS '08, pp. 1-6, Istanbul, Turkey [in English].
6. C.C. Tsai, W.C. Cheng, J.S. Taur and C.W. Tao, 2006, "Face Detection Using Eigenface And Neural Network", Proc. 2006 IEEE International Conference on Systems, Man, and Cybernetics, pp. 4343-4347, Taipei, Taiwan [in English].
7. X. Liu, G. Geng and X. Wang, 2010, "Automatically Face Detection Based On BP Neural Network And Bayesian Decision", Proc. 2010 Sixth International Conference on Natural Computation (ICNC 2010), pp. 1590-1594, Shandong, China [in English].
8. M. Tayyab and M. F. Zafar, 2009, "Face Detection Using 2D-Discrete Cosine Transform And Back Propagation Neural Network", Proc. 2009 International Conference on Emerging Technologies, pp. 35-39, Islamabad, Pakistan [in English].
9. Paliy I.O., Sachenko A.O., Antoshchuk S.H., Burak T.O. (2010) Neiomerezhnyi pidkhid do kompiuternoho rozpoznavannia oblych [Neural network approach to computer facial recognition] *Iskusstvennyi intellekt – Artificial Intelligence*, Vyp. 3, S. 378-387 [in Ukrainian].
10. Tymoshyn Yu. A., Orlenko S. P. (2018) Alhorytm rozpoznavannia oblychchia liudei na bazi zghortkovoї neuronnoi merezhi [Algorithm for recognizing the face of people on the basis of convolutional neural network] *Mizhvidomchyi naukovo-tekhnichnyi zbirnyk «Adaptyvni systemy avtomatychnoho upravlinnia» – Interdepartmental scientific and technical collection "Adaptive automatic control systems"*, Vyp. 1 (32), S. 166-173 [in Ukrainian].
11. Liashenko H.Ie., Danylenko O.I. (2019) Doslidzhennia metodiv rozpoznavannia oblych [Research of face recognition methods] *Mizhnarodna naukovo-praktychna konferentsiia High-Technologies in infocommunications – International scientific-practical conference High-Technologies in infocommunications 23-25 travnia 2019 r.*, Kharkiv – Kamianets-Podilskyi, Ukraina. S. 85-86 [in Ukrainian].
12. Zahorodnia D. I., Paliy I. O., Krylov V. M. (2013) Metod rozpoznavannia oblych za kharakternymi tochkamy konturu v kompiuternykh systemakh videosposterezhennia [Method of face recognition by characteristic contour points in computer video surveillance systems] *MNPK «Suchasni informatsiini ta elektronni tekhnologii» – MNPK "Modern information and electronic technologies"*. Odesa, 27-31 travnia 2013 r. S. 49-52 [in Ukrainian].
13. Dilai V.I. (2018) Ohliad metodiv rozpoznavannia oblych dlia vykorystannia v systemakh kontroliu i upravlinnia dostupom [Review of face recognition methods for use in access control and management systems] *Materialy VII Mizhnarodnoi naukovo-tekhnichnoi konferentsii molodykh uchenykh ta studentiv. Aktualni zadachi suchasnykh tekhnologii – Proceedings of the VII International Scientific and Technical Conference of Young Scientists and Students. Actual problems of modern technologies* Ternopil 28-29 lystopada 2018. S. 48-49 [in Ukrainian].

УДК 35.077:347.97/99:004.8

DOI <https://doi.org/10.32689/maup.it.2022.1.2>

Віктор БОЙКО

кандидат технічних наук, доцент, доцент кафедри кібербезпеки, Національний університет «Одеська юридична академія», вул. Рішельєвська, 28, Одеса, Україна, індекс 65011 (boyko-work@ukr.net)

ORCID: 0000-0001-5929-657X

Микола ВАСИЛЕНКО

доктор фізико-математичних наук, доктор юридичних наук, професор, професор кафедри кібербезпеки, Національний університет «Одеська юридична академія», вул. Рішельєвська, 28, Одеса, Україна, індекс 65011 (vasylenko.it@journals.maup.kiev.ua)

ORCID: 0000-0002-8555-5712

Валерій РАЧУК

асистент кафедри кібербезпеки, Національний університет «Одеська юридична академія», вул. Рішельєвська, 28, Одеса, Україна, індекс 65011 (rachuk960@gmail.com)

ORCID: 0000-0003-1793-016X

Валерія СЛАТВІНСЬКА

викладач кафедри кримінального права, процесу та криміналістики, Міжнародний гуманітарний університет, Фонтанська дорога, 33, Одеса, Україна, індекс 65000 (slatvinskaya_valeriya@ukr.net)

ORCID: 0000-0002-6082-981X

Viktor BOYKO

Candidate of Technical Sciences, Associate Professor, Associate Professor at the Department of Cybersecurity, National University "Odesa Law Academy", 28 Richelevska str., Odesa, Ukraine, postal code 65011 (boyko-work@ukr.net)

Nikolai VASILENKO

Doctor of Physical and Mathematical Sciences, Doctor of Law, Professor, Professor at the Department of Cybersecurity, National University "Odesa Law Academy", 28 Richelevska str., Odesa, Ukraine, postal code 65011 (vasylenko.it@journals.maup.kiev.ua)

Valery RACHUK

Assistant Professor at the Department of Cybersecurity, National University "Odesa Law Academy", 28 Richelevska str., Odesa, Ukraine, postal code 65011 (rachuk960@gmail.com)

Valeria SLATVINSKAYA

Lecturer at the Department of Criminal Law, Procedure and Criminalistics, International Humanitarian University, 33 Fontanska doroga str., Odessa, Ukraine, postal code 65000 (slatvinskaya_valeriya@ukr.net)

Бібліографічний опис статті: Бойко В., Василенко М., Рачук В., Слатвінська В. Нове (друге) народження мови «ПРОЛОГ» (PROLOG) в контексті систем підтримки прийняття рішень. *Інформаційні технології та суспільство*. 2022. Вип. 1 (3). 16–22. DOI: <https://doi.org/10.32689/maup.it.2022.1.2>

Bibliographic description of the article: Boyko, V., Vasilenko, M., Rachuk, V., Slatvinskaya, V. (2022). Nove (druhe) narodzhennia movy «PROLOH» (PROLOG) v konteksti system pidtrymky pryiniattia rishen [The New (second) birth of the "Prolog" language in the context of Decision Support Systems]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, 1 (3), 16–22. DOI: <https://doi.org/10.32689/maup.it.2022.1.2>

**НОВЕ (ДРУГЕ) НАРОДЖЕННЯ МОВИ «ПРОЛОГ» (PROLOG)
В КОНТЕКСТІ СИСТЕМ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ**

Анотація. Штучний інтелект і машинне навчання мають вагому частку в сучасних інформаційних технологіях і надають широкий набір інструментів: від експертних систем до нейромереж. Їх реалізація потребує специ-

фічного середовища та відповідного програмного забезпечення. Однією із мов програмування для цього підходить мова «Пролог». В той же час існують серйозні ризики щодо помилок при використанні цього забезпечення. Обговорено диференціацію завдань з управління у таких складних соціально- та організаційно-технічних системах як штучний інтелект. В статті надано всебічний аналіз і осмислення значення мови «Пролог» в системі підтримки рішень (DSS) та проведено встановлення його місця в цій агломерації. В зв'язку з цим обґрунтовано необхідність створення DSS нового покоління. В контексті поставленої мети показано перспективність використання DSS для прийняття стратегічних рішень в дедуктивному навчанні в порівнянні з індуктивним. Доведено, що саме такою стратегічною технологією стає DSS. Воно використовує дедуктивне навчання, яке при використанні мови «Пролог» в процесі формування систем штучного інтелекту створює максимальну прозорість і надає обґрунтованість при прийнятті рішень. Акцентовано увагу на тому, чому мова «Пролог» має стати перспективною мовою систем AI/ML. Пропонується також концепція гібридної DSS, що поєднує в собі переваги обох систем. Така система дозволяє приймати рішення на різних рівнях, отримуючи вигоди від систем з індуктивним навчанням на тактичному рівні та дедуктивних систем на стратегічному рівні прийняття рішень.

Ключові слова: штучний інтелект, мова «Пролог», система підтримки рішень, дедуктивне, індуктивне, гібридне машинне навчання.

THE NEW (SECOND) BIRTH OF THE “PROLOG” LANGUAGE IN THE CONTEXT OF DECISION SUPPORT SYSTEMS

Abstract. Artificial intelligence and machine learning have a significant share in modern information technologies and provide a wide range of tools: from expert systems to neural networks. Their implementation requires a specific environment and appropriate software. One of the programming languages suitable for this is the Prologue language. At the same time, there are serious risks of errors when using this software. Differentiation of management tasks in such complex socio – organizational and technical systems as artificial intelligence is discussed. The article provides a comprehensive analysis and understanding of the meaning of the Prologue language in the decision support system (DSS) and establishes its place in this agglomeration. In this regard, the need to create a new generation of DSS is justified. In the context of this goal, the prospects of using DSS for strategic decision-making in deductive learning in comparison with inductive learning are shown. It is proved that DSS is becoming such a strategic technology. It uses deductive learning, which, when using the Prologue language in the process of forming artificial intelligence systems, creates maximum transparency and gives validity in decision-making. Attention is focused on why the Prologue language should become a promising language for AI/ML systems. The concept of hybrid DSS is also proposed, combining the advantages of both systems. Such a system allows you to make decisions at different levels, benefiting from systems with inductive learning at the tactical level and deductive systems at the strategic level of decision-making.

Key words: artificial intelligence, “Prolog”, decision support system, deductive, inductive, hybrid machine learning.

Актуальність проблеми. Натепер отримали широке поширення системи підтримки прийняття рішень (Decision Support Systems – DSS) різного класу, основним призначенням яких є збір інформації з поставленої проблеми вимагає вирішення, вироблення і аналіз можливих альтернатив рішення для полегшення прийняття рішень у складних соціально- та організаційно-технічних системах (COTC). До такого класу, наприклад, належать системи управління розумним містом Alibaba's City Brain [1]. Збір та аналіз інформації в COTC тісно пов'язаний з такими сферами знань, як датамайнінг, Big Data, системи штучного інтелекту (Artificial Intelligence – AI) і машинне навчання (Machine Learning – ML). Оскільки даних збирається найчастіше більше, ніж може бути оброблено вручну або в напівавтоматизованому режимі, в DSS достатньо широко використовуються системи аналізу даних на основі ML.

Не викликає сумнівів, що AI становить галузь, визначення якої залежить від того, хто саме робить це визначення, враховуючи його багатофункціональність на всіх етапах створення. AI й ML зараз надають широкий набір інструментів: від експертних систем до нейромереж. Вони використовуються для вирішення проблем, пов'язаних з великими витратами обчислювальних ресурсів або ті, що погано піддаються алгоритмізації. Слушно відзначити й те, що питання програмного забезпечення набуває принципового значення, коли створювати ефективні механізми контролю над внутрішньою роботою інформаційно-комунікаційних технологій та алгоритмів, що обробляють дані, а також коли слід давати гарантії щодо належного контролю за функціонуванням систем та підзвітністю за використанням технологій AI, а саме, в такій галузі як і ML. Реалізація цього потребує специфічного середовища та відповідного програмного забезпечення. З іншого боку, існує занепокоєння тим, що в результаті розвитку технологій AI виникнуть нові ризики через відповідні помилки, які поставлять під загрозу існування людського виду у цілому.

Метою статті є встановлення місця і осмислення значення мови «Пролог» в системі підтримки прийняття рішень.

Аналіз останніх досліджень і публікацій. В роботі [2] йдеться про те, яким чином впроваджують глибинне навчання, коли алгоритм здатний самостійно визначати набір ознак для розрізнення різних категорій даних. Втручання людини при обробці даних не потрібно, що відкриває набагато більше можливостей при написанні програм щодо систем AI. Існує певний процедурний підхід який дозволяє розглядати цільове твердження програми, як виклик однієї процедури, або як послідовність викликів кількох процедур користувача та стандартних. Послідовність викликів процедур задається порядком запису ці-

лей у кон'юнкції. Аргументи цільових тверджень трактуються як фактичні параметри процедур. Умовні твердження програми та факти, що записані одним предикатом так і називають процедурою. Тіло може містити виклики інших процедур, які виконують необхідні для основної процедури дії, одержують дані з фактів тощо. Якщо умовне твердження має кілька гілок умов, то процедура розгалужується. У кожній гілці є одна або кілька поточних цілей, які задають умови виконання гілки. Такі цілі розташовують на початку гілки, інші поточні цілі гілки виконують необхідні за умовами дії або викликають інші процедури, що виконуватимуть необхідні дії. Серед поточних цілей гілок можуть заходитися виклики процедур до себе, тобто процедура може виконувати дії, які повторюються рекурсивно. Тут доречними стає використання індуктивних та дедуктивних систем ML, які визначаються як найбільш розповсюджені. Індуктивне навчання (навчання за прецедентами) використовує навчальну вибірку (базу даних прецедентів з відомими виходами) з якої різними шляхами можуть бути отримані емпіричні правила передбачення результатів за цією вибіркою. У такому навчанні виділяють приблизно десять основних базових алгоритмів, а саме: k-Means – найбільш популярний метод кластеризації, який прагне мінімізувати сумарне квадратичне відхилення точок кластерів від центрів цих кластерів, SVM – метод опорних векторів – алгоритми навчання з учителем для завдань класифікації та регресійного аналізу, C4.5 – алгоритм для побудови дерев рішень, розроблений Джоном Квінлоу Apriori, EM, PceRank, AdaBoost, kNN, наївний баєсівський класифікатор, CART [3; 4]. Перевагою індуктивного навчання є можливість автоматизованої обробки великих обсягів інформації майже без участі людини. Це стало однією з вирішальних переваг в цей час при впровадженні систем ML в різних областях використання DSS, оскільки в сучасних інформаційних системах кількість доступної до обробки інформації не є критичним або обмежувальним фактором – широке поширення датчиків, систем геопозиціонування і машинного зору надає величезну кількість доступних для використання даних (див. [5; 6]). Однак, все частіше звертають уваги на недоліки властиві системам ML, заснованим на індуктивних моделях навчання. Зокрема, такі системи дуже чутливі до якості навчальної вибірки – якість отриманих емпіричних закономірностей сильно залежить від відповідності навчальній вибірці ситуації в реальному світі [7]. З якістю вибірки також безпосередньо пов'язана необхідність очищення та вибору необхідних даних [8], вибір тих параметрів, які найбільшою мірою являють собою інформацію, необхідну для вироблення рішень [6] тощо. Людський фактор у таких системах хоча і неявно все ще відіграє більшу роль – з чим пов'язаний великий попит на фахівців у галузі аналізу та обробки даних (Analytics and Data Science).

Системи з дедуктивним навчанням являють собою системи різного роду, засновані на отриманні знань не емпіричним шляхом, з навчальної вибірки прецедентів, а безпосередньо від експерта (фахівця) в предметній галузі.

Виклад основного матеріалу. Принциповим розумінням стало те, що в системах з дедуктивним навчанням далі треба переводити знання в формалізовану форму (символічне уявлення, системи, що породжують правила, фрейми, графи, дерева, асоціативні мережі тощо) і переносяться в базу знань. Так, у роботі [6] виділяються такі стадії набуття знань:

- ідентифікація – аналіз основних характеристик проблеми,
- концептуалізація – розробка концепції представлення знань,
- формалізація – розробка структури організації знань,
- реалізація – формулювання правил, що містять в собі знання.

Все завершується тестуванням, яке за результатами уточнює характеристики всіх стадій процесу набуття знань.

На відміну від першої категорії систем, системи з дедуктивним навчанням вимагають наявності відразу декількох кваліфікованих фахівців (експерта, інженера по роботі зі знаннями), а сам процес передачі знань вибудовується досить трудомістко. Тому, системи DSS, що використовують дедуктивні системи AI/ML в якийсь момент відійшли на другий план, поступившись місцем індуктивним системам, як простішим і таким, що не вимагає високої кваліфікації персоналу, а значить дешевшим і простішим у впровадженні. Однак стало помітним, що намітилося зростання інтересу саме до цього класу систем, що пов'язано з все більшою роллю, яку DSS почали грати в процесі планування й управління великомасштабними COTS. *Категоріальна диференціація завдань з управління COTS.* Не всі рішення, які потребують використання DSS потребують максимально швидкої реакції. Якщо прийняти в якості критерію використання методів прийняття рішень час на пошук рішення і ймовірність безпомилкового рішення, то в підтримці прийняття рішень у реальному світі можна виділити дві основних категорії, два умовних «полюси» прийняття рішень. Якщо розглянути проблему управління і прийняття рішень для такої COTS, як міська інфраструктура «розумного міста» (smart city), то можна виділити в ній дві принципово різні категорії завдань з прийняття рішень [9; 10]. До першої категорії належать завдання, що потребують швидкої реакції. Прийнятним прикладом є завдання управління дорожнім рухом. Це завдання

вимагає швидкої реакції – інтелектуальна транспортна система (ITS) повинна відстежувати і реагувати на завдання і проблеми, що виникають і потребують вирішення в режимі реального часу: аналіз дорожньо-транспортних пригод, керування транспортними потоками в умовах перевантаження, супровід службових машин прогнозування ймовірності виникнення аварій з використанням алгоритмів великих даних тощо. Наприклад, використання системи ET City Brain в регіоні Ханчжоу Сяошань дозволило відмовитися від поліцейського ескорту для машин швидкої допомоги при русі в умовах заторів. Система прогнозує обстановку на вулицях ще до виїзду швидкої допомоги, визначає оптимальну траєкторію руху і на всьому протязі маршруту забезпечує рух наскрізною роботою світлофорів на перехрестях. Це дозволяє досягти скорочення часу реакції вдвічі [11]. У таких системах дані отримуються з різномірних ресурсів, таких як високопродуктивні IoT-пристрої, відеодетектори і GPS. Широко використовуються методи BigData онлайн-моніторинг і моніторинг керованих систем в реальному часі. Основний прогрес в цих областях досягається в основному внаслідок збільшення і розширення можливостей обчислювальних систем, знаходження ефективних алгоритмів пошуку оптимальних рішень, скорочення часу реакції і знаходження рішення і так далі [12]. Друга категорія завдань управління міською інфраструктурою являє собою завдання довготривалого планування, в яких час реакції не є критичним, проте на перший план виходить максимальна деталізація і безпомилковість прийнятих рішень: створення та планування забудови, розвиток міської інфраструктури з урахуванням можливого забруднення повітря, планування та впровадження нових транспортних та інфраструктурних проєктів, розширення міської забудови з урахуванням зростання населення тощо. Як було сказано вище, для таких завдань характерно зміщення фокусу з часу реакції на зменшення кількості помилок, оскільки процес планування з одного боку являє собою тривалу процедуру, з іншого боку, допущені в процесі планування помилки важче виявляються, обходяться дорожче на кілька порядків і важче виявляються – що безпосередньо пов'язано з довготривалістю планування і масштабністю прийнятих рішень – якщо дорога або трубопровід прокладені в невдалому місці або з невірним урахуванням динаміки розвитку міста, виправлення помилки може обійтися вельми дорого [5; 9]. Деякі автори, безпосередньо говорять про існування різних рівнів у вирішенні завдань – зокрема про стратегічне і тактичне планування в різних областях роботи з масштабними COTS [13] в яких повторюється поділ завдань щодо співвідношення часу на рішення до безпомилковості рішення. При цьому важливу роль відіграє не стільки час прийняття рішення, скільки можливість прийняття оптимального в даній ситуації рішення.

Необхідність DSS нового покоління. Необхідність вирішення завдань стратегічного рівня, зумовила інтерес до DSS нового покоління. У таких системах робиться акцент на поясненість рішень AI/ML і приділяється велика увага обґрунтуванню цих рішень.

У роботі [14] зазначається, що в наявних моделях ML завжди є ймовірність «катастрофічної помилки» – збою або прийняття неправильного рішення з різних причин. Це може бути, наприклад, недостовірність навчальної вибірки, на якій проводилося тренування системи. При цьому сама недостовірність може бути обумовлена різними причинами: невірним вибором вектора ознак навчальних прецедентів, невідповідністю вибірки реальному стану речей, застаріванням вибірки стосовно динамічно змінюваного стану речей. У результаті завжди існує ризик серйозних помилок [15] і ціна цього ризику може бути досить великою, особливо у випадку, коли система використовується для вироблення стратегічних рішень у складних COTS. Все це зумовило необхідність розуміння поведінки ML і механізмів прийняття рішення, необхідне для аналізу потенційних збоїв, корекції поведінки та критеріїв прийняття рішень ML і так далі.

Загалом загальні ризики використання «базових моделей» (foundation models) AI/ML як з точки зору ймовірних помилок, так і з точки зору соціальних аспектів проблеми докладно розглядаються в статті [16].

Так само на подібні ризики звертають увагу і великі гравці ринку. Наприклад, у 2019 році агенція DARPA опублікувала дані про проєкт «поясненого штучного інтелекту» (explainable artificial intelligence – XAI) – основним напрямком якого, є створення нової системи, акцент в якій робиться на зрозумілість і обґрунтованість запропонованих системою рішень [17]. У XAI виділяють три основні напрямки: вивчення моделей AI/ML, які дозволяють створювати зрозумілі рішення, створення ефективних людино-машинних інтерфейсів і розробка психологічних принципів і моделей для підвищення ефективного розуміння таких пояснень особами залученими в процесі прийняття рішень.

У роботі [17] також проводиться диференціація вирішуваних завдань аналогічна попередньому розділу, однак тут завдання розподілені не за критерієм час-безпомилковість, а за критерієм час-пояснюваність. Найбільш швидкодійними системами є нейронні мережі та системи глибокого навчання (Deep Learning), проте їх рішення є також найменш прозорими для користувача такою системою.

Концепція гібридної DSS. З нашого аналізу випливає, що для рішень, які приймаються в складних COTS і потребують стратегічного підходу, все частіше потрібно не просто вироблення, а й пояснення та обґрунтування вироблюваних системою рішень. Є перспективним використання гібридної DSS,

в якій генерація вихідних альтернатив на тактичному рівні може виконуватися з використанням традиційних, базових моделей, а побудова стратегічних рішень – з використанням методології ХАІ, а далі коригуватися і доповнюватися з використанням наявних технологій в тому числі технологій використовують дедуктивний підхід. Деякі напрацювання такого підходу наведено у статті [18].

Використання концепції гібридної DSS може дозволити працювати у двох основних режимах прийняття рішень. У режимі швидкого реагування (при невисокій точності) у випадках потребують негайної генерації можливих результатів і впливів на ситуацію DSS буде працювати в режимі «звичайної», «традиційної» системи AI/ML. За аналогією з [13] будемо називати таку систему прийняття рішень «тактичною».

У режимі повільного планування (при високій точності та прозорості результатів) «звичайний», «традиційний» режим роботи DSS буде використовуватися на вході для попередньої обробки інформації та генерації перед-альтернатив. В основному ж режимі повинні використовуватися системи з дедуктивним навчанням, а сам процес вироблення рішень налаштовуватися і відбуватися в тому числі за участю експертів та інженерів знань. При цьому DSS може використовувати для зберігання та використання знань засоби логічного та формального виводу – таким же чином, як це відбувається в системах символічних обчислень CAS Maxima або бібліотеці символічних обчислень SymPy. Таким чином, процес прийняття рішень набуде необхідної прозорості та поясненості, що позитивно позначиться на точності. Тимчасові витрати на прийняття рішення зростуть, однак, з урахуванням невисокої частоти прийняття подібних рішень і ризиків, пов'язаних з помилковими рішеннями, таке співвідношення має виявитися цілком прийнятним.

Мова «Пролог» як перспективна мова систем AI/ML. Отже, з урахуванням наведеного вище можна стверджувати, що завдяки інтересу до систем ХАІ й дедуктивних AI/ML систем мова «Пролог» нині переживає друге народження. При цьому мається на увазі, що більшість сучасних мов програмування використовують імперативний підхід, який базується на описі алгоритму, що дозволяє досягти потрібного результату. Однак, існує альтернативний, а саме, декларативний підхід, коли замість алгоритму дій описується готовий результат, а пошук дій для досягнення результату перекладається на інструментальну базу мови програмування. Такий підхід, зокрема, характерний для мов роботи з системами управління базами даних. Наприклад, SQL-запит сам по собі не містить алгоритму вилучення даних з таблиць, він тільки описує необхідний результат («отримати зведену таблицю адрес, відсортовану по місту проживання клієнта») і умови, необхідні для його отримання («поєднати таблиці клієнтів та їх адрес за загальним ключем клієнта»). Парадигма декларативного підходу добре підходить для користувачів, які не є фахівцями в предметній галузі, і отримала широке поширення за межами програмування як такого [19].

Пролог також використовує декларативний підхід і роботу в парадигмі логічного програмування з використанням логіки предикатів. Логічна нотація Прологу заснована на обчисленні предикатів першого порядку (часто званому просто – «логікою першого порядку», яка допускає висловлювання щодо змінних, фіксованих функцій і предикатів) [20].

Це полегшує побудову на його основі систем DSS і робить її перспективною мовою для побудови нової гібридної системи DSS для прийняття стратегічних рішень для управління складною СОТС.

Старі версії «Прологу» були орієнтовані в основному на академічні та фундаментальні дослідження, що зумовило репутацію мови, як суто академічного інструменту. Однак, сучасні версії мови (зокрема – SWI-Prolog) володіють усім необхідним набором інструментів. Наприклад, до його складу входить безліч розширень і бібліотек, для роботи з багатопотоковими завданнями, шифруванням, такими як TCP/IP, TIPC, ODBC, SGML/XML/HTML, RDF, JSON, YAML, HTTP, засобами машинної графіки тощо. SWI-Prolog дозволяє вбудовування, як у мови середнього і низького рівня (C), так і інтеграцію з мовами надвисокого рівня (Python), а також зі складними середовищами розробки і моделювання (Jupyter) при цьому вбудовування і використання може відбуватися в обидва боки: як система «Пролог» може розширяться бібліотеками мови C, так і мова C може бути розширена коштом інструментів написаних на мові «Пролог».

Висновки та перспективи подальших досліджень. Таким чином, цифровізація прийняття рішень, що розширюється, і використання систем підтримки рішень (в процесах прийняття стратегічних рішень вимагають іншого підходу до прийняття рішень, ніж прийнятий у конвенційних DSS, здебільшого заснованих на індуктивному навчанні. Таким підходом може стати як створення дедуктивної «стратегічної» DSS, що використовує дедуктивне навчання і максимальну прозорість і обґрунтованість прийняття рішення, так і пропонується в статті концепція гібридної DSS, що поєднує в собі переваги обох систем. Така система дозволить приймати рішення на різних рівнях, отримуючи вигоди від систем з індуктивним навчанням на тактичному рівні та дедуктивних систем на стратегічному рівні прийняття рішень.

Список використаних джерел:

1. Caprotti F, Liu D. Platform urbanism and the chinese smart city: The co-production and territorialisation of hangzhou city brain // GeoJournal. – Springer, 2020. – P. 1–15.

2. Василенко М. Д., Шевченко Т.В. Застосування штучного інтелекту в публічному управлінні, судочинстві та правоохоронній діяльності: міждисциплінарне дослідження. *Право і суспільство*. 2021. № 5. С. 148-154.
3. Wu X., Kumar V., Quinlan J. R., Ghosh J., Yang Q., Motoda H., McLachlan G. J., Ng A., Liu B., Philip S. Y., others. Top 10 algorithms in data mining // *Knowledge and information systems*. – Springer, 2008. – Vol. 14, no. 1. – P. 1–37.
4. Settouti N., Bechar M. E. A., Chikh M. A. Statistical comparisons of the top 10 algorithms in data mining for classification task // *International Journal of Interactive Multimedia and Artificial Intelligence*. – 2016. – Vol. 4, no. 1. – P. 46–51.
5. Jordan M. I., Mitchell T. M. Machine learning: Trends, perspectives, and prospects // *Science*. – American Association for the Advancement of Science, 2015. – Vol. 349, No. 6245. – P. 255–260.
6. Jackson P. Introduction to expert systems. – Addison-Wesley Pub. Co., Reading, MA, 1986.
7. Kuwajima H., Yasuoka H., Nakae T. Engineering problems in machine learning systems // *Machine Learning*. – Springer Science; Business Media LLC, 2020. – Vol. 109, no. 5. – P. 1103–1126.
8. McCallum Q. E. Bad data handbook: Cleaning up the data so you can get back to work. – "O'Reilly Media, Inc.", 2012.
9. Hadi Amini M., Shafie-khah M. Cyberphysical smart cities infrastructures: Optimal operation and intelligent decision making / 1st ed. – Wiley, 2022.
10. Ahmad K., Maabreh M., Ghaly M., Khan K., Qadir J., Al-Fuqaha A. Developing future human-centered smart cities: Critical analysis of smart city security, data management, and ethical challenges // *Computer Science Review*. – Elsevier BV, 2022. – Vol. 43. – P. 100452.
11. Jazzyyear. The Chronicles of Cloud Building in Hangzhou: Part 2 / Alibaba Cloud Community. – 2020.
12. Usman M., Jan M. A., He X., Chen J. A survey on big multimedia data processing and management in smart cities // *ACM Computing Surveys (CSUR)*. – ACM New York, NY, USA, 2019. – Vol. 52, no. 3. – P. 1–29.
13. Pérez A. T. E., Camargo M., Rincón P. C. N., Marchant M. A. Key challenges and requirements for sustainable and industrialized biorefinery supply chain design and management: A bibliographic analysis // *Renewable and Sustainable Energy Reviews*. – Elsevier BV, 2017. – Vol. 69. – P. 350–359.
14. Ignatiev A. Towards trustable explainable AI / *IJCAI*. – 2020.
15. Szegedy C., Zaremba W., Sutskever I., Bruna J., Erhan D., Goodfellow I., Fergus R. Intriguing properties of neural networks // arXiv preprint arXiv:1312.6199. – 2013.
16. Bommasani R., Hudson D. A., Adeli E., Altman R., Arora S., et al. On the opportunities and risks of foundation models. – 2021.
17. Gunning D., Aha D. DARPA's explainable artificial intelligence (XAI) program // *AI Magazine*. – Association for the Advancement of Artificial Intelligence (AAAI), 2019. – Vol. 40, no. 2. – P. 44–58.
18. Honda H., Hagiwara M. Analogical reasoning with deep learning-based symbolic processing // *IEEE Access*. – Institute of Electrical; Electronics Engineers (IEEE), 2021. – Vol. 9. – P. 121859–121870.
19. Flach P., Sokol K. Simply logical: Intelligent reasoning by example – online edition. – Zenodo, 2018.
20. Wielemaker J., Schrijvers T., Triska M., Lager T. SWI-Prolog // *Theory and Practice of Logic Programming*. – 2012. – Vol. 12, no. 1-2. – P. 67–96.

References:

1. Caprotti F., Liu D. (2020). Platform urbanism and the chinese smart city: The co-production and territorialisation of hangzhou city brain. *GeoJournal*. Springer, 1–15. [in English].
2. Vasilenko M. D., Shevchenko T.V. (2021). Application of artificial intelligence in public administration, legal proceedings and law enforcement: an interdisciplinary study. *Law and society*, No 5, 148-154. [in Ukrainian].
3. Wu X., Kumar V., Quinlan J. R., Ghosh J., Yang Q., Motoda H., McLachlan G. J., Ng A., Liu B., Philip S. Y., others. (2008). Top 10 algorithms in data mining. *Knowledge and information systems*. Springer, Vol. 14, no. 1. 1–37. [in English].
4. Settouti N., Bechar M. E. A., Chikh M. A. (2016). Statistical comparisons of the top 10 algorithms in data mining for classification task. *International Journal of Interactive Multimedia and Artificial Intelligence*. Vol. 4, no. 1, 46–51. [in English].
5. Jordan M. I., Mitchell T. M. (2015). Machine learning: Trends, perspectives, and prospects. *Science*. *American Association for the Advancement of Science*, Vol. 349, No. 6245, 255–260. [in English].
6. Jackson P. (1986). Introduction to expert systems. Addison-Wesley Pub. Co., Reading, MA, [in English].
7. Kuwajima H., Yasuoka H., Nakae T. (2020). Engineering problems in machine learning systems. *Machine Learning*. – Springer Science; Business Media LLC, Vol. 109, no. 5, 1103–1126. [in English].
8. McCallum Q. E. (2012). Bad data handbook: Cleaning up the data so you can get back to work. – "O'Reilly Media, Inc.". [in English].
9. Hadi Amini M., Shafie-khah M. (2022). Cyberphysical smart cities infrastructures: Optimal operation and intelligent decision making / 1st ed. – Wiley. [in English].
10. Ahmad K., Maabreh M., Ghaly M., Khan K., Qadir J., Al-Fuqaha A. (2022). Developing future human-centered smart cities: Critical analysis of smart city security, data management, and ethical challenges // *Computer Science Review*. – Elsevier BV, Vol. 43, 100452. [in English].
11. Jazzyyear. (2020). The Chronicles of Cloud Building in Hangzhou: Part 2 / Alibaba Cloud Community. [in English].
12. Usman M., Jan M. A., He X., Chen J. (2019). A survey on big multimedia data processing and management in smart cities // *ACM Computing Surveys (CSUR)*. – ACM New York, NY, USA, Vol. 52, no. 3, 1–29. [in English].
13. Pérez A. T. E., Camargo M., Rincón P. C. N., Marchant M. A. (2017). Key challenges and requirements for sustainable and industrialized biorefinery supply chain design and management: A bibliographic analysis // *Renewable and Sustainable Energy Reviews*. – Elsevier BV, Vol. 69, 350–359. [in English].

14. Ignatiev A. (2020). Towards trustable explainable AI / IJCAI. [in English].
15. Szegedy C., Zaremba W., Sutskever I., Bruna J., Erhan D., Goodfellow I., Fergus R. (2013). Intriguing properties of neural networks // arXiv preprint arXiv:1312.6199. [in English].
16. Bommasani R., Hudson D. A., Adeli E., Altman R., Arora S., et.al.(2021). On the opportunities and risks of foundation models. [in English].
17. Gunning D., Aha D. (2019). DARPA's explainable artificial intelligence (XAI) program. *AI Magazine*. – Association for the Advancement of Artificial Intelligence (AAAI), Vol. 40, no. 2, 44–58. [in English].
18. Honda H., Hagiwara M. (2021). Analogical reasoning with deep learning-based symbolic processing // IEEE Access. – Institute of Electrical; Electronics Engineers (IEEE), Vol. 9, 121859–121870. [in English].
19. Flach P., Sokol K. (2018). Simply logical: Intelligent reasoning by example – online edition. – Zenodo. [in English].
20. Wielemaker J., Schrijvers T., Triska M., Lager T. (2012). SWI-Prolog. Theory and Practice of Logic Programming. Vol. 12, no. 1-2, 67–96. [in English].

УДК [004.75:004.455]:378.4(477.52)
DOI <https://doi.org/10.32689/maup.it.2022.1.3>

Андрій ЖМАКІН

начальник відділу системного забезпечення та адміністрування, Сумський державний університет, вул. Петропавлівська, 57, Суми, Україна, індекс 40000 (a.zhmakyn@cto.is.sumdu.edu.ua)

ORCID: 0000-0001-7418-9633

Віталій КОВАЛЬ

кандидат фізико-математичних наук, старший викладач кафедри кібербезпеки, Сумський державний університет, вул. Р-Корсакова, 2, індекс 40007 (v.koval@dcs.sumdu.edu.ua)

ORCID: 0000-0002-1593-5605

Володимир ЛЮБЧАК

кандидат фізико-математичних наук, доцент, завідувач кафедри кібербезпеки, Сумський державний університет, вул. Р-Корсакова, 2, індекс 40007 (v.liubchak@dcs.sumdu.edu.ua)

ORCID: 0000-0002-7335-6716

Святослав ШПІЦГЛУЗ

провідний фахівець відділу центру телекомунікаційних технологій та комп'ютерного забезпечення, Сумський державний університет, вул. Петропавлівська, 57, Суми, Україна, індекс 40000 (s.shpitsluz@cto.is.sumdu.edu.ua)

ORCID: 0000-0003-3683-3920

Andrew ZHMAKIN

Head of the System Support and Administration Department, Sumy State University, 57 Petropavlivska str., Sumy, Ukraine, postal code 40000 (a.zhmakyn@cto.is.sumdu.edu.ua)

Vitalii KOVAL

Ph.D. in Physics and Mathematics, Senior Lecturer at the Department of Cybersecurity, Sumy State University, 2 Rymskogo-Korsakova str., Sumy, Ukraine, postal code 40007 (v.koval@dcs.sumdu.edu.ua)

Volodymyr LIUBCHAK

Ph.D. in Physics and Mathematics, Associate Professor, Chair of the Department of Cybersecurity, Sumy State University, 2 Rymskogo-Korsakova str., Sumy, Ukraine, postal code 40007 (v.liubchak@dcs.sumdu.edu.ua)

Sviatoslav SHPITSHLUZ

Leading Specialist of the Center Telecommunication Technology and Computer Support, Sumy State University, 57 Petropavlivska str., Sumy, Ukraine, postal code 40000 (s.shpitsluz@cto.is.sumdu.edu.ua)

Бібліографічний опис статті: Жмакін А., Коваль В., Любчак В., Шпіцглуз С. Програмно-технічні рішення створення бюджетного варіанту комп'ютерних систем навчального та офісного призначення. *Інформаційні технології та суспільство*. 2022. Вип. 1 (3). С. 23–30. DOI: <https://doi.org/10.32689/maup.it.2022.1.3>

Bibliographic description of the article: Zhmakin, A., Koval, V., Liubchak, V., Shpitsluz, S. (2022). Prohramno-tekhnichni rishennia stvorennia biudzhetnoho variantu komp'iuternykh system navchalnoho ta ofisnoho pryznachennia [Software and technical solutions for creating a budget option of computer systems for educational and office purposes]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, 1 (3), 23–30. DOI: <https://doi.org/10.32689/maup.it.2022.1.3>

ПРОГРАМНО-ТЕХНІЧНІ РІШЕННЯ СТВОРЕННЯ БЮДЖЕТНОГО ВАРІАНТУ КОМП'ЮТЕРНИХ СИСТЕМ НАВЧАЛЬНОГО ТА ОФІСНОГО ПРИЗНАЧЕННЯ

Анотація. У статті розглянуто впровадження системи клієнт-серверної технології на основі VMware Horizon в діяльність закладу вищої освіти. Метою роботи є розробка технічного рішення створення систем навчального або

офісного призначення на основі використання застарілого обладнання, яке потрібно залучати до роботи та оптимізувати фінансові витрати на закупівлю нового обладнання. На основі інформаційного огляду можливих шляхів використання застарілої комп'ютерної техніки в діяльності освітнього закладу, для оптимального визначення методів і інструментів вирішення проблеми, методологією обрано хмарні та клієнт-серверні рішення. Наукова новизна полягає у використанні платформи VMware Horizon для розгортання віртуальних робочих столів та додатків для кінцевих користувачів на базі застарілої комп'ютерної техніки. Для взаємодії між користувачем та віртуальним робочим столом використовуються спеціальні протоколи віддаленого відображення. На основі інструментарію VMware Horizon розроблені та впроваджені системи в діяльність Сумського державного університету (СумДУ). Дослідження функціональних характеристик шляхом порівняння часу завантаження системи та стандартних додатків для користувача, який використовує технологію VMware Horizon, з сучасними ПК, на яких встановлено ОС Windows 10/11, підтверджує достатньо швидку та комфортну роботу із системою. Висновки: досвід створення та експлуатації системи віртуальних робочих столів на базі VMware Horizon підтверджує доцільність та оптимальність рішень щодо використання застарілої техніки в підрозділах та навчальних класах закладів освіти. Це дає можливість достатньо швидко та дешево розгорнути використання застарілих персональних комп'ютерів в роботі, зручно та безпечно надавати користувачам можливість працювати з сучасними програмними продуктами.

Ключові слова: хмарні технології, клієнт-серверна технологія, персональний комп'ютер (ПК), віртуальні робочі місця.

SOFTWARE AND TECHNICAL SOLUTIONS FOR CREATING A BUDGET OPTION OF COMPUTER SYSTEMS FOR EDUCATIONAL AND OFFICE PURPOSES

Abstract. The article considers the introduction of the client-server technology system based on VMware Horizon in the activities of higher education institutions. The aim of the work is to develop a technical solution for the creation of educational or office systems based on the use of obsolete equipment that needs to be involved in work and optimize the financial costs for the purchase of new equipment. Based on the information review of possible ways of using outdated computer equipment in the activities of the educational institution, for the optimal definition of methods and tools for solving the problem, the methodology selected cloud and client-server solutions. The scientific novelty is the use of the VMware Horizon platform to deploy virtual desktops and end-user applications based on outdated computer technology. Special remote display protocols are used to interact between the user and the virtual desktop. Based on the VMware Horizon toolkit, the system was developed and implemented in the activities of Sumy State University (SSU). A study of functional characteristics by comparing the boot time of the system and standard applications for the user using VMware Horizon technology with modern PCs running Windows 10/11, confirms a fairly fast and comfortable operation of the system. Conclusions: The experience of creating and operating a virtual desktop system based on VMware Horizon confirms the feasibility and optimal solutions for the use of obsolete equipment in departments and classrooms of educational institutions. This allows you to quickly and cheaply deploy the use of obsolete personal computers in the work, conveniently and safely provide users with the opportunity to work with modern software products.

Key words: cloud technology, client-server technology, personal computer (PC), virtual workstations.

Постановка проблеми. Для більшості бюджетних організацій актуальним є питання раціонального використання комп'ютерної техніки, яка морально або фізично застаріла. В таких організаціях, навчальних закладах є в наявності комп'ютерна техніка, яка потребує списання або модернізації, бо середній термін використання будь-якої комп'ютерної техніки в Україні становить приблизно 10 років. Постає питання: як вирішити проблему з застарілими ПК по впровадженню в робочий та навчальний процес, дати можливість жити морально застарілому «залізу», як оптимізувати витрати на придбання нових персональних комп'ютерів (ПК) та модернізацію існуючого обладнання? Для існуючого парку ПК на сьогодні маємо дві можливості: модернізувати техніку шляхом оновлення складових ПК або виконати оптимізацію інформаційних систем під існуючі ПК.

У даній статті акцент робиться на питаннях застосування ПК для навчального та офісного призначення. Саме для цих сфер:

по-перше, сформувався доволі класичний та стабільний за часом набір програмного забезпечення, з невеликою потребою актуального оновлення;

по-друге, за більшості відсутнє високошвидкісне потужне комп'ютерне обладнання або його активне застосування;

по-третє, вимоги до інформаційної безпеки не є надскладними;

по-четверте, є можливість спільної роботи в мережі, що дозволяє швидко створювати, адаптувати та поширювати навчальні ресурси;

по-п'яте, безкоштовність, надійність та безпека, простота та зручність користування. Педагогам, працівникам та студентам, щоб навчитися працювати з даними сервісами, достатньо кількох занять для пояснення у роботі з даним сервісом.

Аналіз останніх досліджень. Огляд існуючих рішень для систем навчального призначення пріоритетне застосування хмарних технологій [1; 2; 3; 11]. Хмарні рішення дозволяють використовувати сучасну модель ІТ-інфраструктури, яка адаптуватиметься до потреб університетів та шкіл. Завдяки їх

застосуванню доступ до інформації та програмних продуктів буде необмеженим за місцем та часом, і педагоги та працівники освіти зможуть зреалізувати нові форми проведення занять зі студентами та учнями. Необхідно відзначити, що інформаційній безпековій складовій приділяється в цій сфері досить багато уваги.

Однак, хоча основний акцент на збільшення розрахункових потужностей переноситься на сервери хмарних технологій і темпи «росту» параметрів клієнтських ПК вже не такі стрімкі, темпи «осучаснення» вітчизняних ПК все рівно не встигають за потребами. При чому збільшується необхідність застосування додаткових програмних продуктів та спеціальних засобів «комунікації».

Другою технологією побудови комп'ютерної системи навчального та офісного призначення є побудова на принципі застосуванні архітектури клієнт-сервер. Даний підхід є одним із архітектурних шаблонів програмного забезпечення та є домінуючою концепцією у створенні розподілених мережних застосунків і передбачає взаємодію та обмін даними між ними. Вона передбачає такі компоненти: набір серверів, які надають інформацію або інші послуги програмам, які звертаються до них; набір клієнтів, які використовують сервіси, що надаються серверами; мережа, яка забезпечує взаємодію між клієнтами та серверами.

Сервери є незалежними один від одного. Клієнти також функціонують паралельно і незалежно один від одного. Більш ніж типовою є ситуація, коли один сервер одночасно обробляє запити від різних клієнтів; з іншого боку, клієнт може звертатися то до одного сервера, то до іншого. Клієнти мають знати про доступні сервери, але можуть не мати жодного уявлення про існування інших клієнтів. Існує велика кількість модифікацій застосування архітектури клієнт-сервер у освітньо-офісному просторі, які по різному ставлять акценти на застосування серверної складової або клієнтської складової.

У більшості підходів застосування архітектури клієнт-сервер зводиться до інтеграції складових систем між собою [1; 5; 7] та застосування потужного технічного обладнання. У рамках існуючого парку ПК найбільш привабливим і ефективним є застосування двохрівневого типу архітектури, а саме модель тонкого клієнта, в рамках якої основна логіка застосунку, розрахункові потужності та управління даними зосереджена на сервері. Клієнтська програма забезпечує тільки рівень представлення, що потребує набагато менших потужностей і для цього достатньо потужностей існуючого парку ПК.

Одним з варіантів, який дозволить «модернізувати застарілу інфраструктуру» та послабити розрахункові навантаження на клієнтську складову інформаційної системи є застосування VMware Horizon. Цей інструмент дозволяє створити цифрову робочу область з ефективним наданням віртуальних робочих місць та програм [14]. Ці рішення активно використовуються для створення освітніх та навчальних систем у різних країнах світу [10; 13].

Постановка завдання. Основні вимоги, яким повинні задовільнити інформаційні системи навчального призначення, наступні: стандартний пакет програмного забезпечення; інформаційна безпека; необтяжливості застосування в порівнянні з традиційними.

Найчастіше пакет програмного забезпечення, який використовується, є досить визначеним і з часом практично не змінюється. До цього пакету можна віднести пакет офісних програм (Microsoft Office, LibreOffice). Питання, безпекової складової для навчального та офісного призначення теж важливе, застосування застарілого програмного забезпечення може призвести до втрати не лише частини інформаційної системи, а і до зупинки функціонування всієї інформаційної системи. Тому побудова програмно-технічного рішення вимагає в першу чергу створення практично «системи в системі» з підвищеною інформаційною безпековою складовою.

Застосування новітнього підходу повинно носити практично «непомітний» тип інтеграції, оскільки «перенавчання» може потребувати більше часу, ніж оновлення ПК. Рішення, що пропонуються, повинні досить легко інтегруватись у існуючі системи та сприйматись користувачами. Як приклад, це може бути додаткове вікно з мінімальним набором функціоналу або веб-сторінка. При створенні з ПК тонкого клієнту з термінальним режимом роботи необхідно автоматизувати цей процес до мінімального втручання користувача та досить інтуїтивного інформування про можливі типові помилки.

Підкреслимо ще один важливий момент: основна кількість ПК в Україні працює на ОС сімейства Windows. З виходом нової операційної системи Windows 11 питання модернізації та оптимізації роботи систем для застарілих ПК стає ще більш актуальним.

Основні системні вимоги Windows 11 подібні до Windows 10. Однак Windows 11 підтримує лише 64-розрядні системи, що використовують процесор x86-64 або ARM64; підтримку процесорів IA-32 видалено. Для Windows 11 спадковий BIOS більше не підтримується, потрібна система UEFI із безпечним

завантаженням та співпроцесором безпеки TPM 2.0 [6, 9, 12, 15]. Також було збільшено мінімальні вимоги до внутрішньої та оперативної пам'яті: тепер для Windows 11 потрібно щонайменше 4 ГБ оперативної пам'яті та 64 ГБ внутрішньої пам'яті. Режим S (обмежена підтримка драйверів і застосунків задля швидкості та безпеки) підтримується лише для Домашньої версії Windows 11 [8].

Згідно з дослідженнями бельгійської компанії Lansweeper, понад 55 відсотків комп'ютерів не відповідають вимогам Windows 11, а найбільша несумісність спостерігається серед процесорів. Статистику було зібрано про 30 мільйонів комп'ютерів з 60 тисяч організацій по всьому світу.

Таким чином, задача полягає у вирішенні питання по впровадженню застарілих ПК щодо сучасних програмних засобів, які потребують більших вимог до ПК. Застаріле обладнання з мінімальними параметрами буде майже неможливо використовувати з існуючою конфігурацією, а для цього потрібно підібрати, встановити та оптимізувати клієнт-серверну систему на основі тонкого клієнта, яка зможе легко задовольнити користувачів у використанні застарілих ПК з мінімально допустимою конфігурацією.

Метод дослідження. Пропонується для створення та впровадження системи клієнт-серверної технології використовувати тонкий клієнт, який буде реалізовано на основі технології інструменту VMware Horizon. Дана технологія буде використовуватися для залучення застарілих ПК, на яких сучасні операційні системи не можуть нормально функціонувати, тому що конфігурація ПК не дає можливості швидко і зручно працювати з ними.

VMware Horizon – це платформа для швидкої та безпечної доставки віртуальних робочих столів та додатків у гібридних хмарах для кінцевих користувачів [4]. Платформа Horizon забезпечує сучасний підхід до управління настільними комп'ютерами та додатками, який простягається від локальних до гібридних та мультимарних середовищ, використовуючи функції управління та глибоку інтеграцію з технологічною екосистемою VMware.

VMware Horizon, відомий як VMware Horizon View, це продукт, який працює з рештою портфеля VMware. Horizon, а не тільки підтримує термінальні сервери (VNC або Windows Terminal Services, служби віддалених робочих столів). VMware Horizon заснований на VMware vSphere та розміщує віртуальні робочі столи в середовищі vSphere. Користувальницькі робочі столи – це віртуальні машини (ВМ), які працюють на хостах ESXi [4].

Таким чином, доступні моментальні знімки, vMotion, висока доступність, розподілений планувальник ресурсів та інші можливості vSphere. Оскільки дані користувача зберігаються на віртуалізованих серверах, якими можна централізовано керувати, оновлювати і виконувати резервне копіювання, рішення VMware Horizon VDI забезпечує більшу гнучкість і безпеку, ніж термінальні служби. VMware Horizon можна використовувати для будь-якого зі сценаріїв, оскільки це рішення VDI. Особливо корисним може бути при використанні платформи віртуалізації VMware vSphere.

До віртуальних робочих столів можна отримати доступ із персональних комп'ютерів (ПК), планшетних ПК, мобільних телефонів, тонких клієнтів або нульових клієнтів.

Основні компоненти VMware Horizon:

- vCenter Server: для VMware vSphere vCenter Server це централізована система керування;
- ESXi гіпервізор: фізичний сервер, на якому розміщено віртуальні машини, відомий як гіпервізор ESXi;
- Перегляд агента: агент VMware Horizon View – програмний компонент, який необхідно встановити на всіх віртуальних машинах, керованих VMware Horizon View [4];
- Клієнт Horizon: клієнтська програма, яка взаємодіє з View Connection Server для встановлення з'єднання між пристроями користувача кінцевих точок і віртуальними робочими столами Horizon або додатками. Клієнт сумісний із Windows, Linux та MacOS операційних систем;
- Сервер підключень Horizon View – аутентифікує користувачів через Active Directory, пропонує єдиний вхід та, крім інших функцій, пов'язує віртуальні робочі столи з користувачами.
- ThinApp: VMware використовує ThinApp як додатковий компонент для віртуалізації програм. Оскільки ця технологія віртуалізації програм не вимагає використання агентів, для її використання на робочій станції користувача не потрібне встановлення будь-якого програмного забезпечення [4].

Важливим питанням є використання клієнта для підключення до віртуального робочого столу та додатків. Клієнт VMware Horizon є програмою, яка дозволяє користувачам підключатися до віртуального робочого столу VMware Horizon з різних пристроїв та доступний для Windows, Linux, Mac OS X, iOS, Android.

Програма взаємодіє із сервером підключень View для автентифікації користувачів після того, як вони вводять свої облікові дані в Horizon Client. Після автентифікації сервера з'єднаний сервер знаходить відповідні віртуальні робочі столи для користувачів і надає їм доступ з певними дозволами.

Для взаємодії між пристроєм кінцевого користувача та віртуальним робочим столом використовуються спеціальні протоколи віддаленого відображення. Horizon Client підтримує протоколи VMware Blast, Microsoft RDP (протокол віддаленого робочого столу) та PCoIP (ПК через IP) для підключення до віддалених робочих столів [4].

VMware Horizon підтримує лише Blast, власний протокол на основі TCP, створений VMware. Клієнти, яким не вдається встановити Horizon Client на свій пристрій, можуть підключатися за допомогою браузерів, які підтримують HTML-5. Якщо пристрій розроблений спеціально для PCoIP і не підтримує VMware Blast, користувачі пристроїв з нульовим клієнтом повинні використовувати його [4].

Основні результати. Для вирішення завдання розглянемо процес та результати створення системи навчального призначення на прикладі Сумського державного університету (СумДУ). На даний момент дана технологія реалізована в багатьох підрозділах університету, а саме в комп'ютерних класах, університетській бібліотеці та кафедральних та структурних підрозділах, які мають ПК з застарілою конфігурацією.

В університеті достатньо застарілою комп'ютерної техніки, яка знаходиться в робочому стані, але не відповідає сучасним вимогам по параметрам конфігурації ПК. Технологія VMware Horizon дала можливість впровадити клієнт-серверну технологію для застарілих ПК та використовувати в робочому і навчальному процесі.

Застосовано технологію VMware Horizon для роботи на сервері з такими параметрами: Intel(R) Xeon(R) CPU E 5440 @ 2.83GHz 2.83 GHz (2 процесоре) на основі двоядерного процесору з тактовою частотою 2,83 Ghz кожний з 32-розрядною операційною системою, що виконує функцію станції, на якій встановлено операційні системи та додатки для використання в навчальному та робочому процесі (рис. 1):

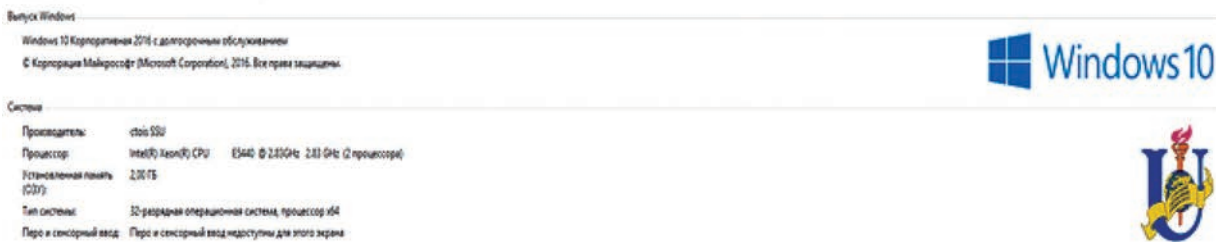


Рис. 1. Параметри серверу для підтримки технології VMware Horizon

Перелічимо додатки та системи, що використовуються в даній технології в СумДУ (рис. 2):

ID	Display Name	Pool or Farm	Version	Publisher	User Count	App Shortcuts	Pre-Launch	Multi-Session Mode	Status
Acrobat_Reader_DC	Acrobat Reader DC	01	19.8.20071.303822	Adobe Systems Incorporated				Disabled	Available
BAS	BAS_ERP	01	8.3.18.1289	BAF			✓	Disabled	Available
Calculator	Калькулятор	01	10.0.14393.0	Microsoft Corporation				Disabled	Available
Excel_2016	Excel 2016	01	16.0.4266.1001	Microsoft Corporation				Disabled	Available
Firefox	Firefox	01	65.0.1	Mozilla Corporation				Disabled	Available
Google_Chrome	Google Chrome	01	59.0.3071.115	Google Inc.				Disabled	Available
Notepad	Блокнот	01	10.0.14393.0	Microsoft Corporation				Disabled	Available
Paint	Paint	01	10.0.14393.0	Microsoft Corporation				Disabled	Available
PowerPoint_2016	PowerPoint 2016	01	16.0.4266.1001	Microsoft Corporation				Disabled	Available
StarSuite_12_64-bit	StarSuite 12 (64-bit)	01	421.12.0.866	StarCorp LP				Disabled	Available
Total_Commander_64-bit	Total Commander 64-bit	01	9.21	Ghesler Software GmbH				Disabled	Available
Word_2016	Word 2016	01	16.0.4732.1000	Microsoft Corporation				Disabled	Available
Wordpad	WordPad	01	10.0.14393.0	Microsoft Corporation				Disabled	Available

Рис. 2. Додатки, які використовує університет на основі технології VMware Horizon

Після авторизації клієнта в домені СумДУ під своїм обліковим записом, користувач має такий вид вікна для використання технології VMware Horizon (рис. 3), який завантажується системою автоматично:

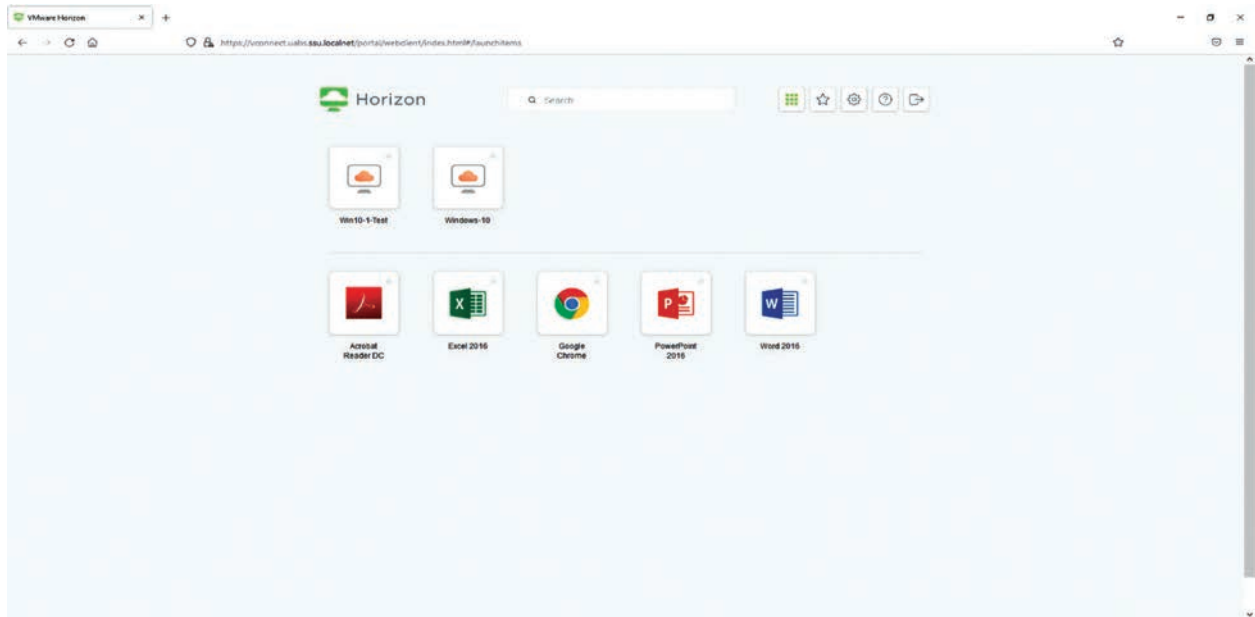


Рис. 3. Вид авторизації користувача на основі технології VMware Horizon

Далі у користувача є можливість завантажити операційну систему Windows 10 через автоматичну авторизацію в домені, яка була здійснена на основі технології VMware Horizon та має наступний вид (рис. 4):

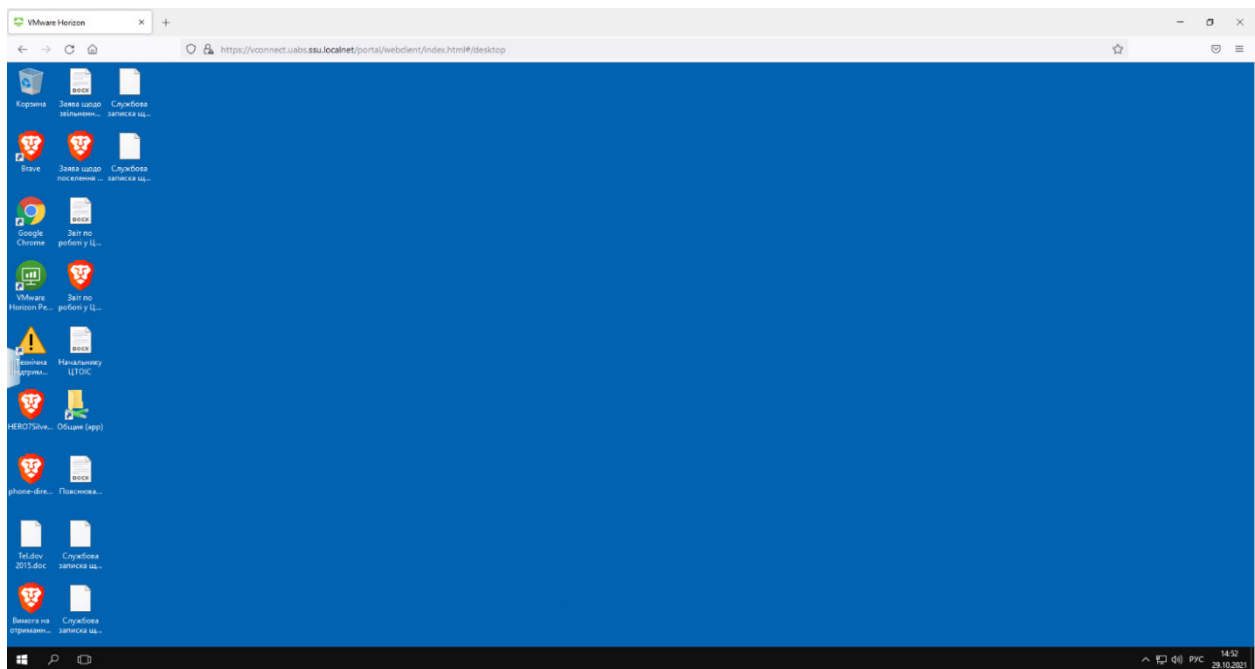


Рис. 4. Авторизація користувача в операційній системі Windows 10 завдяки технології VMware Horizon

Дана операційна система запускається та функціонує через браузер Google Chrome, який встановлений на сервері VMware Horizon. Далі користувач, запустивши операційну систему через браузер має можливість відкрити будь-який додаток та використувувати його в робочому процесі завдяки клієнт-серверній технології, яка працює на сервері VMware Horizon.

Було проведено дослідження функціональних характеристик. А саме, визначено час завантаження системи та стандартних додатків для користувача, який використовує технологію VMware Horizon на сервері з параметрами, зазначеними вище. Протестувавши запуск системи та додатків на основі даної технології визначено наступне: завантаження ПК – 20 секунд, авторизація в домені – 17 секунд, запуск Word та Excel по 2 секунди кожний з додатків, PowerPoint – 4 секунди та Windows 10 через браузер додатку Google Chrome – 4 секунди.

Проаналізувавши, запуск системи і додатків застарілих ПК на основі клієнт-серверної технології VMware Horizon, слід також порівняти запуск з сучасними ПК, на яких встановлено ОС Windows 10/11, але на сучасних ПК встановлено SSD-диски, більший об'єм оперативної пам'яті, а також сучасні та швидкісні процесори останніх поколінь. Це є ключовим фактором, який пришвидшує запуск цих ПК і програмних додатків на них, але не дуже значно, а лише в 2–3 рази, що становить запуск сучасного ПК за 5–7 секунд, завдяки SSD-диску та сучасному процесору.

По зазначеним даним можна стверджувати, що технологія VMware Horizon працює достатньо швидко на ПК, які вже морально застаріли завдяки даній технології та надає можливість даних ПК швидко зв'язуватися з сервером та авторизуватися користувачу для його роботи.

Як вище зазначалося, що технологія VMware Horizon була реалізована у філії університетської бібліотеки, яка знаходиться в Конгрес-центрі у найбільшій кількості з 20 ПК, На кількох кафедрах дану систему було реалізовано білш ніж на 15 ПК, також в комп'ютерних класах навчально-наукового інституту бізнесу, економіки та менеджменту, які спеціалізовано під конкретні задачі для навчання студентів.

Висновки. Отже, проаналізувавши технологію VMware Horizon можна зробити деякі висновки. VMware Horizon View – це рішення для інфраструктури віртуальних робочих столів (VDI), що поєднує переваги інфраструктури віртуальних робочих столів з платформою віртуалізації VMware vSphere. VDI (інфраструктура віртуального робочого столу) – це технологія, яка дозволяє користувачам підключатися до віртуальних робочих столів, а не до реальних ПК, з різних місць та з кількох пристроїв.

VMware Horizon краще підходить там, де вже використовуються технології віртуалізації VMware. Ключовими компонентами Horizon View є vCenter Server, ESXi Hypervisor, View Agent, Horizon Client, Connection Server, ThinApp, View Composer та Horizon Administrator.

Користувачі отримують найкращий досвід і можуть отримати доступ до своїх робочих столів із будь-якого місця за допомогою сучасних технологій віддалених робочих столів. Віртуальні робочі столи не залежать від обладнання, що дозволяє системним адміністраторам уникнути проблем, таких як пошук драйверів для різних пристроїв або ремонт пошкодженого обладнання на ПК. Якщо ви шукаєте альтернативу традиційним робочим столам або рішенням віддалених робочих столів, варто звернути увагу на VMware Horizon View.

Використання хмарних технологій у вищій школі має чималі перспективи розвитку. Студенти та співробітники мають змогу користуватися сучасним програмним забезпеченням при вивченні різних дисциплін. Тому для цього була реалізована технологія VMware Horizon, яка дає змогу легко та зручно використовувати застарілі ПК в роботі, надавати безпеку у використанні даних ПК, а також максимально забезпечити надання функцій та програм, які використовуються в робочому та навчальному процесі.

Список використаних джерел:

1. Використання інформаційно-комунікативних технологій в освітньому процесі. URL: <http://www.journal.org.ua/index.php/appos/article/view/133> (дата звернення: 18.12.2021).
2. Підходи до використання хмарних технологій у навчальному процесі вищої школи у вітчизняній науковій літературі. URL: <https://library.vspu.net/handle/123456789/2544> (дата звернення: 24.12.2021).
3. Сучасні інформаційно-комунікаційні технології в освіті. URL: <https://library.vspu.net/handle/123456789/5447> (дата звернення: 26.12.2021).
4. Як використовувати VMware Horizon у 2021 році: повний посібник з програмного забезпечення VDI. URL: <https://www.bloggersideas.com/ru/how-to-use-vmware-horizon/> (дата звернення: 10.12.2021).
5. Architecture for Integrating Learning Platforms Using Adapter. URL: <https://arxiv.org/abs/1903.06930>.
6. Compatibility for Windows 11 – Compatibility Cookbook. URL: <https://docs.microsoft.com/en-us/windows/compatibility/windows-11>.
7. Designing client-server and service oriented architecture of the distance learning system. URL: <https://ieeexplore.ieee.org/abstract/document/9351455>.
8. greg-lindsay. Windows 11 requirements – What's new in Windows. – June 24, 2021. URL: [https://docs.microsoft.com\(en-us\)](https://docs.microsoft.com(en-us)).
9. Paul Thurrott. Microsoft Unveils Windows 11. – June 24, 2021. URL: <https://www.thurrott.com/windows/windows-11/252299/microsoft-unveils-windows-11>.
10. Transform K-12 Education with Follow Me Learning. URL: <https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/solutions/vmw-solution-brief-k12-edu.pdf>.

11. The use of the cloud technologies to support the educational research in the open science area. URL: <https://www.ccjournals.eu/ojs/index.php/nocote/article/view/824>.
12. Hanson, Matt (June 24, 2021). Windows 11 system requirements are bad news for old laptops and PCs. TechRadar. – June 24, 2021. URL: <https://www.techradar.com/news/windows-11-system-requirements-are-bad-news-for-old-laptops-and-pcs>.
13. U.S. School Districts Transform Education for the Digital Age with VMware View. URL: <https://news.vmware.com/releases/view-edu-momentum>.
14. VMware, Inc. VMware Horizon / VMware, Inc. – 2021. URL: <https://www.vmware.com/products/horizon.html>.
15. Windows 11 Specifications – Microsoft. Windows (en-us). – June 24, 2021. URL: <https://www.microsoft.com/en-us/windows/windows-11-specifications>.

References:

1. Vykorystannia informatsiino-komunikatyvnykh tekhnolohii v osvithomu protsesi [The use of information and communication technologies in the educational process]. [www.journal.org.ua](http://www.journal.org.ua/index.php/appos/article/view/133). Retrieved from <http://www.journal.org.ua/index.php/appos/article/view/133> [in Ukrainian].
2. Pidkhody do vykorystannia khmarnykh tekhnolohii u navchalnomu protsesi vyshchoi shkoly u vitchyzniani naukovii literaturi [Approaches to the use of cloud technologies in the educational process of higher education in the domestic scientific literature]. library.vspu.net. Retrieved from <https://library.vspu.net/handle/123456789/2544> [in Ukrainian].
3. Suchasni informatsiino-komunikatsiini tekhnolohii v osviti [Modern information and communication technologies in education]. library.vspu.net. Retrieved from <https://library.vspu.net/handle/123456789/5447> [in Ukrainian].
4. Iak vykorystovuvaty VMware Horizon u 2021 rotsi: povnyi posibnyk z prohramnoho zabezpechennia VDI [How to use VMware Horizon in 2021: a complete guide to VDI software]. www.bloggersideas.com. Retrieved from <https://www.bloggersideas.com/ru/how-to-use-vmware-horizon/> [in Ukrainian].
5. Architecture for Integrating Learning Platforms Using Adapter. URL: <https://arxiv.org/abs/1903.06930> [in English].
6. Compatibility for Windows 11 – Compatibility Cookbook. URL: <https://docs.microsoft.com/en-us/windows/compatibility/windows-11> [in English].
7. Designing client-server and service oriented architecture of the distance learning system. URL: <https://ieeexplore.ieee.org/abstract/document/9351455> [in English].
8. greg-lindsay. Windows 11 requirements – What’s new in Windows. – June 24, 2021. URL: [docs.microsoft.com\(en-us\)](https://docs.microsoft.com(en-us)) [in English].
9. Paul Thurrott. Microsoft Unveils Windows 11. – June 24, 2021. URL: <https://www.thurrott.com/windows/windows-11/252299/microsoft-unveils-windows-11> [in English].
10. Transform K-12 Education with Follow Me Learning. URL: <https://www.vmware.com/content/dam/digitalmarketing/vmware/en/pdf/solutions/vmw-solution-brief-k12-edu.pdf> [in English].
11. The use of the cloud technologies to support the educational research in the open science area. URL: <https://www.ccjournals.eu/ojs/index.php/nocote/article/view/824> [in English].
12. Hanson, Matt (June 24, 2021). Windows 11 system requirements are bad news for old laptops and PCs. TechRadar. – June 24, 2021. URL: <https://www.techradar.com/news/windows-11-system-requirements-are-bad-news-for-old-laptops-and-pcs> [in English].
13. U.S. School Districts Transform Education for the Digital Age with VMware View. URL: <https://news.vmware.com/releases/view-edu-momentum> [in English].
14. VMware, Inc. VMware Horizon / VMware, Inc. – 2021. URL: <https://www.vmware.com/products/horizon.html> [in English].
15. Windows 11 Specifications – Microsoft. Windows (en-us). – June 24, 2021. URL: <https://www.microsoft.com/en-us/windows/windows-11-specifications> [in English].

УДК 004.051

DOI <https://doi.org/10.32689/maup.it.2022.1.4>

Галина КИРИЧЕК

кандидат технічних наук, доцент кафедри комп'ютерних систем та мереж, Національний університет «Запорізька політехніка», вул. Жуковського, 64, Запоріжжя, Україна, індекс 69063 (kirgal08@gmail.com)

ORCID: 0000-0001-5725-5942

Марія ТЯГУНОВА

кандидат технічних наук, доцент кафедри комп'ютерних систем та мереж, Національний університет «Запорізька політехніка», вул. Жуковського, 64, Запоріжжя, Україна, індекс 69063 (mary.tyagunova@gmail.com)

ORCID: 0000-0002-9166-5897

Анна КУРАЧ

студент, Національний університет «Запорізька політехніка», вул. Жуковського, 64, Запоріжжя, Україна, індекс 69063 (annchubich@gmail.com)

ORCID: 0000-0002-0829-1864

Halyna KYRYCHEK

PhD in Technical Sciences, Associate Professor at Computer Systems and Networks Department, National University "Zaporizhzhya Polytechnic", 64 Zhukovsky str., Zaporizhzhia, Ukraine, postal code 69063 (kirgal08@gmail.com)

Mariia TIAHUNOVA

PhD in Technical Sciences, Associate Professor at Computer Systems and Networks Department, National University "Zaporizhzhya Polytechnic", 64 Zhukovsky str., Zaporizhzhia, Ukraine, postal code 69063 (mary.tyagunova@gmail.com)

Anna KURACH

Student, National University "Zaporizhzhya Polytechnic", 64 Zhukovsky str., Zaporizhzhia, Ukraine, postal code 69063 (annchubich@gmail.com)

Бібліографічний опис статті: Киричек Г., Тягунова М., Курач А. Автоматизоване тестування веб-платформ з використанням Java та Selenium. *Інформаційні технології та суспільство*. 2022. Вип. 1 (3). С. 31–37. DOI: <https://doi.org/10.32689/maup.it.2022.1.4>

Bibliographic description of the article: Kyrychek, H., Tiahunova, M., Kurach, A. (2022). Avtomatyzovane testuvannia veb-platform z vykorystanniam Java ta Selenium [Automated testing of web platforms using Java and Selenium]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, 1 (3), 31–37. DOI: <https://doi.org/10.32689/maup.it.2022.1.4>

АВТОМАТИЗОВАНЕ ТЕСТУВАННЯ ВЕБ-ПЛАТФОРМ З ВИКОРИСТАННЯМ JAVA ТА SELENIUM

Анотація. На даний час тестування програмного забезпечення є одним із основних етапів забезпечення контролю його якості та ефективності використання. Перехід на автоматизацію дозволяє скоротити час тестування і значно прискорити цей процес. Система, яка пропонується для реалізації, дозволить швидше та якісніше виконувати автоматизовані тести. **Метою роботи** є реалізація системи автоматизованого тестування веб-платформ із використанням мови програмування Java та інструменту Selenium, з підтримкою усіх популярних браузерів і операційних систем. Для досягнення основної мети пропонується вирішити наступні **завдання**: дослідити аналоги систем автоматизованого тестування; реалізувати метод автоматизованого тестування веб-платформ із використанням мови програмування Java та інструменту Selenium; розробити алгоритм запуску тестів в рамках системи для різних браузерів та операційних систем; навести метод автоматизованого тестування із використанням Maven та Selenium Web Driver та результати тестування системи в порівнянні з аналогами. **Наукова новизна.** Авторами пропонується застосовувати загальний фреймворк автоматизованого тестування, як систему, набір умов, концепцій та практик, спрямованих на перевикористання, зменшення витрат на підтримку, підвищення надійності, швидкості та якості виконання тестів, включаючи його використання широким колом фахівців, включаючи розробників та спеціалістів з ручного тестування. **Висновком**, у роботі є те, що структура тестів реалізована за допомогою анотацій, що є зрозумілою для користувача та інформує систему про призначення поміченого коду. Окрім цього, при запуску та виконанні автоматизованих тестів за допомогою TestNG, використовується багатопотоковість, яка дозволяє одночасно виконувати декілька тестів. Перевагою системи є: підтримка різних браузерів та операційних систем; кращі швидкісні характеристики; детальна система звітів за результатами тестувань; умовна безкоштовність та реалізація проекту з відкритим вихідним кодом.

Ключові слова: тестування, автоматизація, веб-додатки, артефакт, фреймворк, Page Object pattern.

AUTOMATED TESTING OF WEB PLATFORMS USING JAVA AND SELENIUM

Abstract. Currently, software testing is one of the main stages of ensuring control over its quality and efficiency. Switching to automation reduces test time and significantly speeds up the process. The system proposed for implementation will allow you to perform automated tests faster and better. The **aim** of the work is to implement an automated testing system of web platforms using the Java programming language and the Selenium tool, with support for all popular browsers and operating systems. To achieve the main aim, it is proposed to solve the following **tasks**: to research the analogs of automated testing systems; to implement a method of automated testing of web platforms using the Java programming language and the Selenium tool; to develop an algorithm for running tests within the system for different browsers and operating systems; to give the method of automated testing using Maven and Selenium Web Driver and the results of testing the system in comparison with analogs. **Scientific novelty.** The authors proposed to use the general framework of automated testing as a system, set of conditions, concepts, and practices aimed at reuse, reduce maintenance costs, improve reliability, speed, and quality of test performance, including its use by a wide range of professionals, including developers and specialists in manual testing. The **conclusion** of the paper is that the structure of the tests is implemented using annotations, which are understandable to the user and inform the system about the purpose of the observed code. In addition, when running and running automated tests with TestNG, multithreading is used, which allows you to run multiple tests at the same time. The advantages of the system are: support for different browsers and operating systems; best speed characteristics; detailed system of reports on test results; conditional free and open source project implementation.

Key words: testing, automation, web applications, artifact, framework, Page Object pattern.

Актуальність проблеми. У сучасному світі розробка програмного забезпечення (ПЗ) постійно еволюціонує, тому тестування є одним з основних етапів контролю якості програмного забезпечення при його проектуванні, під час реалізації та при подальшому супроводженні [1]. При цьому автоматизоване тестування використовує програмні засоби для запуску тестів в процесі перевірки ПЗ, із попередньо вказаними очікуваними результатами, що допомагає скоротити час при проведенні тестувань. Актуальність вирішення цих проблем в зростанні кількості використань гнучких методів реалізації програмного забезпечення, основними напрямками яких є оптимізація виробничого процесу та мінімізація ризиків, шляхом зведення етапів розробки до серії коротких циклів. Тестування є процесом, спрямованим на виявлення характеристик інформаційних систем та демонстрації відмінностей між їх потрібним та фактичним станами [1; 2]. Окрім того воно є частиною процесу валідації при перевірці достовірності та верифікації, а точніше доказу того, що вірогідний факт або твердження є істинним [3]. При цьому автоматизація тестування ПЗ є методом, який виконується з використанням спеціальних програмних засобів, які необхідні для виконання набору тестових прикладів [4]. Тому, після автоматизації набору тестів, втручання тестувальника в їх виконання більше не потрібне [5]. Метою автоматизації є зменшення кількості тестових сценаріїв, які запускаються вручну, не виключи ручне тестування. При цьому, поєднання двох підходів часто є найбільш виграним варіантом, у якому частка тестів автоматизованих та виконаних вручну залежить від вимог проекту, його бюджету, експертизи команди та термінів проведення тестувань [3; 5].

Аналіз останніх досліджень та публікацій. При виконанні досліджень розглянуто п'ять фреймворків, які використовуються для автоматизованого тестування, це: Katalon, JBehave, Test Complete, Ranorex та Robot Framework. Авторами проаналізовані їх переваги та недоліки.

Katalon є дуже популярним інструментом при автоматизації процесу тестування у початківців Quality Assurance (QA) інженерів і має функції спрощеного запису скриптів та тестування із використанням ключових слів. Але використання мови Groovy, яка побудована на Java та завантаження безлічі бібліотек для аналізу тестових даних і об'єктів та записів журналу робить процес більш повільнішим, ніж у Java. Також в Katalon відсутні деталізовані звіти та немає можливості одночасного запуску декількох тестів [6]. JBehave є одним із найкращих фреймворків тестування Java, він підтримує розробку на основі поведінки, приділяючи особливу увагу взаємодії з користувачем та автоматизації дій. Але у JBehave високий поріг входження для новачків та він має складний етап установки тестового фреймворку [7]. TestComplete є ефективним інструментом для тестування десктопних, мобільних і веб-додатків та допомагає реалізувати свої тест-кейси у різних скриптових мовах: JavaScript, Python, VBScript, Delphi Script та JavaScript. Але він доступний тільки для операційної системи (ОС) Windows; має високу вартість та в ньому відсутній спрощений запис скриптів [8]. Ranorex Studio є приватним корпоративним інструментом для автоматизації тестування графічного інтерфейсу Windows, веб-додатків і мобільних додатків. При редагуванні записів або створенні тестів він використовує платформу Microsoft .NET і мови програмування C# і VB.NET, є платним та недостатньо поширеним [9]. Robot Framework (RF) є відкритим фреймворком автоматизації тестування для приймального тестування Acceptance Test Driven Development (ATDD) і Robotic Process Automation (RPA) та його ядро може запускатися на Jython (Java-реалізація Python) та IronPython (Python для .NET framework) [10; 11].

Виходячи з характеристик досліджених фреймворків, авторами пропонується застосовувати загальний фреймворк автоматизованого тестування, як систему, набір умов, концепцій та практик, спрямованих на перевикористання, зменшення витрат на підтримку, підвищення надійності, швидкості та якості виконання тестів, включаючи його використання широким колом фахівців, включаючи розробників та спеціалістів з ручного тестування.

Постановка завдання. Метою роботи є реалізація системи автоматизованого тестування веб-платформ із використанням мови програмування Java [2; 11], інструменту Selenium [1] та з підтримкою всіх популярних браузерів та операційних систем. Об'єктом дослідження є процес автоматизації тестування web-сервісів з використанням мови програмування Java та інструменту Selenium. Предметом є моделі, методи та програмні засоби автоматизації тестування web-сервісів. Процедурі автоматизованого тестування виконано із використанням Maven та Selenium Web Driver [1]. Структура проекту застосовує паттерн Page Object та Maven у середовищі IntelliJ IDEA.

Для досягнення основної мети авторами пропонується реалізувати: метод автоматизованого тестування веб-платформ із використанням мови програмування Java та інструменту Selenium; алгоритм запуску тестів в рамках системи, що дозволить проводити тестування із підтримкою різних браузерів та операційних систем; метод автоматизованого тестування із використанням Maven та Selenium Web Driver та модель системи з застосуванням паттерну Page Object та Maven у середовищі IntelliJ IDEA [12; 13]. Система, яка пропонується для реалізації, дозволить швидше та якісніше виконувати автоматизовані тести.

Виклад основного матеріалу. Система реалізована у вигляді фреймворка автоматизованого тестування веб-платформ. Середовище IntelliJ IDEA будує синтаксичне дерево, коли тільки вводимо код. В процесі вводу та виконання воно аналізує код, виявляє помилки та пропонує рішення, видаючи лише один варіант для автодоповнення, один, але вірний. Це робить розробку набагато більш оптимізованою за існуючі аналоги. Авторами для реалізації фреймворка обрано паттерн проектування PageFactory. Кожну веб-сторінку проекту описуємо як об'єкт класу. Взаємодію користувача наводимо у методах класу, а в тесті залишаємо лише бізнес-логіку. Паттерн Page Object у Selenium реалізовано за допомогою бібліотеки PageFactory та класу сторінки. Page Object є окремим класом, що містить локатори елементів, методи роботи з ними і конструктор, що приймає в якості параметру об'єкт WebDriver.

Методи класу Page Object можуть повертати об'єкти інших Page Object класів. За допомогою цього можна відтворити копію переходів та поведінки веб-програми. Одним із наслідків такого підходу є те, що необхідно моделювати як успішні так і неуспішні методи. Або, наприклад, якщо натискання на елемент може відкривати різні сторінки в залежності від умов, то також необхідно створювати різні методи для кожного необхідного випадку [12].

В роботі для тестування веб-сторінок браузерів застосовано Selenium WebDriver, який надає автотестам доступ до браузера [1; 3]. Він підтримує основні мови програмування: C #, Ruby, Java, Python, Perl, тощо; зв'язується із браузером; йому не потрібен Server, а необхідні три основні програмні компоненти: браузер, роботу якого треба автоматизувати; драйвер браузера для керування ним та скрипт/тест із набором команд для драйвера браузера. Головними особливостями реалізованого фреймворка є декларативний опис проекту (усі необхідні параметри налаштовуються за замовчанням) та гнучке управління залежностями. Maven вміє складати проекти підвантажувати до свого локального репозитарію сторонні бібліотеки, вибирати необхідну версію пакету та обробляти транзитивні залежності. Файл pom.xml у кореневому каталозі є файлом опису проекту, на основі якого здійснюються всі операції Maven.

```
<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
```

```
  <modelVersion>4.0.0</modelVersion>
  <groupId>seleniumTestNgFramework</groupId>
  <artifactId>seleniumTestNgFramework</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>jar</packaging>
  <name>seleniumTestNgFramework</name>
  <url>http://maven.apache.org</url>
  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>
  <build>
    <plugins>
      ...
    </plugins>
    <resources>
      ...
    </resources>
  </build>
  <dependencies>
    ...
  </dependencies>
</project>
```

Тег project є базовим та містить всю інформацію про проект. У заголовку вказано інформацію, необхідну Maven для розуміння файлу pom.xml. Тег modelVersion вказує на поточну версію POM. Ці два теги зазвичай генеруються автоматично, міняти їх не потрібно. Далі формується унікальний ідентифікатор проекту: теги groupId та artifactId. Тег version теж входить до цієї групи. Він зазвичай генерується та оновлюється автоматично. Після номера версії йде суфікс -SNAPSHOT. Він означає, що проект перебуває у стадії розробки. Коли програмне забезпечення випускається фреймворк прибирає цей суфікс, а якщо розробка продовжиться він автоматично збільшить номер версії. Разом ці три теги дозволяють однозначно ідентифікувати артефакт. Тег name містить ім'я артефакту, що відображається, а url - посилання на сайт. Оскільки сайт не заданий під час створення, pom.xml містить нагадування про це у вигляді коментаря. Крім того, можна додати короткий опис до description. Ці теги використовуються для формування документації. Далі йде дуже важливий блок dependencies. У ньому описуються всі використовані у проекті залежності. Кожну виділяємо тегом dependency та вказуємо унікальні ідентифікаційні дані. Maven сам завантажує транзитивні залежності. Крім того, за допомогою тега score можна вказати етапи, на яких використано артефакт. При цьому життєвий цикл Maven складається з 9 етапів. Автотестування відбувається у фоновому режимі і може проходити паралельно з роботою будь-яких програм. Метод прискорення швидкості виконання автоматизованих тестів базується на декількох основних принципах (рис. 1).

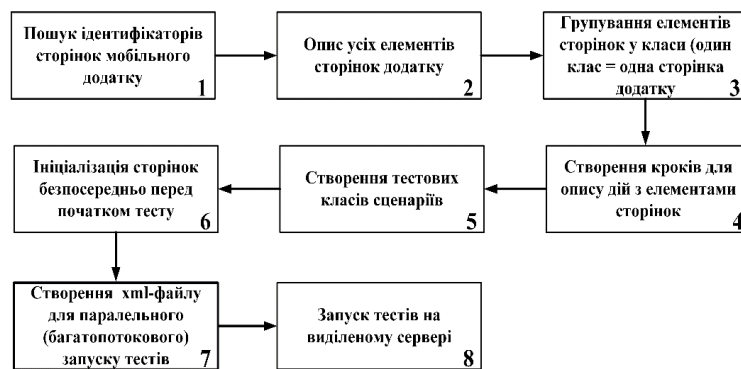


Рис. 1. Метод прискорення швидкості виконання тестів

Основні принципи роботи реалізованої системи автоматизації тестування (рис. 2): створення тестового сценарію; пошук та опис елементів веб-сторінки (використовуються різні види локаторів); групування усіх елементів в окремі класи; використання анотацій при створенні тестів; проведення попередньої ініціалізації об'єктів сторінок та реалізація багатопотоковості при запуску та виконанні автоматизованих тестів за допомогою TestNG.

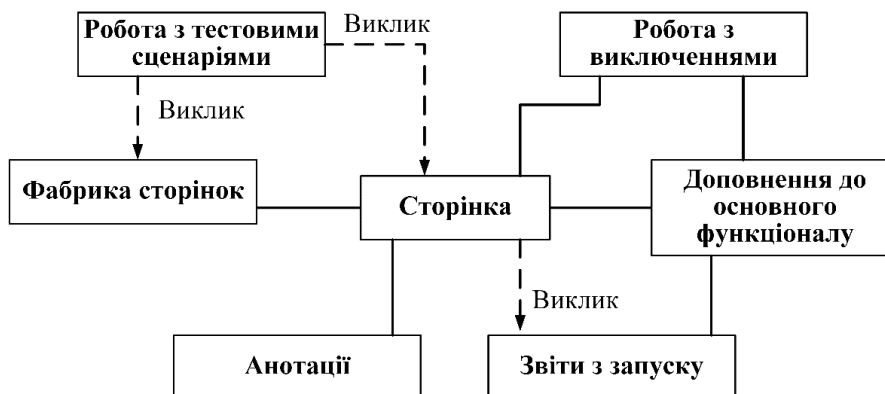


Рис. 2. Класи фреймворку для авто тестування

Алгоритм роботи системи для запуску тесту HomePageTest.java включає: запуск BasePage.java в якому реалізовані загальні методи обробки помилок; клас HomePage.java з ініціалізацією елементів веб-сторінки за допомогою локаторів та методи роботи з ними; BaseTest.java реалізує загальні пара-

метри та методи; HomePageTest.java реалізує виклик методів порівняння контенту за тест-кейсом. Усі інші тести, а саме LoginTest, LogoutTest, NegativeCartTest та PositiveCartTest реалізовано аналогічно. Основний функціонал є ядром роботи фреймворку, а службовий відповідає за надання додаткових функцій при реалізації тестів. Для створення звітності розроблено декілька видів звітів у форматі HTML.

Тестування системи. На основі проведених тестувань авторами узагальнені швидкісні показники системи, яка реалізована, та фреймворків Katalon та Ranorex. В процесі тестування проведено по десять запусків тест-ранів для тест-кейсів: HomePageTest, LoginTest та LogoutTest. Кожен тест-ран реалізовано із використанням фреймворків Katalon та Ranorex. У таблицях наведено час у секундах при виконанні автотесту для кожної із систем, що досліджуються. Результати спроб для HomePageTest тест-рану наведено у таблиці 1. У цьому класі запрограмовані переходи між сторінками: пошук по сайту; аналіз результатів пошуку та UI тестування переходу до корзини.

Таблиця 1

Результати тестів для HomePageTest тест-рану

тест-ран t, сек	1	2	3	4	5	6	7	8	9	10
Реалізована система, T ₁	51,545	57,327	50,036	47,457	51,672	55,431	53,214	48,016	52,567	58,318
Katalon, T ₂	54,543	52,755	48,645	55,532	57,145	58,116	49,056	51,355	55,014	53,558
Ranorex, T ₃	60,023	57,345	61,754	55,644	58,462	60,016	58,612	57,016	63,045	57,764

Розрахунок середньо арифметичного для часу виконання тестів із використанням кожної з досліджуваних систем при n=10, де n є кількістю спроб:

$$T_1 = \sum_{i=1}^n t_i = 52,558 \text{ сек.}; T_2 = \sum_{i=1}^n t_i = 53,571 \text{ сек.}; T_3 = \sum_{i=1}^n t_i = 58,968 \text{ сек.}$$

Графік порівняння отриманих результатів тестування зображено на рисунку 3.

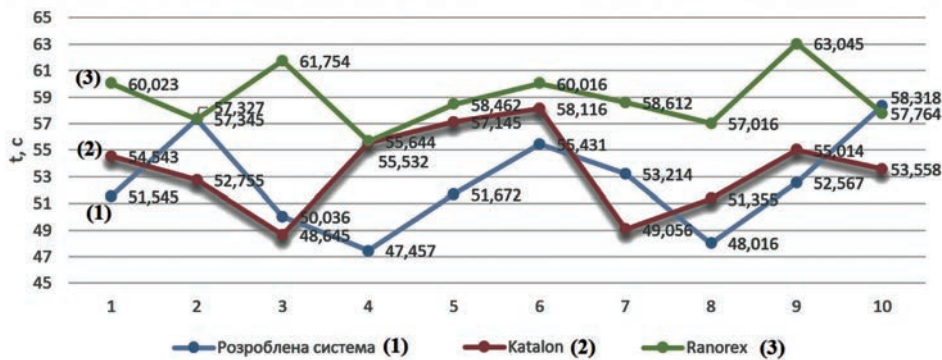


Рис. 3. Результат виконання тест-рану HomePageTest

Аналогічні результати (табл. 2) отримали для LoginTest тест-рану.

Таблиця 2

Результати тестів для LoginTest тест-рану

тест-ран t, сек	1	2	3	4	5	6	7	8	9	10
Реалізована система, T ₁	23,329	20,776	19,867	24,660	21,459	21,056	22,048	19,888	20,116	21,558
Katalon, T ₂	24,551	26,412	24,002	23,761	27,558	26,005	27,389	26,779	25,338	26,339
Ranorex, T ₃	27,337	28,004	27,118	26,003	29,408	26,117	28,450	26,449	27,448	26,709

Розрахунок середньо арифметичного для часу виконання тестів із використанням кожної з досліджуваних систем при n=10: T₁ = ∑_{i=1}ⁿ t_i = 21,475 сек.; T₂ = ∑_{i=1}ⁿ t_i = 25,813 сек.; T₃ = ∑_{i=1}ⁿ t_i = 27,304 сек.

Графічно результати дослідження для LoginTest тест-рану наведені на рисунку 4.

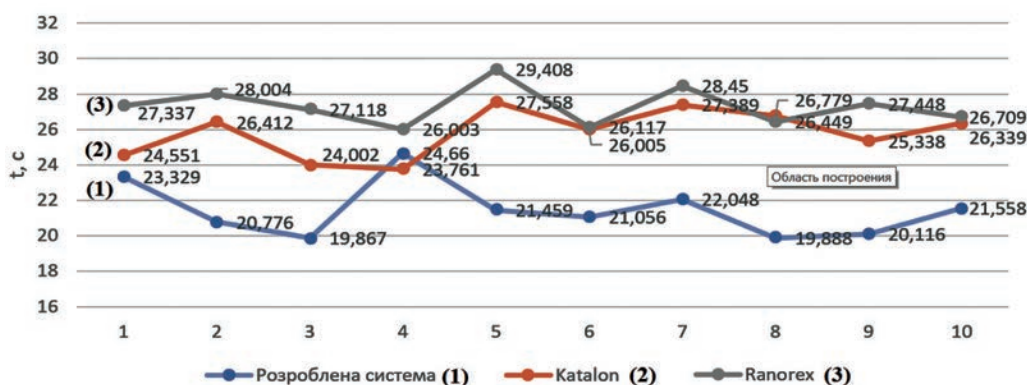


Рис. 4. Результат виконання тест-рану LoginTest

Далі наведемо результати дослідження при виконанні тест-рану LogoutTest, який перевіряє процес входу користувача до системи (табл. 3).

Таблиця 3

Результати тестів для LogoutTest тест-рану

тест-ран t, сек	1	2	3	4	5	6	7	8	9	10
Реалізована система, T ₁	34,772	32,004	33,568	32,038	35,001	34,117	32,882	31,669	34,002	33,006
Katalon, T ₂	35,885	34,641	32,038	32,278	33,110	34,725	31,890	33,955	34,805	35,679
Ranorex, T ₃	36,489	37,038	35,814	35,890	34,771	33,993	34,278	35,743	37,778	38,055

Розрахунок середньо арифметичного для часу виконання тестів із використанням кожної з досліджуваних систем при

$$n=10: T_1 = \sum_{i=1}^n t_i = 33,305 \text{ сек.}; T_2 = \sum_{i=1}^n t_i = 33,900 \text{ сек.}; T_3 = \sum_{i=1}^n t_i = 35,984 \text{ сек.}$$

Графічно результати дослідження наведені на рисунку 5. Аналізуючи результати тестувань можна зробити висновок, що за швидкісними показниками реалізована система має перевагу над двома іншими, порівнювальними системами. Окрім цього треба враховувати, що дослідження проводилось на короткотривалих тестах, тому для реальних систем, коли тести виконуються декілька годин, розрив у показниках швидкості буде більший.

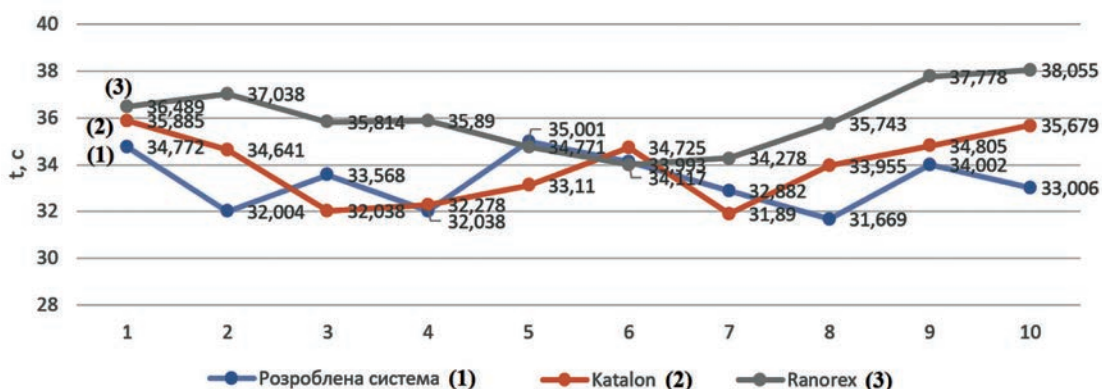


Рис. 5. Результат виконання тест-рану LogoutTest

Висновки. Всі елементи веб-сторінок згруповані в окремі класи – об'єкти сторінок. Це забезпечує швидке знаходження та заміну інформації на веб-сторінці. Структура тестів реалізована за допомогою анотацій, що забезпечує зрозумілу структуру для користувача та дає зрозуміти системі призначення коду, який був помічений. Окрім цього реалізована багатопотоковість при запуску та виконанні авто-

матизованих тестів за допомогою TestNG, що дозволяє виконувати декілька тестів одночасно. До переваг реалізованої системи відносимо: підтримку різних браузерів: Google Chrome, Mozilla Firefox, Edge; підтримку основних ОС: Windows, MacOS, Linux; кращі швидкісні характеристики у порівнянні з фреймворками Katalon та Ranorex; наявність детальної системи звітів по результатам тестування; умовна безкоштовність; реалізація проекту з відкритим вихідним кодом.

Список використаних джерел:

1. Krishna V. V., Gopinath G. Test Automation of Web Application Login Page by Using Selenium Ide in a Web Browser. Management. 2021. P. 713-732.
2. Rudkovskiy O. R., Kirichek G. G. Interaction support system of network applications. CEUR Workshop Proceedings 2832. 2020. P. 11-23.
3. Groeneveld F., Mesbah A., Van Deursen A. Automatic invariant detection in dynamic web application's. TUD-SERG-2010-037. 2010. P. 1-10.
4. Tiahunova M., Kyrychek H., Bohatyrova T., Moshynets D. System and method of automatic collection of objects in the room. CEUR Workshop Proceedings 3077. 2021. P. 174-186.
5. Altiero F. et al. Inspecting Code Churns to Prioritize Test Cases. IFIP. Springer, Cham. 2020. P. 272-285.
6. Srivastava N., Kumar U., Singh P. Software and Performance Testing Tools. Journal of Informatics Electrical and Electronics Engineering. 2021. 2(01). P. 1-12.
7. Okolnychiy A., Fögen K. A study of tools for behavior-driven development. Full-scale Software Engineering/Current Trends in Release Engineering. 2016. P. 7-12.
8. Lenka R. K., Mamgain S., Kumar S., Barik R. K. Performance Analysis of Automated Testing Tools: JMeter and TestComplete. IEEE. 2018. P. 399-407.
9. Bhargava S., Jain P. B. Software Quality Assurance Methodology with GUI Testing Tool: Ranorex. Journal of Software Engineering Tools & Technology Trends. 2018. 5(2). P. 11-17.
10. Bisht S. Robot framework test automation. Packt Publishing Ltd, 2013.
11. Tiahunova M., Tronkina O., Kirichek G., Skrupsky S. The neural network for emotions recognition under special conditions. CEUR Workshop Proceedings 2864. 2021. P. 121-134.
12. Khan R., Qahmash A., Hussain M. R. Automatic Testing for Web Application Using HP-ALM Tool. International Journal of Engineering Research and Technology. 13(12). 2020. P. 4662-4665.
13. Mann M., Tomar P., Sangwan O. P. Automated software test optimization using test language processing. Int. Arab J. Inf. Technol. 2019. 16(3). P. 348-356.

References:

1. Krishna, V. V., Gopinath, G. (2021). Test Automation of Web Application Login Page by Using Selenium Ide in a Web Browser. Management (pp. 713-732).
2. Rudkovskiy, O. R., Kirichek, G. G. (2020). Interaction support system of network applications. In CEUR Workshop Proceedings 2832 (pp. 11-23).
3. Groeneveld, F., Mesbah, A., Van Deursen, A. (2010). Automatic invariant detection in dynamic web applications. Technical Report Series TUD-SERG-2010-037
4. Tiahunova, M., Kyrychek, H., Bohatyrova, T., Moshynets, D. (2021). System and method of automatic collection of objects in the room. In CEUR Workshop Proceedings 3077 (pp. 174-186).
5. Altiero, F., Corazza, A., Di Martino, S., Peron, A., Starace, L. L. L. (2020). Inspecting Code Churns to Prioritize Test Cases. In IFIP International Conference on Testing Software and Systems (pp. 272-285). Springer, Cham.
6. Srivastava, N., Kumar, U., Singh, P. (2021). Software and Performance Testing Tools. Journal of Informatics Electrical and Electronics Engineering, 2(01), 1-12.
7. Okolnychiy, A., Fögen, K. (2016). A study of tools for behavior-driven development. Full-scale Software Engineering/Current Trends in Release Engineering, 7-12.
8. Lenka, R. K., Mamgain, S., Kumar, S., Barik, R. K. (2018). Performance Analysis of Automated Testing Tools: JMeter and TestComplete. In International Conference on Advances in Computing, Communication Control and Networking (ICACCCN) (pp. 399-407). IEEE.
9. Bhargava, S., Jain, P. B. (2018). Software Quality Assurance Methodology with GUI Testing Tool: Ranorex. Journal of Software Engineering Tools & Technology Trends, 5(2), 11-17.
10. Bisht, S. (2013). Robot framework test automation. Packt Publishing Ltd.
11. Tiahunova, M., Tronkina, O., Kirichek, G., Skrupsky, S. (2021). The neural network for emotions recognition under special conditions. In CEUR Workshop Proceedings 2864 (pp. 121-134).
12. Khan, R., Qahmash, A., Hussain, M. R. (2020). Automatic Testing for Web Application Using HP-ALM Tool. International Journal of Engineering Research and Technology, 13(12), 4662-4665.
13. Mann, M., Tomar, P., Sangwan, O. P. (2019). Automated software test optimization using test language processing. Int. Arab J. Inf. Technol., 16(3), 348-356.

УДК 004.664

DOI <https://doi.org/10.32689/maup.it.2022.1.5>

Олена КРИВОРУЧКО

доктор технічних наук, професор, завідувач кафедри інженерії програмного забезпечення та кібербезпеки, Державний торговельно-економічний університет, вул. Кіото, 19, Київ, Україна, індекс 02000 (ev_kryvoruchko@ukr.net)

ORCID: 0000-0002-7661-9227

Юлія КОСТЮК

здобувач PhD, асистент кафедри інженерії програмного забезпечення та кібербезпеки, Державний торговельно-економічний університет, вул. Кіото, 19, Київ, Україна, індекс 02000 (kostyuk.yu@ukr.net)

ORCID: 0000-0001-5423-0985

Olena KRYVORUCHKO

DSc (Eng.), Professor, Head of the Department of at Software Engineering and Cyber Security, Kyiv National University of Trade and Economics, 19 Kyoto str., Kyiv, Ukraine, postal code 02000 (ev_kryvoruchko@ukr.net)

Yuliia KOSTIUK

Assistant at the Department of at Software Engineering and Cyber Security, Kyiv National University of Trade and Economics, 19 Kyoto str., Kyiv, Ukraine, postal code 02000 (kostyuk.yu@ukr.net)

Бібліографічний опис статті: Криворучко О., Костюк Ю. Структурно-функціональне моделювання технологічного процесу виробництва вершкового масла. *Інформаційні технології та суспільство*. 2022. Вип. 1 (3). С. 38–44. DOI: <https://doi.org/10.32689/maup.it.2022.1.5>

Bibliographic description of the article: Kryvoruchko, O., Kostiuk, Yu. (2022). Strukturno-funktsionalne modeliuвання tekhnolohichnoho protsesu vyrobnytstva vershkovoho masla [Structural and functional modeling of the technological process of butter production]. *Informatsiini tekhnolohii ta suspilstvo – Information echnology and society*, 1 (3), 38–44. DOI: <https://doi.org/10.32689/maup.it.2022.1.5>

СТРУКТУРНО-ФУНКЦІОНАЛЬНЕ МОДЕЛЮВАННЯ ТЕХНОЛОГІЧНОГО ПРОЦЕСУ ВИРОБНИЦТВА ВЕРШКОВОГО МАСЛА

Анотація. Досліджується інформаційно-технологічна модель процесу виробництва вершкового масла, яке характеризується великою кількістю технологічних параметрів та вимагає постійного оперативного контролю та реагування на можливі ситуації. **Метою статті** є побудова структурно-функціональної моделі технологічного процесу виробництва вершкового масла, що дозволить наочно відобразити стан всіх процесів і етапів виробництва з точки зору інформаційних потоків, ресурсів та технологічних регламентів. Реалізація поставленої мети передбачає використання **методології IDEF0 CASE-** технології із застосуванням програмного продукту AllFusion Process Modeler 7 (BPwin), як невід'ємну складову впровадження інформаційних технологій при вдосконаленні бізнес-процесів. **Наукова новизна.** У статті побудовані контекстні діаграми процесу виробництва вершкового масла за стандартом IDEF0. Побудовано діаграму користувачів для підсистеми діагностики та прогнозування. **Висновки.** Контекстні діаграми процесу виробництва вершкового масла дозволили описати інформаційні потоки у вигляді ресурсів, стандартів, технологічного регламенту, механізмів, які задіяні та впливають на сам процес виробництва вершкового масла за стандартом IDEF0. Під час структурно-функціонального моделювання було виділено проблему стабілізації вмісту вологи у вершковому маслі в умовах змінного складу сировини по фізико-хімічним параметрам. Була розроблена діаграма користувача із використати об'єктно-орієнтованої мови моделювання UML для підсистеми діагностики та прогнозування проходження процесу збивання вершків у масло для контролю та прогнозування можливих відхилень вмісту вологи у маслі в залежності від виробничої ситуації, яка дозволить зменшити час реагування на можливі причини виникнення відхилень під час технологічного процесу.

Ключові слова: інформаційна система, контекстна діаграма, вершкове масло, діагностика, прогнозування.

STRUCTURAL-FUNCTIONAL MODELING OF TECHNOLOGICAL PROCESS OF CREAM OIL PRODUCTION

Abstract. The information-technological model of the butter production process is studied, which is characterized by a large number of technological parameters and requires constant operational control and response to possible situations. **The aim** of the article is to build a structural and functional model of the technological process of butter production, which will clearly reflect the state of all processes and stages of production in terms of information flows, resources and technological regulations. Achieving this goal involves the use of the methodology IDEF0 CASE-technology using the software product

AllFusion Process Modeler 7 (BPwin), as an integral part of the implementation of information technology in improving business processes. **Scientific novelty.** The article constructs contextual diagrams of the butter production process according to the IDEF0 standard. The diagram of users for the subsystem of diagnostics and forecasting is constructed. **Conclusions.** Context diagrams of the butter production process allowed to describe information flows in the form of resources, standards, technological regulations, mechanisms that are involved and influence the process of butter production according to the IDEF0 standard. During the structural-functional modeling the problem of stabilization of moisture content in butter in the conditions of variable composition of raw materials on physical and chemical parameters was singled out. A user diagram was developed to use the object-oriented modeling language UML for the subsystem for diagnosing and predicting the process of whipping cream into butter to control and predict possible deviations of moisture in the oil depending on the production situation, which will reduce response time to possible causes deviations during the technological process.

Key words: information system, context diagram, butter, diagnostics, forecasting.

Актуальність. Організація та управління виробничими та інформаційним процесами на підприємстві є одним із важливих напрямків для ефективного керування в цілому. Кожне підприємство характеризується великою кількістю процесів, які вимагають постійного контролю та коригування [1–2]. Саме тому, постає питання підвищення інформативності та керованості виробничими процесами за рахунок застосування різноманітних програмних продуктів CASE- технологій, котрі можуть бути використані для опису як окремих етапів виробничих процесів, так і охоплювати всі процеси в цілому. Структурно-функціональні моделі CASE-технологій дозволяють представити технологічний процес виробництва вершкового масла у форматі відображення структур та функцій системи, інформаційних потоків та матеріальних об'єктів на основі методології моделювання IDEF0.

Аналіз останніх досліджень та публікацій. На даний час достатньо висвітлено питання застосування CASE- технологій для моделювання різного роду виробничих та бізнес-процесів, які викладені у вигляді наукових статей, монографій, навчальних посібників та інших матеріалів [3–7]. Так, Цюцюра М.І. та Демідов П.Г. [8] описали технологію розробки баз даних ERP – системи управління промисловим підприємством, Бреус Н.М. [9] застосувала структурно-функціональне моделювання для опису технології виробництва морозива. Управління технологічним процесом включає заходи планування та забезпечення виробничої діяльності підприємства, які змінюються за рахунок впровадження вдосконалених технологій та засобів автоматизації. Тому, виникає потреба у застосуванні програмно-технічних засобів, які використовують CASE- технологію методології проектування інформаційних систем. Така технологія побудована на основі структурного та об'єктно-орієнтованого аналізу бізнес-процесів у вигляді діаграм для опису зв'язків між моделями системи.

Метою дослідження є побудова структурно-функціональної моделі технологічного процесу виробництва вершкового масла, що дозволить наочно відобразити стан всіх процесів і етапів виробництва з точки зору інформаційних потоків, ресурсів та технологічних регламентів.

Виклад основного матеріалу. У сучасному світі інформаційні системи стали необхідним інструментом, що охоплює різні сфери діяльності людини. Із розвитком інформаційних технологій та складністю об'єктів керування зростає й складність самих функціональних та інформаційних моделей таких систем.

Виділення бізнес-процесів для підприємства є досить важливим етапом структуризації технологічного процесу виробництва продукції. Це дозволяє виділити основні та додаткові процеси. Так, основні процеси описують саме виробництво продукції, а додаткові – допомагають підвищити її інформативність та якість.

Основу сучасного підходу до організації методів керування технологічного процесу складає реінжиніринг бізнес-процесів, який має на меті фундаментальне переосмислення всіх бізнес-процесів на підприємстві, що направлені на підвищення основних показників, а саме собівартості, якості та продуктивності. Реінжиніринг використовує засоби, методи та відповідні інформаційні технології, які направлені на процес стратегічного інноваційного розвитку із збільшенням автоматизації існуючих процесів.

Для створення інформаційної технологічної моделі виробництва вершкового масла використовується CASE- технологія із застосуванням програмного продукту AllFusion Process Modeler 7 (BPwin). Методологія такого моделювання дозволяє описати систему в цілому у вигляді взаємозв'язаних дій та функцій [8]. BPwin підтримує стандарти IDEF0, DFD, IDEF3, які дозволяють оптимізувати процедури в компанії на основі стандарту якості ISO9000, має вбудований генератор звітів та засоби документування моделей та проектів.

Методологія IDEF0 базується на трьох основних елементах:

- 1) функціональні блоки;
- 2) дуги взаємодій;
- 3) принципи декомпозиції розбиття складного процесу на його складові.

Побудова структурно-функціональної моделі виробництва вершкового масла (рис. 1) з урахуванням інформаційної складової, що включає методи діагностики та прогнозування ведення технологічного процесу починається із контекстної діаграми верхнього рівня.

На діаграмі технологічний процес виробництва вершкового масла представлений у вигляді функціонального блоку, який містить вхідну та вихідну інформацію (ресурси, показники якості, результати), механізми і керуючі впливи. До механізмів впливу відносяться: головний технолог, устаткування, яке бере участь у процесі та методи діагностики та прогнозування у вигляді карт Шухарта та нейронних мереж. До керуючих впливів відносяться стандарти ДСТУ та ISO, а також виробничі завдання. На виході отримуємо результати перетворення інформаційних та матеріальних потоків вхідної інформації (даних про якість початкової сировини, початкової та допоміжної сировини) у вигляді готової продукції, інформації про її якість та відходів виробництва (маслянка).

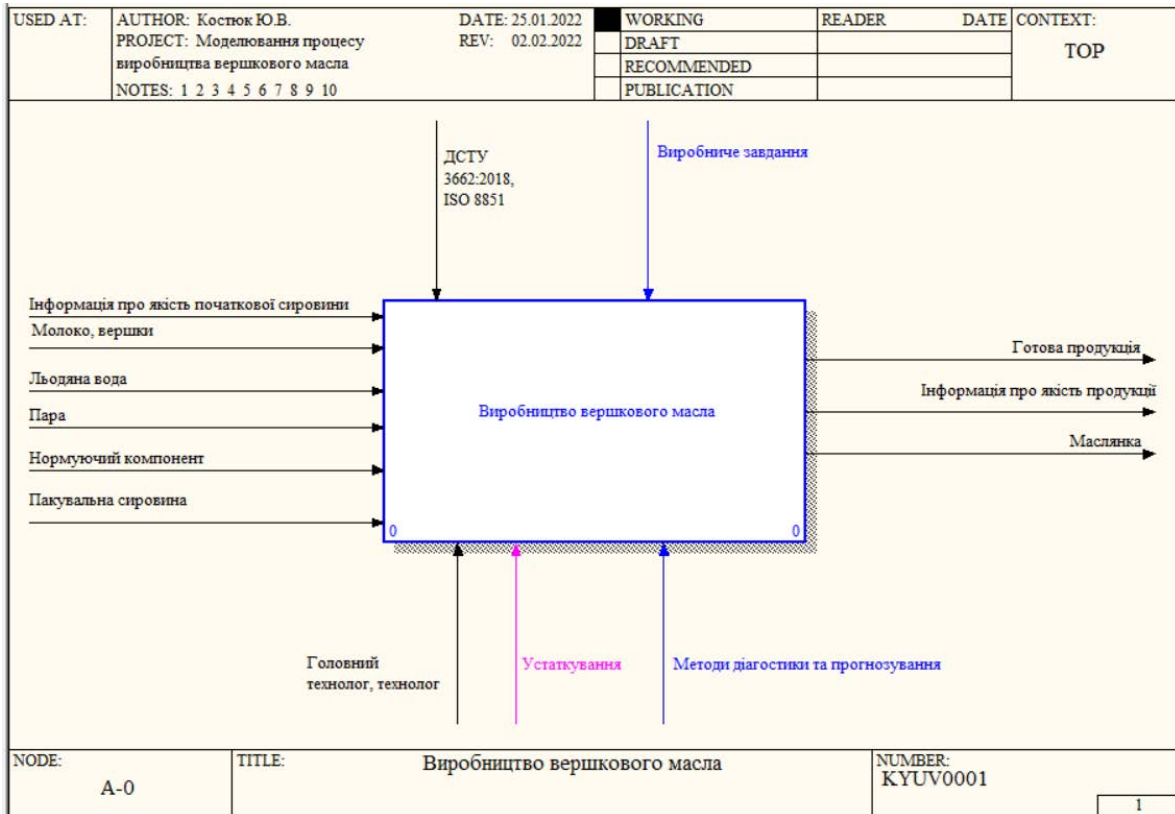


Рис. 1. Верхній рівень технологічної системи «Виробництво вершкового масла»

Для детального опису процесу виробництва вершкового масла було виконана декомпозиція верхнього рівня на наступні етапи (рис. 2): визначити рецептуру та скласти оперативну інформацію, підготувати сировину та інгредієнти, отримати вершкове масло, провести пакування продукції. На першому етапі відповідно до інформації про якість початкової сировини головним технолог керуючись ДСТУ 3662:2018 та ISO 8851 із використанням методів прогнозування складає оперативну інформацію, яка буде слугувати керуючими впливами для всіх наступних етапів виробництва вершкового масла. На другому етапі відбувається підготовка вхідної сировини, а саме вершків із залученням технологів або головного технолога та устаткування відповідно до нормативних документів та розроблених виробничих завдань.

На етапі підготовки сировини та інгредієнтів проводять контроль якості та сортування молока або вершків. Контролю піддають кожен партію молока або вершків одного сорту, що надійшло на виробництво. Далі визначають наступні показники молока або вершків згідно ДСТУ 3662:2018: органолептичні (смак, запах, колір, консистенцію), фізико-хімічні (температуру, кислотність, масову частку жиру, масову частку білка, сухих речовин, густину, механічну забрудненість) та мікробіологічні показники (загальну бакобсеменінність, кількість соматичних клітин, інгібувальні речовини, антибіотики). Сюди входить процес сепарування молока, який ведуть при температурі 35–40°C і кислотності не більше 20 од. рН. На виробництво масла направляються вершки однакової жирності, так як інакше будуть потрібні різні умови підготовки їх до збивання. Для виробництва вершкового масла використовуються вершки з масовою часткою жиру 32–50%. Для пастеризації вершків застосовують пастеризаційно-

охолоджувальні установки. Виходом даного етапу є вершки пастеризовані, які надходять на етап отримання вершкового масла.

На третьому етапі відбувається отримання вершкового масла із вершків методом безперервного збивання відповідно до ДСТУ 3662:2018 та ISO 8851, керуючись виробничим завданням та оперативною рецептурою. Механізмами даного етапу є масловиготовлювач, технолог або головний технолог та методи діагностики та прогнозування. Вихідну інформацію формують вершкове масло, яке є вхідним елементом для етапу пакування продукції, а також інформація про якість готової продукції та маслянка, яка утворюється під час збивання вершків у масло.

Четвертий етап описує процес пакування продукції із використанням пакувальних механізмів під контролем технологів виробництва. Вхідним параметром слугує власне вершкове масло та пакувальна сировина відповідно до масової частки жиру у маслі. Результатом такого етапу є готова продукція.

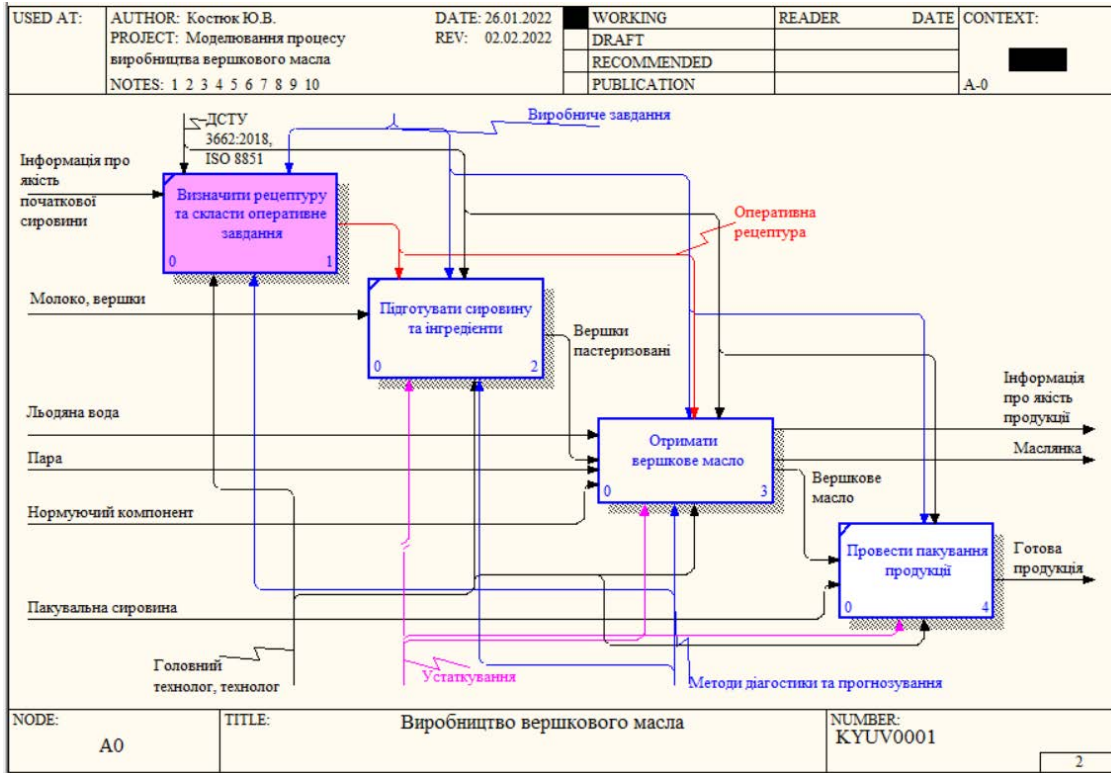


Рис. 2. Декомпозиція верхнього рівня технологічної системи «Виробництво вершкового масла»

Проведемо декомпозицію етапу отримання вершкового масла (рис. 3) на три етапи, а саме: процес дозрівання вершків, процес збивання вершків та оцінку показників якості. Даний етап є дуже важливим, оскільки він залежить від багатьох чинників, які впливають на якість готової продукції, адже вершкове масло це високожирний молочний продукт.

Процес дозрівання вершків проводиться для затвердіння молочного жиру і фізико-хімічних змін оболонки жирових кульок. Метою дозрівання є переведення рідкого жиру у твердий стан. На вміст вологи у кінцевому продукті на стадії дозрівання вершків у значній мірі впливає склад та фізико-хімічні властивості вершків, що надходять на збивання. Температуру збивання вершків встановлюють залежно від масової частки жиру у вершках, періоду року, режимів дозрівання вершків, конструкції масловиготовлювача.

Процес збивання вершків у масло - складний процес, який залежить від багатьох чинників. Основними з яких є: температура вершків, волога у маслі, частота обертання мішалки барабана збивального пристрою, температура збивання вершків, об'єм вершків, рівень маслянки у шнековій камері. Ступінь заповнення масловиготовлювача вершками впливає на тривалість збивання вершків. Оптимальною вважають ступінь заповнення 40–50%. До найважливіших технологічними параметрами, що впливає на якість готового вершкового масла відноситься температура збивання. При зниженні температурі подовжується тривалість збивання, що веде до вироблення масла з невиробленою вологою і засаленої консистенцією. Завищення температури збивання призводить до підвищення жирності і отримання

масла з м'якою консистенцією. Тривалість збивання становила 45–60 хв. Скорочення тривалості збивання призводить до погіршення якості масляного зерна і значного відходу жиру з масляною. При збільшенні тривалості збивання масляне зерно виходить занадто тверде, пружне, воно погано обробляється, а отримане масло може мати грубу консистенцію.

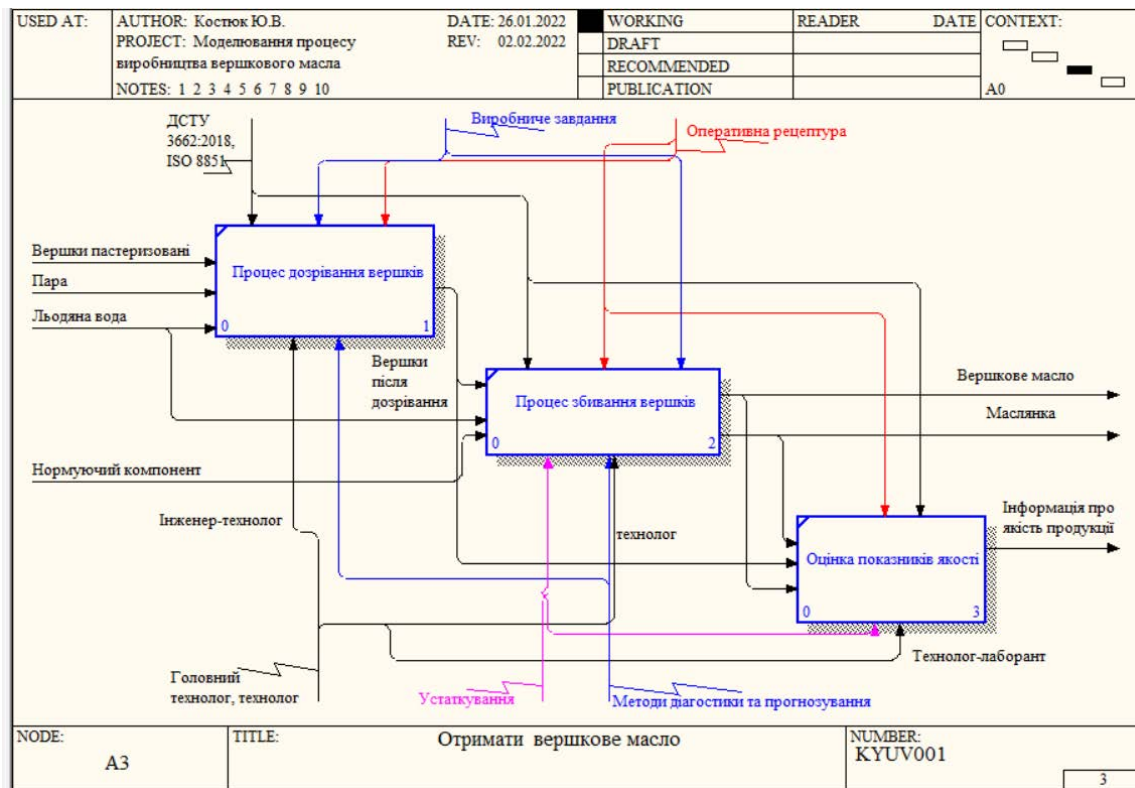


Рис. 3. Декомпозиція операції «Отримати вершкове масло»

На етапі оцінки показників якості технолог-лаборант із використанням спеціального устаткування згідно ДСТУ 3662:2018 та ISO 8851 визначає основні показники якості готового вершкового масла: органолептичні показники, температура, масова частка жиру, масова частка сухого знежиреного залишку, кислотність плазми та жирової фази масла тощо. Якісні показники масла багато в чому характеризують його зовнішній вигляд (структура, колір, оформлення). Смак і запах, колір як показники якості вершкового масла практично зумовлюються, в першу чергу, якістю вихідної сировини. Багато в чому якість і властивості масла залежить від методів переробки вершків, початкової сировини, смакових і ароматичних добавок. В технологічному процесі виробництва вершкового масла основним параметром, який визначає якість вершкового масла є його масова частка вологи у готовому продукті. Якісні показники вершкового масла, в тому числі і масова частка вологи, залежать від фізико-хімічних параметрів початкової та проміжної сировини, що ускладнюють ведення самого процесу. На складність процесу впливає також перехідний процес масловиготовлювача, який пов'язаний із послідовним підключенням резервуарів із вершками після стадії дозрівання для забезпечення неперервності технологічного процесу.

Отже, під час проведення структурно-функціонального аналізу процесу виробництва вершкового масла з використанням середовища AllFusion Process Modeler 7 (BPwin) було виділено проблему стабілізації вмісту вологи у вершковому маслі в умовах змінного складу сировини по фізико-хімічним параметрам, вимагають оперативного керування і, відповідно, забезпечення заданого вмісту вологи в допустимих відхиленнях.

Для вирішення задачі удосконалення інформаційної системи підтримки прийняття рішень щодо стабілізації вмісту вологи у маслі, необхідно розробити підсистему оперативного керування на основі методів діагностики та прогнозування можливих відхилень вмісту вологи у готовому продукті під час процесу збивання вершків у масло, яка дозволить зменшити час реагування на можливі причини виникнення відхилень під час технологічного процесу. Постійне вдосконалення систем підтримки прийняття рішень разом із причинно-наслідковими зв'язками утворюють логічну основу всієї діяльності.

Тому, варто використати об'єктно-орієнтовану мову моделювання UML [10] у вигляді діаграми функцій користувача (Use Case diagram) для підсистеми діагностики та прогнозування. Основні елементи даної діаграми представлені на рис. 4.

Сам процес прийняття рішень складний та багатограний, який містить послідовні етапи заходи. Процес прийняття рішень розпочинається з моменту отримання повноти інформації щодо проходження технологічного процесу. Вивчення та ідентифікація ситуації відбувається на основі вхідної інформації із застосуванням аналітичного мислення, аналогії, порівнянь, аналізу тощо.

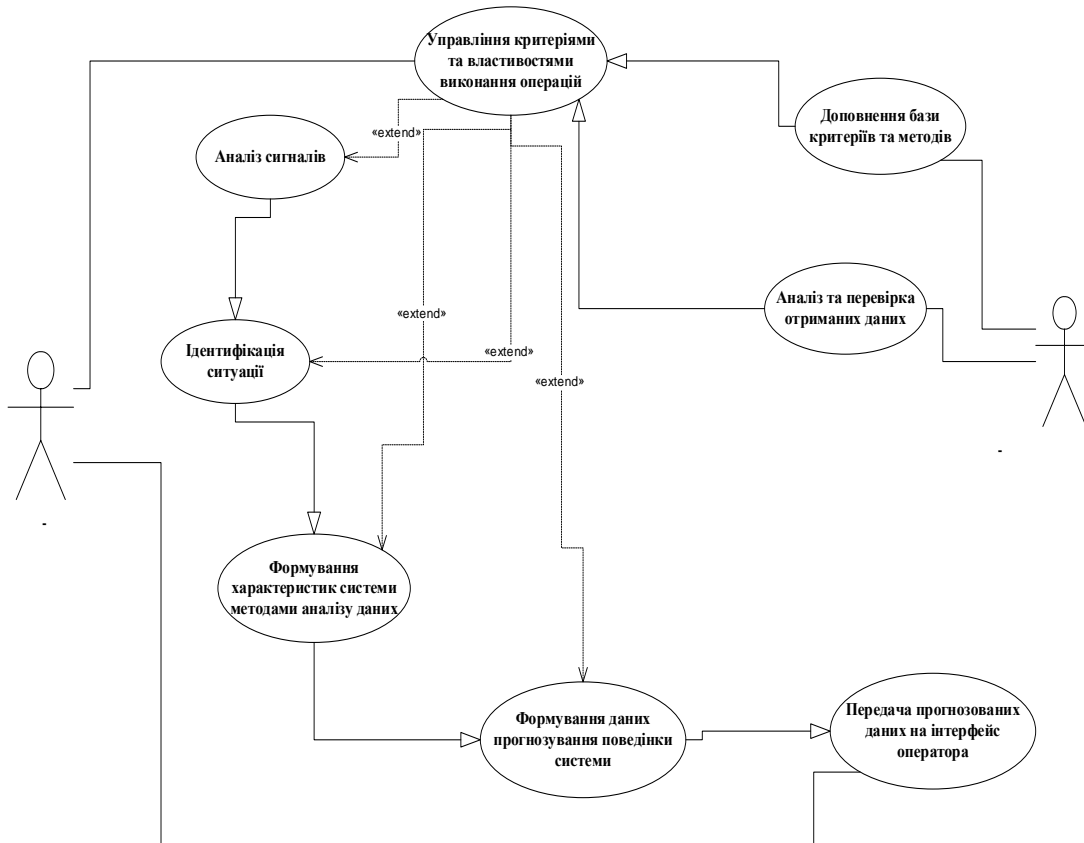


Рис. 4. Діаграми функцій користувача для підсистеми діагностики та прогнозування

Для обробки отриманої інформації застосовуються методи аналізу даних: статистичні, регресійний аналіз, логіко-математичні тощо. Результатом такого етапу виступають структуровані та проаналізовані дані, які поєднуються із об'єктивними та суб'єктивними факторами, що дозволяють оператору-технологу чітко сформулювати характеристику та ступінь важливості проблеми, яка виникла, поставити мету та цілі, яких необхідно досягти при вирішенні даної проблеми.

При формуванні альтернативних рішень оператор-технолог в певній мірі керується невизначеністю ситуації. Тому, наступним етапом є прогнозування можливих сценаріїв розвитку ситуації із використанням інтелектуальних методів, а саме нейронної мережі, яка на основі експериментальних даних навчена за певним алгоритмом та протестована.

Після проведення прогнозування існує можливість розглянути можливі варіанти, які порівнюються із критеріями для цієї конкретної проблеми. Вибір оптимального рішення для конкретних умов із забезпечення ефективного керування відбувається на основі результатів порівняння можливих варіантів розвитку подій. Таким чином, для оператора-технолога процес прийняття рішень є досить складним і містить безліч етапів, які залежать від складності проблеми, що вирішується.

Наступним важливим етапом є впровадження рішення та моніторинг за його виконанням. Ефективність такого моніторингу не можливий без чіткого опису робіт, які повинні бути виконані усіма учасниками. Тут варто враховувати кваліфікацію виконавців та їх досвід. В кінцевому випадку, результати виконання керуючих дій повинні бути узагальнені незалежно від стану реалізації та перевірені на їх ефективність для подальшого прийняття та реалізації в аналогічних ситуаціях.

Висновки. Для аналізу процесу та етапів виробництва вершкового масла було використано структурно-функціональне моделювання з подальшою декомпозицією технологічного процесу «AS-IS» у стандарті IDEF0. Контекстні діаграми дозволили описати інформаційні потоки у вигляді ресурсів, стандартів, технологічного регламенту, механізмів, які задіяні та впливають на сам процес виробництва вершкового масла. Під час структурно-функціонального моделювання було виділено проблему стабілізації вмісту вологи у вершковому маслі в умовах змінного складу сировини по фізико-хімічним параметрам. Тому виникає потреба у оперативному керуванні технологічним процесом для забезпечення заданого вмісту вологи в допустимих відхиленнях і, відповідно, забезпечення належної якості готового продукту. Була розроблена діаграма користувача із використати об'єктно-орієнтованої мови моделювання UML для підсистеми діагностики та прогнозування проходження процесу збивання вершків у масло для контролю та прогнозування можливих відхилень вмісту вологи у маслі в залежності від виробничої ситуації, яка дозволить зменшити час реагування на можливі причини виникнення відхилень під час технологічного процесу.

Список використаних джерел:

1. Криворучко О. В., Костюк Ю. В., Самойленко Ю. О. Інформаційна підсистема контролю якості продукції з використанням карт Шухарта. *Управління розвитком складних систем*. 2021. № 47. С. 190–195, dx.doi.org\10.32347/2412-9933.2021.47.190-195.
2. Криворучко О.В., Костюк Ю.В., Самойленко Ю.О. Формування підсистеми підтримки прийняття рішень процесом виробництва вершкового масла. *Specialized and multidisciplinary scientific researches: Collection of scientific papers «ΛΟΓΟΣ» with Proceedings of the International Scientific and Practical Conference (Vol. 2), December 11, 2020. Amsterdam, The Netherland: European Scientific Platform*, P.p. 125–126.
3. Bruce Silver. *BPMN Method and Style: A levels-based methodology for BPM process modeling and improvement using BPMN 2.0*. Cody-Cassidy. 2009. 236 p.
4. Stephen A. White, Derek Miers. *BPMN Modeling and Reference Guide: Understanding and Using BPMN*. Future Strategies Inc., 2008. 226 p.
5. What is business process management (BPM). URL: <https://www.bizagi.com/en/blog/what-is-business-process-management-bpm> (дата звернення: 25.01.2022)
6. Маклаков С.В. BPwin и ERwin: CASE-средства для разработки информационных систем. Москва : Диалог-МИФИ, 2000. 43 с.
7. Маклаков С.В. Моделирование бизнес-процессов с ALLFusion Process Modeler. Москва : Диалог-МИФИ, 2008. 240 с.
8. Демідов П.Г., М.І. Цюцюра. Технології розробки бази даних ERP-системи управління промисловим підприємством. *Управління розвитком складних систем*. 2017. № 30. С. 101–107.
9. Бреус Н.М. Інформаційна технологія моделювання рецептур морозива: дис. канд. техн. наук: 05.13.06. Київ, 2019. 153 с.
10. Фаулер М. UML. Основы, 3-е издание (пер. с англ.). Санкт-Петербург : Символ-Плюс, 2004. 192 с.

References:

1. Kryvoruchko, Olena, Kostiuk, Yuliia & Samoilenko, Yuliia. (2021). Informatsiina pidsystema kontroliu yakosti produktsii z vykorystanniam kart Shukharta [Information subsystem of product quality using Shuhart chart]. *Management of Development of Complex Systems*, 47, 190–195, dx.doi.org\10.32347/2412- 9933.2021.47.190-195. [in Ukrainian].
2. Kryvoruchko, O., Kostiuk, Yu., & Samoilenko, Yu. (2020). Formuvannia pidsystemy pidtrymky pryiniattia rishen protsesom vyrobnytstva vershkovoho masla [Formation of a decision support subsystem in the process of butter production]. *Specialized and multidisciplinary scientific researches: Collection of scientific papers «ΛΟΓΟΣ» with Proceedings of the International Scientific and Practical Conference (Vol. 2), December 11, 2020. Amsterdam, The Netherland: European Scientific Platform*, P.p. 125-126. [in Ukrainian].
3. Bruce Silver. (2009). *BPMN Method and Style: A levels-based methodology for BPM process modeling and improvement using BPMN 2.0*. Cody-Cassidy [in English].
4. Stephen A. White, Derek Miers. (2008). *BPMN Modeling and Reference Guide: Understanding and Using BPMN*. USA, Future Strategies Inc [in English].
5. What is business process management (BPM). Retrieved from: <https://www.bizagi.com/en/blog/what-is-business-process-management-bpm> [in English].
6. Maklakov, S.V. (2000). *BPwin y ERwin: CASE-sredstva dlia razrabotky ynformatsyonnykh system [BPwin and ERwin: CASE tools for information systems development]*. Moscow: Dyaloh-MYFY. [in Russian].
7. Maklakov, S.V. (2008). *Modelyrovanye byznes-protsessov s ALLFusion Process Modeler [Business process modeling with ALLFusion Process Modeler]*. Moscow: Dyaloh-MYFY [in Russian].
8. Demidov, P.H. & Tsiutsiura, M.I. (2017). Tekhnolohii rozrobky bazy danykh ERP-systemy upravlinnia promyslovym pidpriemstvom [Technologies for developing an ERP database for industrial enterprise management systems]. *Upravlinnia rozvytkom skladnykh system*. 30, 101 – 107. [in Ukrainian].
9. Breus, N.M. (2019). Informatsiina tekhnolohiia modeliuvannia retseptur morozyva [Information technology for modeling ice cream recipes]. Candidate's thesis Kyiv: NUFT [in Ukrainian].
10. Fauler M. (2004). *UML. Osnovy, 3-e izdanie (per. s angl.) [Fundamentals, 3rd edition (translated from English)]*. Spb.: Simvol-Plyus [in Russian].

УДК 681.5.01:004.94

DOI <https://doi.org/10.32689/maup.it.2022.1.6>

Іван МАРИНИЧ

кандидат технічних наук, доцент, доцент кафедри автоматизації, комп'ютерних наук і технологій, Криворізький національний університет, вул. В. Матусевича, 11, Кривий Ріг, Дніпропетровська область, Україна, індекс 50011 (marynych@knu.edu.ua)

ORCID: 0000-0002-9036-8532

Ольга СЕРДЮК

кандидат технічних наук, старший викладач кафедри автоматизації, комп'ютерних наук і технологій, Криворізький національний університет, вул. В. Матусевича, 11, Кривий Ріг, Дніпропетровська область, Україна, індекс 50011 (olgajs28@knu.edu.ua)

ORCID: 0000-0003-0505-0800

Ivan MARYNYCH

Candidate of Technical Sciences, Associate Professor, Associate Professor at the Department of Automation, Computer Science and Technology, Kryvyi Rih National University, 11 V. Matusevicha str., Kryvyi Rih, Dnipro region, Ukraine, postal code 50011 (marynych@knu.edu.ua)

Olga SERDIUK

Candidate of Engineering Science, Senior Lecturer at the Department of Automation, Computer Science and Technology, Kryvyi Rih National University, 11 V. Matusevicha str., Kryvyi Rih, Dnipro region, Ukraine, postal code 50011 (olgajs28@knu.edu.ua)

Бібліографічний опис статті: Маринич І., Сердюк О. Застосування нейронних регуляторів при моделюванні керування стадією подрібнення в умовах гірничо-збагачувального комбінату. *Інформаційні технології та суспільство*. 2022. Вип. 1 (3). С. 45–53. DOI: <https://doi.org/10.32689/maup.it.2022.1.6>

Bibliographic description of the article: Marynych, I., Sediuk, O. (2022). Zastosuvannya neyronnih regulyatoriv pri modelyuvanni keruvannya stadiyeu podribnennya v umovah girnicho-zbagachuvalnogo kombinatu [Use of neural regulators in control simulation of crushing stage in conditions of mining and processing plant]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, 1 (3), 45–53. DOI: <https://doi.org/10.32689/maup.it.2022.1.6>

ЗАСТОСУВАННЯ НЕЙРОННИХ РЕГУЛЯТОРІВ ПРИ МОДЕЛЮВАННІ КЕРУВАННЯ СТАДІЄЮ ПОДРІБНЕННЯ В УМОВАХ ГІРНИЧО-ЗБАГАЧУВАЛЬНОГО КОМБІНАТУ

Анотація. Стаття присвячена можливості застосування стандартних типів нейрорегуляторів, що пропонує середовище MATLAB & Simulink при моделюванні керування технологічним процесом, а саме стадією подрібнення, шляхом застосування узгодженого інтелектуального керування в умовах невизначеності. Застосування технологій штучного інтелекту в гірському ділі є досить актуальним в цей час. На відміну від «класичних» детермінованих автоматизованих систем керування, які засновані на використанні жорстких алгоритмів (або чіткої логіки), системи з використанням штучного інтелекту мають властивості навчання та самонавчання (тобто накопичення та узагальнення досвіду). Використання штучних нейро-нечітких мереж для моделювання і ідентифікації об'єкта керування – підхід, який зазвичай розглядається як альтернатива методам, заснованим на фізичних або технологічних принципах. Зокрема, це стосується можливості використання нейронних мереж та нечіткої логіки для управління технологічними процесами дроблення-подрібнення та збагачення корисних копалин. В роботі було розглянуто три можливих типи регуляторів, які пропонує середовище MATLAB & Simulink, а саме регулятора з передбаченням NN Predictive Controller, регулятора на основі моделі авторегресії NARMA-L2 та контролера на основі еталонної моделі – Model Reference Controller. Кожен з розглянутих регуляторів може застосовуватись при моделюванні технологічного процесу, але доцільність використання того чи іншого типу, в першу чергу залежить від характеру технологічного процесу. При моделюванні була досліджена можливість керування технологічним процесом за допомогою штучного інтелекту (регуляторів на основі нейронних мереж). Аналіз результатів моделювання трьох типів нейрорегуляторів, показав, що найбільш доцільним при моделюванні керування технологічного процесу подрібнення є застосування регулятора типу NARMA-L2.

Ключові слова: інтелектуальна система управління, моделювання, нейрорегулятор, нейромережа, стадія подрібнення.

USE OF NEURAL REGULATORS IN CONTROL SIMULATION OF CRUSHING STAGE IN CONDITIONS OF MINING AND PROCESSING PLANT

Abstract. The article is devoted to possibility of using standard types of neuroregulators that offer the MATLAB & Simulink environment in modeling control of technological process, namely grinding stage, by applying consistent intelligent control under uncertainty. Use of artificial intelligence technologies in mining is quite relevant at this time. Unlike "classical" deterministic control systems based on rigid algorithms use (or clear logic), systems using artificial intelligence have properties of learning and self-learning (that is, accumulation and generalization of experience). Use of artificial neural networks to model and identify control object is approach that is usually considered as alternative to methods based on physical or technological principles. In particular, this concerns possibility of using neural networks and fuzzy logic to control of crushing-grinding and enrichment of minerals technological processes. Paper considers three possible types of controllers offered by MATLAB & Simulink, namely the NN Predictive Controller, the NARMA-L2 autoregressive control controller and the Model Reference Controller controller. Each of the considered regulators can be applied at modeling of technological process, but expediency of use of this or that type, first of all depends on character of technological process. During the simulation, the possibility of controlling the technological process with the help of artificial intelligence (regulators based on neural networks) was investigated. Analysis of results of modeling three types of neuroregulators showed that the most appropriate for modeling the control of the technological process of grinding is the use of a regulator type NARMA-L2.

Key words: intelligent control system, modeling, neuroregulator, neural network, grinding stage.

Постановка проблеми. Сучасний стан підприємств гірничої галузі характеризується гострою необхідністю вирішення цілого комплексу проблем. В умовах зносу основних засобів виробництва і неможливості їх швидкого відновлення однією з основних проблем є те, що переорієнтація на світові ринки збуту вимагає постійного підвищення конкурентоспроможності продукції, що випускається, зниження її енергоємності.

Відомі різні шляхи кардинального вирішення цих питань: заміна застарілого обладнання, впровадження ефективніших технологій, удосконалення існуючих схем ланцюгів апаратів тощо. Світовий досвід свідчить, що у умовах найефективнішим шляхом підвищення якості та зниження собівартості є комплексна автоматизація основних технологічних процесів, заснована на застосуванні сучасних інтегрованих систем інтелектуального, оптимального і адаптивного керування [3; 8].

Головною проблемою, що виникає під час реалізації таких систем, є відсутність надійних засобів контролю необхідної точності або досить значна вартість окремих датчиків [4]. В той же час, зараз досить активно розвивається альтернативний спосіб побудови автоматизованих систем керування (АСК) на основі використання технологій штучного інтелекту (нейронні мережі, нечітка логіка, генетичні алгоритми тощо). На відміну від «класичних» детермінованих АСК, які засновані на використанні жорстких алгоритмів (або чіткої логіки), системи з використанням штучного інтелекту мають властивості навчання та самонавчання (тобто накопичення та узагальнення досвіду). Також на розробку та впровадження інтелектуальних систем управління витрачаються значно менші кошти за рахунок зменшення необхідності використання дорогого обладнання (датчики, комунікації).

Враховуючи всі зазначені фактори, можна стверджувати, що проблема застосування технологій штучного інтелекту в гірничій справі є відносно новою і дуже актуальною. Зокрема, це стосується можливості використання нейронних мереж та нечіткої логіки для управління технологічними процесами дроблення-подрібнення та збагачення корисних копалин [4].

Аналіз досліджень та постановка задачі. Аналіз робіт показує, що переважна більшість АСУ реалізовано на основі застосування класичних підходів теорії управління, адаптивних та оптимальних систем. Як математичні моделі, як правило, застосовується 1–2 каналні лінійні системи. Більшість таких систем досить просто реалізується за допомогою класичних ПІД-регуляторів. Відносна простота реалізації та порівняно висока надійність таких систем зумовлюють їхнє застосування приблизно у 80–90% систем промислової автоматизації [8]. Водночас відомо, що такі системи не завжди можуть забезпечувати необхідну якість управління, особливо в умовах не стаціонарності, інерційності, запізнення, випадкових обурень, наявності нечіткої та неповної інформації. До того ж, ПІД-регулятори вимагають постійного переналаштування власних коефіцієнтів у разі зміни технологічної ситуації. Тому одним з найбільш перспективних напрямків є застосування нейромережевих технологій в задачах управління. За останні десятиліття даний напрямок отримав значний розвиток [7].

У роботі [2] виконано синтез нейрорегулятора з передбаченням NN Predictive Controller для рішення задачі управління електромеханічною системою з урахуванням пружності механічних зв'язків.

Наведено порядок синтезу нейрорегулятора для заданого об'єкту управління з використанням пакету прикладних програм Neural Network Toolbox системи MATLAB. Виконано моделювання нейромережевої системи та показано, що синтезований нейрорегулятор забезпечує високі динамічні характеристики системи з пружними зв'язками в кінематичних передачах.

Наприклад у роботі [5] Виконано синтез нейрорегулятора на основі моделі авторегресії для рішення задачі управління системою наведення і стабілізації озброєння легкоброньованих машин. Наведено порядок синтезу нейрорегулятора NARMA-L2 Controller для заданого об'єкту управління та виконано моделювання системи на ЕОМ.

Аналізуючи ці та інші роботи [6; 10] можна дійти висновку, що кожен з типів регуляторів з набору блоків Neural Network Blockset краще підходить до відповідних процесів, тому метою дослідження є синтез трьох типів нейрорегуляторів пакету Neural Network Toolbox системи MATLAB для управління стадією подрібнення шляхом застосування узгодженого інтелектуального управління в умовах невизначеності.

Виклад основного матеріалу. Динамічні моделі систем управління з нейромережевими регуляторами розміщені в спеціальному розділі Control Systems набору блоків Neural Network Blockset. У MATLAB реалізовані три нейрорегулятори: регулятор на основі моделі авторегресії зі змінним середнім – NARMA-L2 Controller; регулятор на основі еталонної моделі – Model Reference Controller і регулятора з передбаченням NN Predictive Controller [2].

При подальшому розгляді детальний опис синтезу кожного з наведених регуляторів приводити не будемо, так як він є стандартним і залежить від об'єкту керування, а приділимо увагу доцільності використання регуляторів цих типів для технологічного процесу подрібнення.

Застосування регулятора з передбаченням NN Predictive Controller.

Регулятор з передбаченням NN Predictive Controller використовує модель об'єкта управління у вигляді нейронної мережі для того, щоб передбачати його майбутню поведінку. Алгоритм оптимізації обчислює управляючі сигнали, які мінімізують різницю між бажаними і дійсними змінами сигналу на виході моделі і таким чином оптимізує поведінку об'єкта на заданому інтервалі часу. Побудова моделі об'єкта управління виконується автономно з використанням нейронної мережі, регулятор навчається в груповому режимі з використанням одного з алгоритмів навчання. Регулятор вимагає значного обсягу обчислень, оскільки для розрахунку оптимального закону управління оптимізація виконується на кожному такті управління [1; 9].

Для побудови нейронної мережі для управління технологічним процесом необхідно сформулювати нелінійний регулятор наступного виду [9; 11]:

$$y(k) = \sum_{j=N_1}^{N_2} (y_r(t+j) - y_m(t+j))^2 + p \sum_{j=1}^{N_u} (u'(t+j-1) - u'(t+j-2)) \quad (1)$$

Структурна схема нейроконтролера та результати моделювання управління технологічним процесом представлені відповідно на рис. 1-2.

Функція predort - виконує завдання передбачення майбутнього значення.

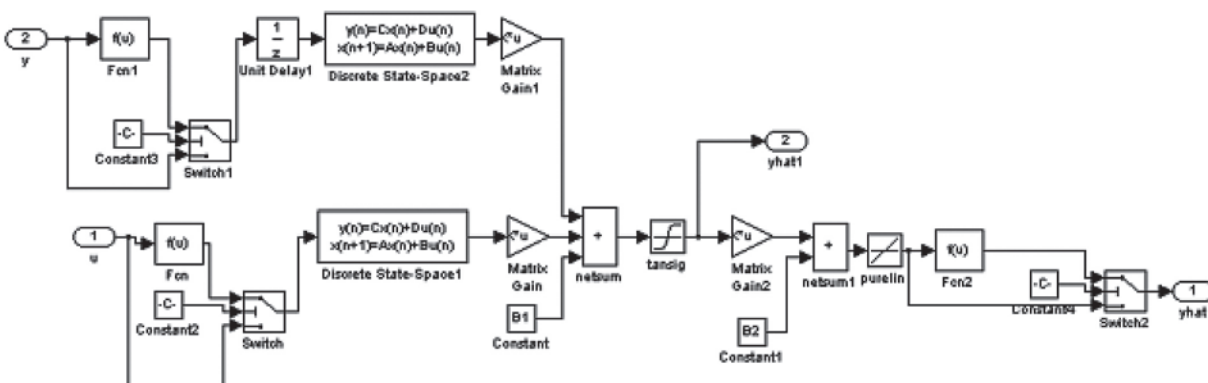


Рис. 1. Структурная схема нейроконтролера NN Predictive Controller

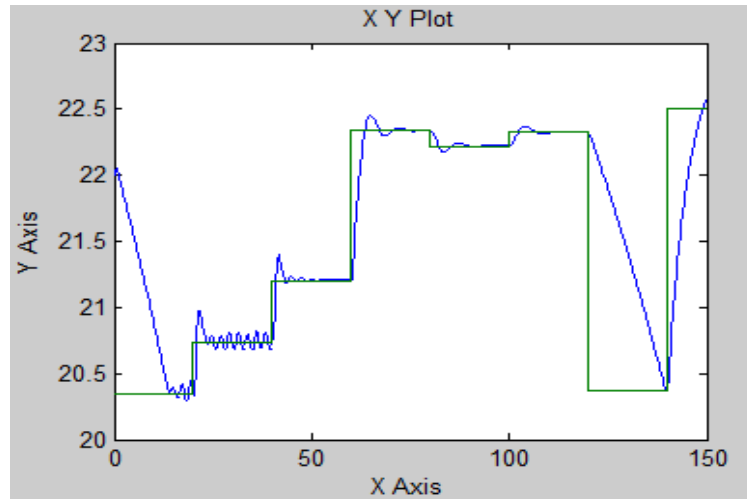


Рис. 2. Результати моделювання технологічного процесу

Застосування регулятора на основі моделі авторегресії NARMA-L2.

З усіх архітектур цей регулятор потребує найменшого обсягу обчислень. Цей регулятор – це деяка реконструкція нейромережевої моделі керованого процесу, отриманої на етапі ідентифікації [9; 11]. Обчислення у час пов'язані лише з реалізацією нейронної мережі. Недолік методу полягає в тому, що модель процесу повинна бути задана в канонічній формі простору стану, що відповідає матриці, що супроводжує, що може призводити до обчислювальних похибок.

Для побудови нейронної мережі для управління технологічним процесом необхідно сформуванати нелінійний регулятор наступного виду:

$$u(k) = G[y(k), y(k - 1), \dots, y(k - n + 1), y_r(k + d), u(k - 1), \dots, u(k - m + 1)] \quad (2)$$

Але хоча такий регулятор за допомогою нейронної мережі може бути побудований, - для цього потрібно провести дуже багато розрахунків, для зменшення середньоквадратичної похибки у зв'язку з використанням методу зворотного поширення помилки. Для практичного вирішення цього завдання, вчені Нарендра та Макхопадхаї запропонували іншу NARMA-модель, що отримала назву NARMA-L2 [9; 11], і має такий вигляд:

$$u(k) = \frac{y_r(k + d) - f[y(k), y(k - 1), \dots, y(k - n + 1), u(k - 1), u(k - 1), \dots, u(k - m + 1)]}{g[y(k), y(k - 1), \dots, y(k - n + 1), u(k - 1), u(k - 1), \dots, u(k - m + 1)]} \quad (3)$$

Але таке використання реалізації регулятора не бажано, оскільки управління $u(k)$ залежить від значення виходу $y(k)$, тому (3) модифікується наступним чином [9,11]:

$$u(k + 1) = \frac{y_r(k + d) - f[y(k), y(k - 1), \dots, y(k - n + 1), u(k - 1), u(k - 1), \dots, u(k - m + 1)]}{g[y(k), y(k - 1), \dots, y(k - n + 1), u(k - 1), u(k - 1), \dots, u(k - m + 1)]} \quad (4)$$

Структурна схема нейроконтролера представлена на рис. 3.

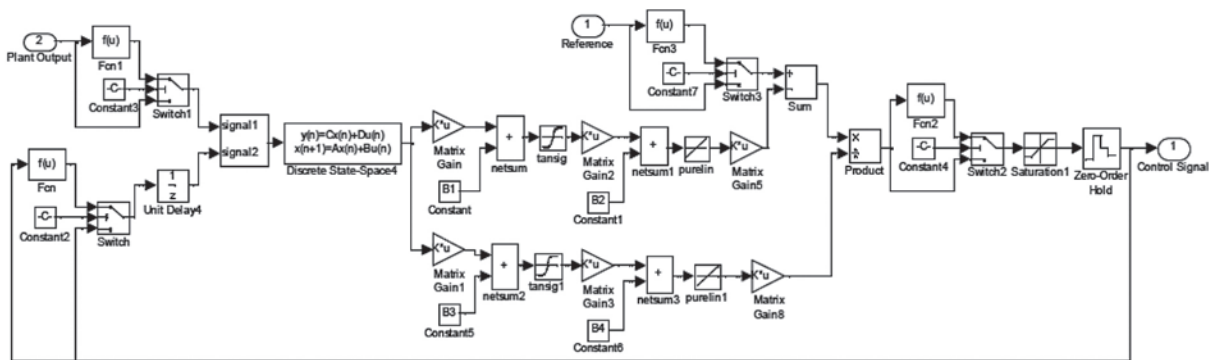


Рис. 3. Структурна схема нейроконтролера NARMA-L2

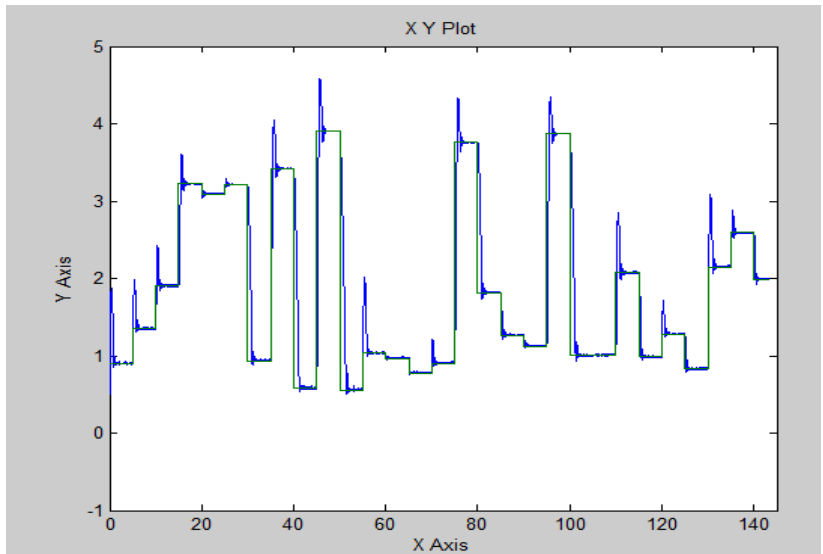


Рис. 4. Результати моделювання технологічного процесу

Застосування регулятора на основі еталонної моделі Model Reference Controller.

Необхідний обсяг обчислень для цього регулятора, порівняно з попереднім, менший. Однак архітектура регулятора з еталонною моделлю потребує навчання нейронної мережі керованого процесу та нейронної мережі регулятора [1; 6]. При цьому навчання регулятора виявляється досить складним, оскільки навчання ґрунтується на динамічному варіанті методу зворотного поширення помилки. Перевагою регуляторів на основі еталонної моделі є те, що вони можуть бути застосовані до різних класів керованих процесів.

При управлінні на основі еталонної моделі регулятор – це нейронна мережа, яка навчена управляти об'єктом так, щоб він відстежував поведінку еталонної моделі. При цьому модель керованого об'єкта активно використовується при налаштуванні параметрів самого регулятора. При побудові нейрорегулятора Model Reference Controller спочатку виконується ідентифікація об'єкта управління, а потім проводиться синтез закону управління. Необхідний обсяг обчислень для цього регулятора можна порівняти з попереднім. Однак архітектура регулятора з еталонною моделлю вимагає навчання нейронної мережі об'єкта управління і нейронної мережі регулятора. При цьому навчання регулятора виявляється достатньо складним, оскільки засноване на динамічному варіанті методу зворотного розповсюдження помилки. Структурна схема нейроконтролера представлена на рис. 5.

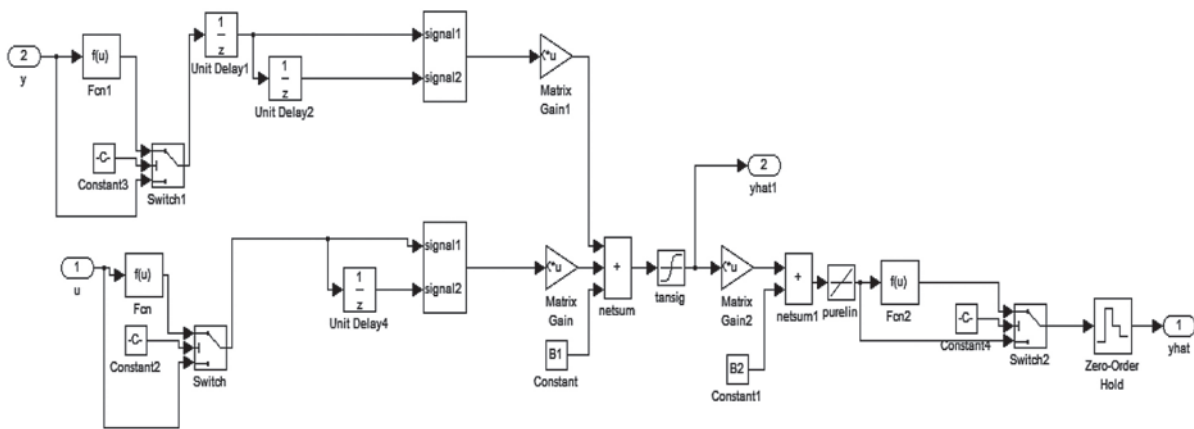


Рис. 5. Структурна схема нейроконтролера Model Reference Controller

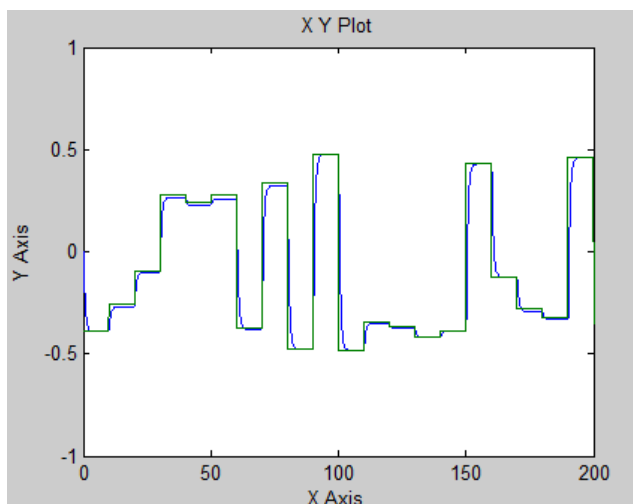


Рис. 6. Результати моделювання технологічного процесу

На перший погляд більш доцільним є використання регулятора на основі моделі авторегресії NARMA-L2. Тому для розглянемо моделювання нейромережевого контролера NARMA-L2 для стадії подрібнення. Як базова модель для отримання навчальної послідовності комплексу прийемо класичну модель в операторній формі. Відповідно до [7; 8] такий об'єкт, що складається з декількох дробарок, розділяючих грохотів і перевантажувальних конвеєрів представляється наступною схемою (рис. 7).

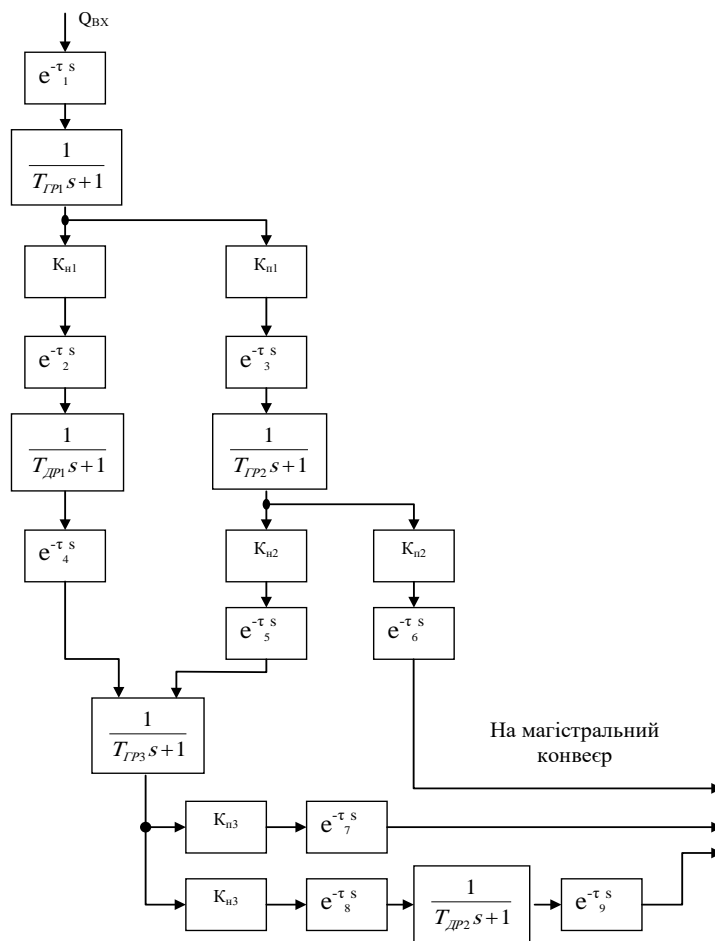


Рис. 7. Структурна схема стадії подрібнення

За допомогою цієї моделі була сформована навчальна вибірка з 1200 спостережень. Діапазон зміни вхідного впливу від 85 т/годину до 115 т/годину.

При синтезі нейрорегулятора NARMA-L2 Controller використовуються такі файли, розміщені в каталозі toolbox/nnet/ncontrol системи SIMULINK: Nncontrolutil – підтримка, що забезпечує можливість звертання до приватних функцій із системи SIMULINK; Sfunxy2 – функція виведення графіків; Nnidet.m – функція, що використовується під час ідентифікації об'єкта управління (ця функція використовується при побудові нейромережевої моделі об'єкта управління при синтезі всіх регуляторів, реалізована в ППП Neural Network Toolbox системи MATLAB) [9; 11].

Проектування нейрорегулятора складається з двох етапів: етап ідентифікації керованого об'єкта та етапу синтезу закону управління. На етапі ідентифікації розробляється модель керованого об'єкта як нейронної мережі, яка на етапі синтезу використовується для синтезу регулятора.

Процес синтезу нейроконтролера починається шляхом активізації блоку NARMA-L2 Controller. Вікно Plant Identification NARMA-L2 універсальне і може бути використане для побудови нейромережевих моделей будь-якого динамічного об'єкта, який описаний моделлю SIMULINK [9; 11].

Навчання нейромережевого NARMA – регулятора, що містить 10 нейронів у прихованому шарі, на тренувальній послідовності проводилося протягом 100 циклів. Система з нейрорегулятором показано на рис. 8, де блок Subsystem містить модель наведену на рис. 7.

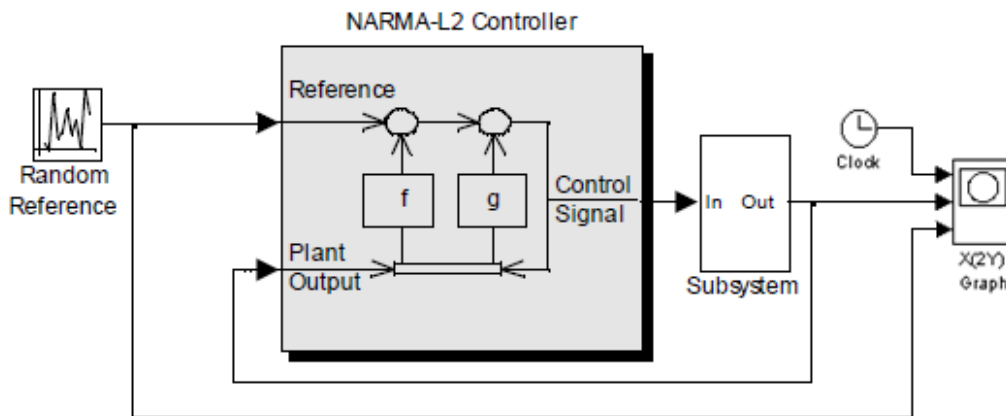


Рис. 8. Система с NARMA-L2 – регулятором

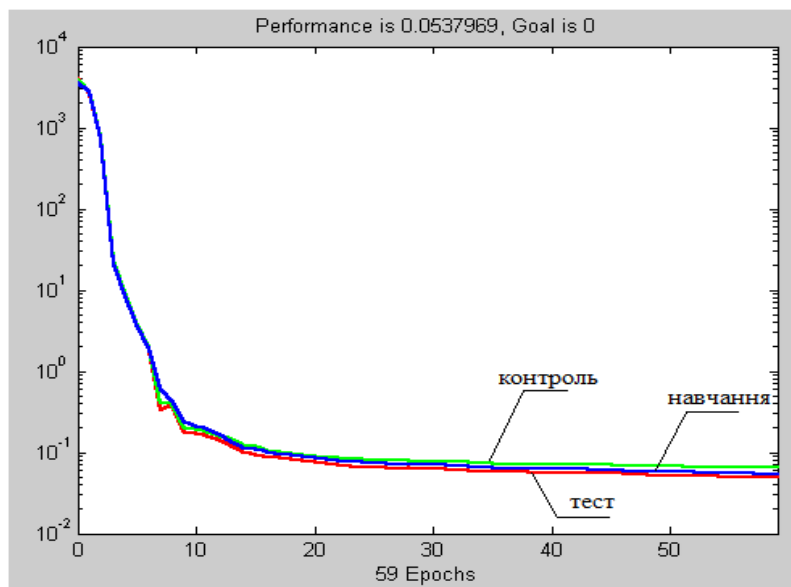


Рис. 9. Помилка навчання, контрольної та тестової послідовності для нейромережевого регулятора

На рис. 9 представлені графіки помилки для навчальної, тестової та контрольної послідовностей.

Результат роботи системи з кваліфікованим регулятором представлений на рис. 10 де крива 1 відображає вхідний вплив; а крива 2 – вихідний сигнал.

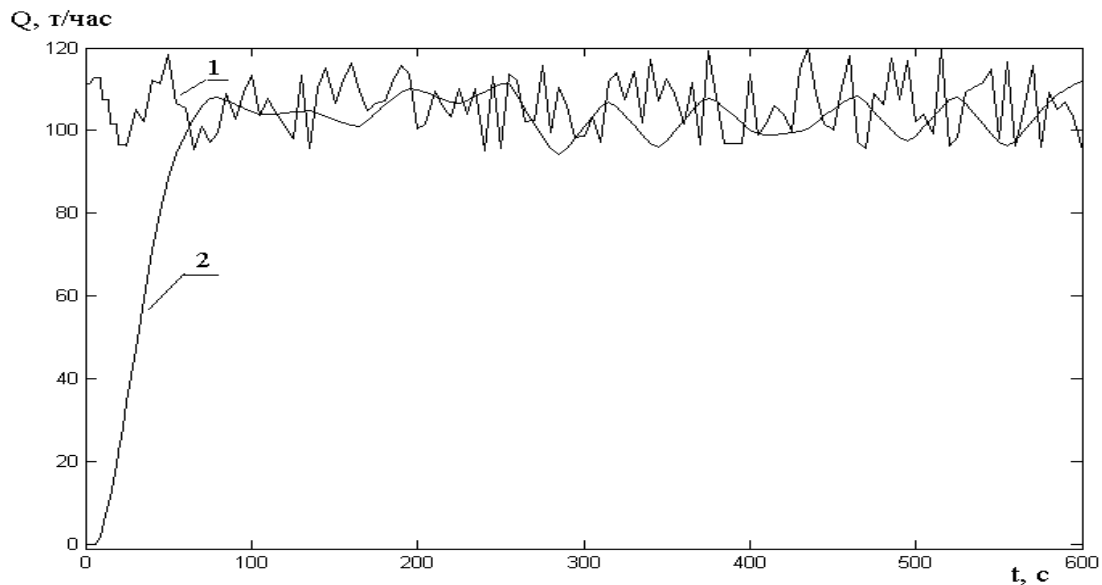


Рис. 10. Вхідний та вихідний сигнали системи з NARMA-L2-регулятором

Порівняння графіків вхідного (випадкового) сигналу та виходу системи показує, що застосування регулятора дозволяє досягти стабільнішої роботи ділянки по вихідному продукту, при випадкових змінах вхідного вантажопотоку.

Висновки. Було розглянуто три типи регуляторів нейрорегуляторів пакету Neural Network Toolbox системи MATLAB, а саме регулятор на основі моделі авторегресії зі змінним середнім – NARMA-L2 Controller; регулятор на основі еталонної моделі – Model Reference Controller і регулятора з передбаченням NN Predictive Controller. Проаналізовано відмінності у їх роботі, побудові та ідентифікації нейронних мереж об'єкта та самого регулятора. На основі результатів моделювання можна стверджувати, що найбільш доцільно для керування технологічним процесом подрібнення використовувати регулятор на основі моделі авторегресії зі змінним середнім – NARMA-L2 Controller.

Список використаних джерел:

1. Astolfi A., Karagiannis D., Ortega R. Nonlinear and adaptive control with applications. Berlin: Springer. 2008. 290 p.
2. Василюк Т. Ю., Варфоломійєв О. О., Іщенко В. С., Ковальчук С. Л. Синтез нейромережевого регулятора для електромеханічної системи з пружними зв'язками в кінематичних передачах. *Системи обробки інформації*. 2018. 2(153). С. 7–17.
3. Гвоздик В. С., Купін А. И. Реализация согласованного управления мельницами измельчения на основе применения нечеткого контролера. *Разраб. рудн. месторожден*. Кривой Рог. 2005. Вып. 88. С. 148–152.
4. Купін А. И. Інтелектуальна ідентифікація та керування в умовах процесів збагачувальної технології: монографія. Кривий Ріг : КТУ. 2008. 204 с.
5. Кузнецов Б. И., Василюк Т. Е., Варфоломеев А. А. Синтез нейросетевого регулятора NARMA-L2 CONTROLLER для системы наведения и стабилизации. *Електротехніка і Електромеханіка*. 2011. № 4. С. 41–46.
6. Кузнецов Б. И., Василюк Т. Ю., Варфоломійєв О. О. Нейромережева система наведення і стабілізації з регулятором на основі еталонної моделі Model Reference Controller. *Електротехніка і електромеханіка*. 2015. №4. С. 35–39.
7. Marynych I. A. Reason for application of intelligent systems for disintegrating complex control. *Metallurgical and Mining Industry*. 2014. No6. P. 25–29.
8. Назаренко В. М., Назаренко М. В., Хоменко С. А., Купін А. И. Современные информационные технологии для управления работой рудником горнообогатительного комбината. *Разраб. рудн. месторожден*. Кривой Рог. 2002. Вып. 77. С. 66–70.
9. Пупков К. А., Егупов Н. Д. Методы робастного, нейро-нечеткого и адаптивного управления. Москва : Изд-во МГТУ им. Н.Э. Баумана. 2001. 744 с.
10. Shuzhi Sam Ge, Chenguang Yang, Tong Heng Lee. Adaptive Predictive Control Using Neural Network for a Class of Pure-Feedback Systems in DiscreteTime. *Neural Networks IEEE Transactions*, Sept. 2008. P. 1599–1614.
11. Терехов В. А., Ефимов Д. В., Тюкин И. Ю. Нейросетевые системы управления. Москва : Высш.шк. 2002. 183 с.

References:

1. Astolfi A., Karagiannis D., Ortega R. (2008). Nonlinear and adaptive control with applications. Berlin: Springer. p. 290. [in English].
2. Vasilec' T.Yu., Varfolomiev O.O., Ishchenko V.S., Koval'chuk S.L. (2018). Sintez nejromerezhevogo reguljatora dlya elektromekhanichnoi sistemi z pruzhnimi zv'yazkami v kinematichnih peredachah [Synthesis of a neural network controller for electro-mechanical system with elastic coupling in kinematic transmissions]. *Sistemi obrobki informacii*. no 2(153). pp. 7–17. doi.30748/soi.2018.153.01 [in Ukrainian].
3. Gvozdk V.S., Kupin A.I. (2005). Realizacija soglasovannogo upravleniya mel'nicami izmel'cheniya na osnove primeneniya nechetkogo kontrolera. [Implementation of grinding mills coordinated control based on fuzzy controller use]. *Razrab. rudn. mestorozhden. Krivoj Rog*. vol. 88. pp. 148–152. [in Russian].
4. Kupin A.I. (2008). Intelektual'na identifikaciya ta keruvannya v umovah procesiv zbagachuval'noi tekhnologii [Intellectual identification and identification in minds of processes of smart technology]: monografiya. Krivij Rig: KTU. 204 p. [in Ukrainian].
5. Kuznecov B.I., Vasilec T.E., Varfolomeev A.A. (2011). Sintez nejrosetevogo reguljatora NARMA-L2 CONTROLLER dlya sistemy navedeniya i stabilizacii. [NARMA-L2 controller synthesis for a guidance and stabilization system]. *Elektrotehnika i Elektromekhanika*. no 4. pp. 41–46. [in Russian].
6. Kuznecov B.I., Vasilec' T.Yu., Varfolomiev O.O. (2015). Nejromerezheva sistema navedennya i stabilizacii z reguljatorom na osnovi etalonnoi modeli Model Reference Controller [Neuromede guidance and stabilization system with regulator based on Model Reference Controller]. *Elektrotehnika i Elektromekhanika*. no. 4. pp. 35–39. [in Ukrainian].
7. Marynych I.A. (2014). Reason for application of intelligent systems for disintegrating complex control. *Metallurgical and Mining Industry*. no 6. pp. 25–29. [in English].
8. Nazarenko V.M., Nazarenko M.V., Homenko S.A., Kupin A.I. (2002). Sovremennye informacionnye tekhnologii dlya upravleniya rabotoj rudnikom gornoobogatitel'nogo kombinata [Modern information technologies for managing work of mine of mining and processing plant]. *Razrab. rudn. mestorozhden. Krivoj Rog*. vol. 77. pp. 66–70. [in Russian].
9. Pupkov K.A., Egupov N.D. (2001). Metody robnstnogo, nejro-nechetkogo i adaptivnogo upravleniya [Robust, neuro-fuzzy and adaptive control methods]. Moscow : Izd-vo MGTU im. N.E. Baumana. 744 p. [in Russian].
10. Shuzhi Sam Ge, Chenguang Yang, Tong Heng Lee. (2008). Adaptive Predictive Control Using Neural Network for a Class of Pure-Feedback Systems in Discrete Time. *Neural Networks IEEE Transactions*. pp. 1599–1614. [in English].
11. Terekhov V. A., Efimov D. V., Tyukin I. YU. (2002). Nejrosetevye sistemy upravleniya [Neural network control systems]. Moscow : Vyssh.shk.. 183 p. [in Russian].

УДК 004.94

DOI <https://doi.org/10.32689/maup.it.2022.1.7>

Марина НОС

аспірант, Національний аерокосмічний університет імені М. Є. Жуковського «Харківський авіаційний інститут», вул. Чкалова, 17, Харків, Україна, індекс 61000 (marina.nos1701@gmail.com)

ORCID: 0000-0001-8407-440X

Marina NOS

Postgraduate Student, National Aerospace University "Kharkiv Aviation Institute", 17 Chkalova str., Kharkiv, Ukraine, postal code 61000 (marina.nos1701@gmail.com)

Бібліографічний опис статті: Нос М. Огляд сучасних методологій управління вартістю ІТ-проектів. *Інформаційні технології та суспільство*. 2022. Вип. 1 (3). С. 54–60. DOI: <https://doi.org/10.32689/maup.it.2022.1.7>

Bibliographic description of the article: Nos, M. (2022). Ohliad suchasnykh metodolohii upravlinnia vartistiu IT-proiektiv [Overview of modern methodologies of It project management]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, 1 (3), 54–60. DOI: <https://doi.org/10.32689/maup.it.2022.1.7>

ОГЛЯД СУЧАСНИХ МЕТОДОЛОГІЙ УПРАВЛІННЯ ВАРТІСТЮ ІТ-ПРОЄКТІВ

Анотація. Предметом дослідження даної статті є методології управління ІТ-проектів. **Мета роботи** – проведення огляду сучасних механізмів управління ІТ-проектів задля з'ясування підходу до оптимізації (раціоналізації) процесу оцінювання вартості ІТ-проекту. В статті вирішуються такі **завдання**: аналіз стандартів, концепцій та підходів до управління ІТ-проектами, розгляд критеріїв вибору підходящої методології для того чи іншого проекту. Використовуються такі методології: Waterfall (каскадна модель), Agile, Scrum, Kanban, Гібридна та Спіральна методологія. Отримано наступні результати: Визначено, що методології управління ІТ-проектами можна розділити на традиційні (каскадні) та гнучкі, а всі інші — їх комбінації. Розкрито сутність Waterfall методології, Agile, Scrum, Kanban, Гібридної та Спіральної моделі. Позначено, для яких ІТ-проектів більш за все підходить кожна з них. Проведено аналіз національних стандартів в області управління ІТ-проектами. Визначено, що відповідно до діючого Державного стандарту «Інформаційна технологія» створення ІТ-проекту як автоматизованої системи зводиться до восьми послідовних стадій: формування вимог, розробка концепції, технічне завдання, розробка ескізного проекту, технічний проект, розробка робочої документації, введення в дію, супровід. Цей Держстандарт, як інші, вже застарів та не охоплює всіх аспектів управління проектами, як от, наприклад, вартість. Досліджено міжнародні етапони управління ІТ-проектами: IPMA (Individual Competence Baseline), Prince 2 (Project IN Controlled Environments 2), P2M (Project and Program Management for Enterprise Innovation), керівництво до зводу знань з управління проектами (PMBOK), керівництво до зводу знань з інженерії програмного забезпечення (SWEBOK), керівництво до збору знань з бізнес-аналізу (BABOK). Проілюстровано взаємодію керівництв збору знань в формуванні методології управління вартістю ІТ-проектів та визначено, що саме на перетині керівництв до збору знань з управління проектами, інженерії програмного забезпечення та бізнес-аналізу формується методологія управління вартістю ІТ-проектів. В межах огляду стандартів управління ІТ-проектами, виділено, що для корпоративного управління ІТ-проектами можуть бути обрані окремі стандарти через складність узгодження їх між собою. Висновки: За результатами проведеного дослідження можна зробити **висновок**, що методології управління ІТ-проектів є незамінними інструментами для реалізації успішних ІТ-проектів. Вибір залежить від особливостей проекту та його команди і будь-яка методологія має бути адаптована під конкретний проект – універсальних підходів не існує.

Ключові слова: ІТ-проекти, методології, управління проектами, стандарт, автоматизовані системи.

OVERVIEW OF MODERN METHODOLOGIES OF IT PROJECT MANAGEMENT

Abstract. The subject of this article is the methodology of IT project management. The purpose of the work is to review modern mechanisms of IT project management in order to clarify the approach to optimization (rationalization) of the process of estimating the cost of IT project. The article solves the following tasks: analysis of standards, concepts and approaches to IT project management appropriate methodology for a project. The following methodologies are used: Waterfall (cascade model), Agile, Scrum, Kanban, Hybrid and Spiral methodology. The following results were obtained: It is determined that the methodologies of IT project management can be divided into traditional (cascading) and flexible, and all others – their combinations. The essence of Waterfall methodology, Agile, Scrum, Kanban, Hybrid and Spiral models is revealed. It is indicated for which IT projects each of them is most suitable. An analysis of national standards in the field of IT project management. It is determined that according to the current State Standard "Information Technology" the creation of IT project as an automated system is reduced to eight successive stages: requirements, concept development, terms of reference, draft design, technical design, development of working documentation, implementation, maintenance. This State Standard, like others, is outdated and does not cover all aspects of project management, such as cost. International standards of IT project management have been studied: IPMA (Individual Competence Baseline), Prince 2 (Project IN Controlled Environments 2), P2M (Project and Program Management for Enterprise Innovation), Project Management Knowledge Management Guide (PMBOK), Management Guide Knowledge

in Software Engineering (SWEBOK), Business Analysis Knowledge Collection Guide (BABOK). The interaction of knowledge gathering management in the formation of IT project cost management methodology is illustrated and it is determined that it is at the intersection of management to knowledge collection on project management, software engineering and business analysis that the methodology of IT project cost management is formed. As part of the review of IT project management standards, it is highlighted that separate standards can be selected for corporate IT project management due to the difficulty of reconciling them. Conclusions: Based on the results of the study, it can be concluded that IT project management methodologies are indispensable tools for the implementation of successful IT projects. The choice depends on the characteristics of the project and its team and any methodology must be adapted to the specific project – there are no universal approaches.

Key words: IT projects, methodologies, project management, standard, automated systems.

У минулому багато IT-компаній не усвідомлювали важливості управління проектами і навіть не мали таких спеціалістів, як менеджери проєктів. В основному, був лише розробник або команда розробників і клієнт, які безпосередньо спілкувалися один з одним, обговорювали вимоги та робочий процес тощо.

Така організація роботи мала багато недоліків, тому що розробникам доводилося мати справу з великим обсягом зайвих процесів замість того, щоб виконувати свою безпосередню роботу – кодування. В результаті розробники зіткнулися з перевантаженістю, низькою продуктивністю та простроченими термінами.

Тому багато компаній з розробки програмного забезпечення підготували кілька підходів до управління IT-проєктами, щоб замінити модель «розробник-клієнт».

Незважаючи на те, що сьогодні ситуація інша, клієнти компаній-розробників програмного забезпечення не знають, які підходи до управління процесом розробки підходять для їх конкретного випадку. Отже, особливо актуальною стає інформація і знання про методології управління IT-проєктами, їх вартість та коли використовувати той чи інший підхід до управління проєктом.

Аналіз останніх досліджень і публікацій показав, що на сьогодні існує велика кількість наукових доробок вітчизняних і зарубіжних авторів, що присвячено розробці і удосконаленню методологій управління IT-проєктами. Але треба виділити наступних: А. Катренко, Н. Бойко, М. Чайковська, О. Псарьов, Л. Ременяк, А. Сидорова, Л. Піддубна, У. Мельник, Н. Казакова, В. Тіторенко, А. Добровольський, А. Яковенко, Н. Шашкова, Д. Коноваленко, М. Чубенко, М. Беляков. Всі роботи зазначених авторів присвячено розробці окремих елементів чи вирішенню окремих питань вдосконалення методології управління IT-проєктами, але враховуючи постійний розвиток і ускладнення сучасних задач, що вивіщуються всіма учасниками процесу розробки і провадження IT-проєктів, на думку автора, потребує подальшого розвитку питання пов'язані з оптимізацією ресурсів, що задіяні в IT-проєктах.

Виділення невирішених раніше частин загальної проблеми показує, що не зважаючи на значну кількість праць, на сьогоднішній день питання вибору методології для ефективного управління IT-проєктами залишається важливим та до кінця невирішеним. Отже, визначення та підбір методології і її окремих аспектів, які можна застосувати на практиці залишається актуальною задачею.

Мета роботи – розгляд періодизації розвитку методології управління IT-проєктами, а також виділення позитивних і негативних сторін кожної з методологій і формування подальших задач для удосконалення зазначених методологій.

В практиці управління IT-проєктами існує значна кількість методологій управління ними. Всі вони складаються з певного набору інструментів і процесів, що дозволяє використовувати їх до певних проєктів розробки специфічного програмного забезпечення, але залишається відкритим питання щодо їх стандартизації і чіткої регламентації вибору під особливі потреби проєктування інформаційних систем, їх модернізації, чи створення інноваційних IT-продуктів. За даними дослідження [1], більшість IT-проєктів (85%), що не мають корпоративних стандартів управління не досягають поставлених цілей, а 32% з них достроково закриваються, не досягнувши поставленої кінцевої мети.

Методологія управління IT-проєктом є дорожньою картою розробки і реалізації IT-проєкту, яка надає проєктній команді набір інструментів і регламентованих процесів для ефективною і якісною роботи над IT-проєктом. Завдяки вчасно обраній методології проєктна команда може ефективно використовувати час для реалізації проєкту, пришвидшити виконання проєкту і стандартизувати вихідні результати.

Існує значна кількість різних методологій управління і розробки IT-проєктів, які використовуються для реалізації різних IT-проєктів для різних галузей національного господарства, вони відрізняються різним набором інструментів і різними підходами до управління IT-проєктами. Однак вибір методології управління IT-проєктом є складною задачею. Для цього необхідно усвідомлювати сутність кожної з них.

На думку автора, під методологією управління IT-проєктами будемо розуміти структуру організації роботи, цілі, задачі, функціональні блоки, ключові етапи, методи і підходи до управління IT-проєктами із зазначенням метрик і показників ефективності результатів роботи.

В практиці методології управління IT-проєктами можна розділити на дві великі групи: традиційні (каскадні) і гнучкі, всі інші методології є комбінацією цих двох з додаванням окремих прийомів

і процесів. Каскадна методологія характеризується чіткими прийомами, запланованим, запротокольованим і деталізованим підходом до управління IT-проєктом, в якому всі учасники повинні строго дотримуватися плану проєкту. Гнучкі методології є протилежністю традиційним і дають змогу змінювати, повертатися і корегувати етапи розробки, а також встановлювати пріоритетність етапів і процесів IT-проєкту.

Waterfall (каскадна модель) [2] – це послідовна методологія управління IT-проєктом, що складається з п'яти етапів: аналіз вимог, проєктування, розробка, тестування і технічна підтримка продукту. Перехід між етапами суворо регламентований, тільки при завершенні попереднього етапу можливий перехід до наступного. Повернення і виправлення у попередніх етапах заборонено. В каскадній моделі значну увагу приділяють документуванню результатів кожного з етапів. Перехід до наступних етапів без задокументованих етапів не можливий. При виборі даної методології важливою умовою є чітке бачення кінцевого результату IT-проєкту. Каскадна методологія має чітку структуру, її легко використовувати завдяки чітко сформованим строкам і кінцевим цілям. Каскадна модель підходить для середніх IT-проєктів метою яких є впровадження готового рішення, або його доопрацювання, з великим бюджетом проєкту, з проєктною командою від 10 і більше учасників, де ролі участі у проєкті чітко визначені.

Agile методологія – це протилежна каскадній моделі методологія, створена для управління складними IT-проєктами. Ця методологія ґрунтується на гнучкому управлінні і більш швидкоокоому реагуванні на виклики. Головними принципами Agile є наступні:

- здатність адаптуватися до змін вимог;
- якісна комунікація між членами команди;
- можливість впливу на кінцевий результат кожного учасника проєкту;
- постійна взаємодія з замовником програмного продукту.

Agile має множину технік, процесів і прийомів, що дозволяє команді проєкту адаптуватися до динамічних змін. Характерною рисою Agile методології є менша кількість вимог, етапи в методології ітеративні з незначним коригуванням, які вносяться у відповідності до зміни вимог. Agile методологія не регламентує чіткого завершення проєкту, а відповідно немає кінцевої вартості IT-проєкту. Ця методологія може бути використана для управління проєктами з малим і середнім бюджетами, до складу проєктної команди якого входить від п'яти до десяти членів. Ролі членів команди повинні бути чітко регламентовані. Більш за все Agile методологія підходить для розробки інноваційного програмного продукту, впровадження готового рішення, чи його доопрацювання [3].

Гнучкі методології дозволяють вдосконалювати IT-проєкт на основі внесення позитивних змін до програмного продукту. Головну увагу при гнучкому управлінні приділяють не формальному виконанню етапів, а якісному кінцевому результату. В основу гнучких методологій покладено наступні принципи:

- увага приділяється взаємодії учасників проєкту, а не процесам і інструментам;
- готовність команди проєкту і розробників до змін, а не дотримання початкового плану;
- постійна взаємодія з замовником;
- ухилення від створення вичерпної документації.

Гібридна методологія (структурована Agile методологія) – це методологія, що поєднує у собі комбінації Agile і Waterfall методології, звідси і виникає її назва. Ця методологія ґрунтується на кращих практиках двох методологій і представляє собою гнучкий, але структурований підхід. На першому етапі проводиться збір і обробка вимог до IT-проєкту, після чого проводяться гнучкі і швидкі ітерації. Ця методологія може бути використана для складних IT-проєктів з середнім і великим бюджетом, з командою проєкту більше 100 учасників, ролі яких чітко визначені. Гібридна методологія більш за все підходить для IT-проєктів, що представляють собою впровадження готового рішення чи його доробку [3].

Scrum методологія – представляє собою гібридну методологію, в якій акцент робиться на роботу проєктної команди (зустрічі для обговорення результатів і коригування дій). В основу методології Scrum покладені підходи Agile методології, але в цій методології присутні особливі підходи, методи і тактики до управління IT-проєктами. Scrum є найбільш популярною методологією розробки програмних продуктів, одними з головних переваг цієї методології є системний підхід до оцінювання завдань IT-проєкту, ефективному розподілі ролей учасників проєкту, організації зустрічей, мотивації учасників, а також візуалізації процесу розробки програмного продукту. Основним завданням Scrum методології є самоорганізація і самокерованість проєктної команди, тобто ця методологія підходить до команд зі значним досвідом і високою кваліфікацією її учасників [4]. Scrum методологія – може бути запропонована для нескладних IT-проєктів з маленьким та середнім бюджетами, де у складі проєктної команди є від п'яти до десяти членів з чітко визначеними ролями. Використовується ця методологія для розробки нових програмних продуктів, впровадження готових рішень і їх доробку і модернізація. Для впровадження Scrum методології основними вимогами є створення позитивного комунікаційного клімату в компанії-виконавця, готовність керівництва і колективу компанії до впровадження змін, розуміння

команд проектів принципів гнучких методологій, здатність створювати дієвого плану дій у відповідності до змін проекту, а також підготовка команди проекту до цих змін.

Kanban методологія – це методологія управління IT-проектом, що ґрунтується на Agile методології. Специфікою Kanban методології є прозорість процесів управління проектом, а також спрощення процесів обміну інформацією між учасниками проекту, це дозволяє більш швидко розробляти, впроваджувати і реалізовувати IT-проекти. Ця методологія використовує так звані дошки Kanban, яка є інструментом контролю виконання робіт за проектом. Дошка включає наступні питання: що потрібно зробити, які роботи в процесі реалізації, які роботи проходять перевірку і що зроблено. Ця дошка дозволяє проектній команді швидко орієнтуватися в задачах і процесах IT-проекту і розробляти тактичні дії щодо успішного завершення проекту. Основними принципами Kanban є прозорість і своєчасність термінів виконання робіт за проектом і рівномірний розподіл навантаження між його учасниками. Також методологія Kanban дозволяє гнучко вирішувати поточні задачі проекту, без чіткого розділу на ролі, що в свою чергу дає змогу вільного переходу окремих виконавців проекту до інших складних задач. Ця методологія підходить для управління нескладних і середньої складності IT-проектів з малими і середніми бюджетами. Склад проектної команди може складатися від п'яти до десяти членів, де немає чіткого розділення на ролі. Найбільш краще підходить для IT-проектів, що спрямовані на інноваційну розробку IT-продукту, чи доробку вже існуючого [5].

Спіральна методологія – це методологія управління IT-проектами, що ґрунтується на каскадній і ітераційній моделях. Специфікою даної методології є поглиблений підхід до управління ризиками. Кожний виток спіралі відповідає частині програмного забезпечення, що розробляється. Якість вихідного продукту досягається за рахунок уточнення цілей і характеристик IT-проекту на кожному витку спіралі методології. Саме поглиблення і конкретизація задач, робіт, процесів і деталей IT-проекту дає змогу обґрунтовано обрати один з найефективніших варіантів, який потім впроваджується у життя. Кожному витку повинні відповідати наступні його частини: цілі; оцінювання ризиків і їх шляхи зменшення їх впливу; розробка і тестування; планування наступного витку спіралі [6]. Спіральна методологія може бути обрана для реалізації легких і складних IT-проектів з середнім і великим бюджетами. Склад проектної команди в цій методології може бути від десяти до ста і більше учасників з чітким визначенням їх ролей. Ця методологія підходить для будь-яких IT-проектів (розробки нового продукту чи доробка вже існуючого).

Вибір методології управління IT-проектами спирається на великій кількості критеріїв, але сьогодні все більшу популярність набирають гнучкі методології, це в першу чергу пов'язано з динамічністю, складністю і невизначеністю зовнішнього середовища. Гнучкий підхід дозволяє проводити удосконалювати код з кожною інтеграцією, що дає змогу мінімізувати кількість помилок. Гнучкі методології більш орієнтовані на клієнта на основі постійної взаємодії з ним. Крім того гнучкі методології виходять за рамки використання IT-проектів і починають впроваджуватися в інших галузях.

Треба зазначити, що постійний розвиток і модернізація інформаційних технологій й програмного забезпечення призводить до удосконалення існуючих методологій управління IT-проектами. Найактуальнішими викликами сьогодні є: необхідність швидкісної розробки програмних продуктів; постійна модернізація програмного забезпечення; вимоги до якості програмного забезпечення, що постійно зростають; збільшення конкурентного тиску в зв'язку зі збільшенням компаній-розробників програмного забезпечення.

Створення IT-проекту може бути представлено як розробка автоматизованої системи. Відповідно до діючого «ДСТУ 34.601-90 Інформаційна технологія. Комплекс стандартів на автоматизовані системи. Автоматизовані системи» [7] створення автоматизованої системи має вісім послідовних стадій:

1. Формування вимог до автоматизованої системи (обстеження об'єкту, формування вимог користувача, оформлення звіту і заявки про розробку автоматизованої системи);
2. Розробка концепції автоматизованої системи (проведення необхідних науково-дослідних робіт, розробка варіантів концепції автоматизованої системи);
3. Технічне завдання (розробка технічного завдання та його затвердження);
4. Розробку ескізного проекту (розробку окремих частин ескізного проекту, розробка документації автоматизованої системи і її частин);
5. Технічний проект (розробка проектних рішень, розробку документації, розробка завдань на проектування в суміжних частинах проекту);
6. Розробка робочої документації (розробка робочої документації, розробка та адаптація програм);
7. Введення в дію автоматизованої системи (підготовка об'єкту автоматизації в дію, підготовка персоналу, комплектація автоматизованої системи, пускові роботи, проведення випробувань, експлуатація, проведення приймальних випробувань);

8. Супровід автоматизованої системи (проведення робіт у відповідності з гарантійними зобов'язаннями, післягарантійне обслуговування).

На думку автора, цей Державний стандарт є застарілим, був прийнятий ще в 1990 році. Він не повністю відображає сучасну специфіку створення ІТ-продукту і управління ІТ-проектами.

Також треба звернути увагу на Національний стандарт «Інженерія систем і програмного забезпечення. Процеси життєвого циклу програмного забезпечення» (ДСТУ ISO/IEC 12207:2016) [8]. Цей стандарт створений для встановлення загальної структури процесів життєвого циклу програмного забезпечення, визначення термінології. Стандарт чітко визначає процеси придбання, постачання, розробки, експлуатації, супроводу та припинення використання програмного продукту. В цьому стандарті є можливість вибору, структурувати й застосовувати елементи середовища для розробки і створення програмних продуктів та послуг, а саме визначає організацію і сторони проекту, процеси організації забезпечення проекту (процес управління моделями життєвого циклу, процеси управління портфелем проектів, людськими ресурсами, якістю), процеси проекту (процеси планування, управління і оцінки проекту, прийняття управлінських рішень, ризиками, конфігурацією і інформацією). Як видно, в цьому стандарті цей діючий стандарт не охоплює питання управління вартістю ІТ-проектів.

Ще одним Національним стандартом, що регулює питання управління проектами розробки програмних продуктів і послуг є «Розробка систем та програмного забезпечення. Процеси життєвого циклу керування проектами» (ISO/IEC/IEEE 16326:2009) [9]. Цей стандарт охоплює нормативні настанови щодо управління проектами, які охоплюють проекти створення програмних продуктів, а також проектів програмних систем. Основними користувачами цього стандарту є керівники проектів. Стандарт покликаний допомогти керівникам успішно реалізувати проекти з розробки програмних систем і програмних продуктів. Цей стандарт складається з розділів «Елементи плану керування проектом» і «Процеси проекту». До розділу елементів плану керування проектом віднесені наступні:

- контекст проекту;
- планування проекту;
- оцінювання та контроль проекту;
- постачання продуктів;
- плани підтримки процесу;
- додаткові плани.

Розділ «Процеси проекту», дублюють відповідний розділ стандарту ДСТУ ISO/IEC 12207:2016. Як видно з цього стандарту, він теж не розкриває процеси управління вартістю проектів зі створення програмних засобів і систем.

Стандарти управління ІТ-проектами можна розділити за сферами застосування:

- загальні питання процесів управління проектами (PMBOK);
- стандарти які застосовуються до суб'єктів управління (IPMA, PMBOK, Prince 2);
- стандарти, що оцінюють зрілість організації до управління проектами (P2M);
- спеціалізовані стандарти з управління ІТ-проектами (SWEBOK, BABOK).

Ґрунтуючись на зарубіжному досвіді основними стандартами управління ІТ-проектами є наступні настанови:

– IPMA (Individual Competence Baseline) – це стандарт, що ґрунтується на компетентнісному підході, вимог до спеціалістів з управління проектами. Розробником цього стандарту є Міжнародна асоціація з управління проектами. Цей стандарт є системою знань для проведення міжнародної сертифікації спеціалістів з управління проектами. Також цей стандарт послужив основою для розробки національних стандартів членів асоціації. В основі стандарту лежать вимоги до компетенцій проектних менеджерів: технічні (процеси управління проектами, програмами і портфелями за функціональними сферами), соціальні (відображають особисті якості і навички управління проектами, управління командою проекту, комунікацією) і контекстуальні (відображають компетенції, що відповідають за опис елементів зовнішнього середовища) [10].

– Prince 2 (PRoject IN Controlled Environments 2) – це одна з найбільш популярних стандартів управління проектами. Цей стандарт був розроблений у Великобританії як система управління, контролю і організації проектами. Стандарт ґрунтується на наступних принципах [11]: оцінювання доцільності проекту, управління проектами на основі позитивного досвіду, чіткий розподіл сфер відповідальності, поетапне управління, оцінювання відхилень і управління за ними, орієнтація на результат, адаптивність до змін і особливостям проекту. До компоненту управління проектами входять наступні: початок проекту, ініціація, управління командою, контроль, управління поставками, управління межами і завершення проекту. До компоненту управління проектами треба віднести: обґрунтування проекту, організація і планування, управління ризиками і якістю, а також управління змінами.

– P2M (Project and Program Management for Enterprise Innovation) – це система стандартів з управління проектами, яка розроблялася Японською асоціацією інжинірингу і Асоціацією проектних менеджерів Японії. Цей збір знань ґрунтується на кращому досвіді японських підприємств зі значною доданою вартістю і інноваційними проектами. Цей стандарт робить акцент на інноваційному розвитку підприємства і створення додаткової цінності. Стандарт [12] складається з розділів управління проектами, управління програмами і області знань управління проектами. Управління проектами охоплює понятійний апарат управління проектами, середовища управління проектами та знань і навичок з управління проектами. До областей знань у стандарті віднесені наступні: управління стратегією проекту; управління фінансами; управління системами проекту; організаційне управління; управління цілями; управління ресурсами; управління ризиками; управління інформаційними технологіями, управління відносинами, управління вартістю і комунікаціями.

– керівництво до зводу знань з управління проектами (РМВОК) [13] – це довідник, що містить набір регламентованих процесів, фундаментальні і базові практики з управління проектами в будь якій галузі;

– керівництво до зводу знань з інженерії програмного забезпечення (SWEBOK) [14] – це довідник, що містить знання з інженерії програмного забезпечення. Довідник містить десять галузей знань з програмної інженерії: вимоги до програмного забезпечення; проектування, консультування, тестування, супровід, керування конфігурацією, керування проектами, процеси, засоби і інструменти програмного забезпечення, а також його якість;

– керівництво до збору знань з бізнес-аналізу (BABOK) [15] – це довідник, що містить збір знань з практик бізнес-аналізу. Це керівництво дає опис галузей знань з бізнес-аналітики, її методики, компетенції і перспективи. До основних розділів треба віднести: планування і моніторинг бізнес-аналітики, виявлення та взаємодію з замовником, управління вимогами, аналіз стратегій, оцінювання та визначення якості рішень, базові компетенції і методи бізнес-аналізу.

На погляд авторки саме на перетині керівництв до збору знань з управління проектами, інженерії програмного забезпечення та бізнес-аналізу формується методологія управління вартістю ІТ-проектів (рис. 1).

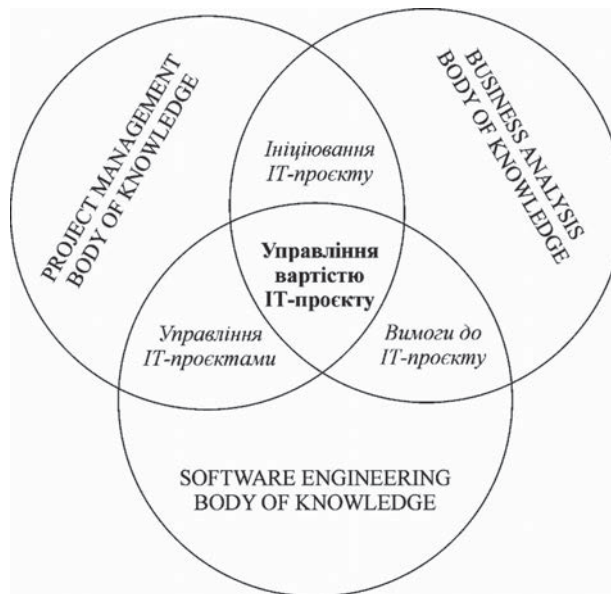


Рис. 1. Взаємодія керівництв збору знань в формуванні методології управління вартістю ІТ-проектів

Відповідно до проведеного огляду стандартів управління ІТ-проектами, треба зазначити, що для корпоративного управління ІТ-проектами можуть бути обрані окремі стандарти, у зв'язку з тим що їх складно узгоджувати між собою. Вибір стандарту, чи їх комбінації (окремих розділів) залежить від специфіки управління, корпоративної культури, сегменту ринку на якому функціонує підприємство, кон'юнктури світового і регіонального ринків, а також відповідати напрацьованій практиці в сфері проектного менеджменту на підприємстві.

Висновки. Методології управління ІТ-проектами є незамінними інструментами, які допоможуть у реалізації успішних проектів із високою цінністю. І хоча кожен підхід дає багато переваг, слід зазначити, що методології не є універсальними.

Для вибору методології управління ІТ-проектами треба використовувати адаптивний підхід, тому що кожна з методологій може бути використана для окремих видів ІТ-проектів, а також різних видів проектних команд. Вибір найбільш підходящої методології може бути складним. Це залежить від великої кількості факторів, багато з яких поза межами нашого контролю, і, можливо, було б розумно зосередитись на прагматичному підході до методологій управління.

Список використаних джерел:

1. Скоробагата Ю. Р., Турченко І. В., Нгой К. Ж., Горбач В. В. Основні проблемні аспекти в управлінні ІТ-проектами. *ISCIENCE.IN.UA «Актуальні наукові дослідження в сучасному світі»*. 2019. № 11 (55). 1. С. 183–186.
2. SDLC – Waterfall Model. URL: <http://www.tutorialspoint.com/sdlc>
3. The Definitive Guide to Project Management Methodologies. URL: <http://www.workamajig.com/blog/project-management-methodologies>
4. What is Scrum Methodology? URL: <http://resources.collab.net/agile-101/what-is-scrum>
5. Книберг Х., Скарин М. Scrum и Kanban: выжимаем максимум. URL: <http://scrum.org.ua/wp-content>
6. Тюлькина А. С., Чжан Ю. Сравнительный анализ каскадной и спиральной методологии управления проектами разработки программного обеспечения. URL: <http://kmu.itmo.ru>
7. ГОСТ 34.601-90 Автоматизированные системы. Стадии создания. Дата введения 01.01.1992 г.
8. ДСТУ ISO/IEC 12207:2016 Інженерія систем і програмного забезпечення. Процеси життєвого циклу програмного забезпечення (ISO/IEC 12207:2008, IDT). Дата введення 27.12.2016 р.
9. ISO/IEC/IEEE 16326:2009 Systems and software engineering — Life cycle processes — Project management. 12.2009.
10. IPMA (Individual Competence Baseline). 2015.
11. PRINCE2 (PProjects IN Controlled Environments 2). 2010.
12. P2M (A Guidebook of Project and Program Management for Enterprise Innovation). 2013.
13. PMBOK (A Guide to the Project Management Body of Knowledge). 2017.
14. SWEBOOK (*Guide to the Software Engineering Body of Knowledge*). 2014.
15. BABOK (A guide to the business analysis. body of knowledge). 2020.

References:

1. Skorobaghata, Ju.R., Turchenko, I.V., Nghoj, K.Zh. and Ghorbach, V.V. (2019). "Osnovni problemni aspekty v upravlinni IT-proektamy [The main problematic aspects in IT project management]". *ISCIENCE.IN.UA Aktualjnye nauchnye yssledovanyja v sovremennom myre*, No 11 (55). 1, p. 183–186 [in Russian].
2. SDLC – Waterfall Model. URL: <http://www.tutorialspoint.com/sdlc> [in English].
3. The Definitive Guide to Project Management Methodologies, available at: <http://www.workamajig.com/blog/project-management-methodologies> [in English].
4. What is Scrum Methodology? available at: <http://resources.collab.net/agile-101/what-is-scrum> [in English].
5. Knybergh, Kh. and Skaryn, M. (2010). "Scrum y Kanban: vyzhymaem maksimum [Scrum and Kanban: making the most of it]", available at: <http://scrum.org.ua/wp-content> [in English].
6. Tjuljkyna, A.S. and Chzhan, Ju. (2019). "Sravnitelnyj analiz kaskadnoj y spyralnoj metodologyj upravlenija proektamy razrabotky proghramnogho obespechenij [Comparative analysis of waterfall and spiral methodologies of software development project management]", available at: <http://kmu.itmo.ru> [in Russian].
7. 1992. "Avtomatyzyrovannye systemy. Stadyj sozdanyja [Automated systems. Stages of creation]". *GhOST 34.601-90*. [in Russian].
8. 2016. "Inzhenerija system i proghramnogho zabezpechnnja. Procesy zhyttjevogho cyklu proghramnogho zabezpechnnja (ISO/IEC 12207:2008, IDT) [Systems and software engineering. Software life cycle processes]". *DSTU ISO/IEC 12207:2016* [in Russian].
9. ISO/IEC/IEEE 16326:2009 Systems and software engineering — Life cycle processes — Project management. 12.2009 [in English].
10. IPMA (Individual Competence Baseline). 2015 [in English].
11. PRINCE2 (PProjects IN Controlled Environments 2). 2010 [in English].
12. P2M (A Guidebook of Project and Program Management for Enterprise Innovation). 2013 [in English].
13. PMBOK (A Guide to the Project Management Body of Knowledge). 2017 [in English].
14. SWEBOOK (*Guide to the Software Engineering Body of Knowledge*). 2014 [in English].
15. BABOK (A guide to the business analysis. body of knowledge). 2020 [in English].

НАУКОВЕ ВИДАННЯ

**ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ
ТА СУСПІЛЬСТВО**

**INFORMATION TECHNOLOGY
AND SOCIETY**

**ВИПУСК 1 (3)
ISSUE 1 (3)**

2022

*Коректура
Ірина Чудеснова*

*Комп'ютерна верстка
Наталія Кузнецова*

Формат 60x84/8. Гарнітура Cambria.
Папір офсет. Цифровий друк. Ум. друк. арк. 7,21. Замов. № 0322/103. Наклад 300 прим.

Видавництво і друкарня – Видавничий дім «Гельветика»
65101, Україна, м. Одеса, вул. Інглєзі, 6/1
Телефон +38 (048) 709 38 69, +38 (095) 934 48 28, +38 (097) 723 06 08
E-mail: mailbox@helvetica.ua
Свідоцтво суб'єкта видавничої справи
ДК No 6424 від 04.10.2018 р.