

Міністерство освіти і науки України
НТУ «Дніпровська політехніка»
Ройтлінгенський університет техніки та економіки (Німеччина)
Еслінгенський університет прикладних наук (Німеччина)
Технічний університет Фрайберзька гірничо академія (Німеччина)
Університет Кобленц-Ландау (Німеччина)
Краківська гірничо-металургійна академія (Польща)
Вроцлавський технічний університет (Польща)
Дніпропетровський національний університет імені Олеся Гончара
ДКХ «Дніпровський машинобудівний завод»
ДАТ «КБ Дніпровське»



ПРОБЛЕМИ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ В ОСВІТІ, НАУЦІ ТА ПРОМИСЛОВОСТІ

XVII МІЖНАРОДНА КОНФЕРЕНЦІЯ

24 листопада 2022 року
м. Дніпро

Збірник наукових праць
№ 7

Дніпро
НТУ «ДП»
2023

УДК 622 (06)

П78

Редакційна колегія:

О.О. Азюковський, Г.Г. Півняк, О.Б. Іванов, І.М. Удовик, М.О. Алексєєв, В.І. Корнієнко, В.В. Гнатушенко, В.В. Ткачов, В.В. Слесарєв, Л.І. Мещеряков, Б.І. Мороз, А.В. Бубликов, Т.А. Желдак, Г. Грюллер, Н. Нойбергер, А. Дерен, Я. Сконечний, Яцек Стефанські, Павел Чарнул, О.І. Сироткіна.

П78 **Проблеми** використання інформаційних технологій в освіті, науці та промисловості: XVII міжнар. конф. (24 листопада 2022 р., м. Дніпро): зб. наук. пр. [Електронний ресурс] / ред. кол.: О.О. Азюковський та ін.; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Електрон. текст. дані – Дніпро: НТУ «ДП», 2023. – № 7. – 217 с. – Режим доступу: <https://ir.nmu.org.ua/handle/123456789/163499>. – Назва з екрана.

ISBN 978-966-350-777-4

Подано результати теоретичних та експериментальних досліджень з різних аспектів використання інформаційних технологій в освіті, науці та керування промисловістю. У публікаціях розглянуто питання створення та вдосконалення програмних засобів обробки та передачі інформації, математичного моделювання, дистанційної освіти, інформаційної безпеки та телекомунікації.

Для наукових, інженерно-технічних співробітників і студентів, які спеціалізуються в галузі обчислювальної техніки та інформаційних технологій.

УДК 622 (06)

ISBN 978-966-350-777-4

©НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
«Дніпровська політехніка», 2023

РОЗДІЛ 1

МІЖНАРОДНЕ СПІВРОБІТНИЦТВО У СФЕРІ ОСВІТИ, НАУКИ І ВИРОБНИЦТВА

УДК 005.8:004.4

Л.В. Кабак¹, С.І. Кострицька¹, М.В. Ларикова¹, В. Потемпа²

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

²Технічний університет «Краківська гірничо-металургійна академія
ім. Станіслава Сташиця» (AGH), м. Краків, Польща

XOPS AS «UNIVERSAL SOLDIER» ON THE BIG DATA «BATTLEFIELD»

Анотація. Описано обов'язки та важливість XOps у області великих даних та аналізу, розкладена структура та множина позицій, що вкладена у спеціалізацію XOps.

Ключові слова: *Big Data, XOps, DevOps, SecOps, AIOps, MLOps, manage application, machine learning, operations.*

Introduction. Nowadays, oil for IT-world called Big Data is one of the most popular, mainstream, and expensive things. Everyone surrounded by data, which demand a lot of space, algorithms to analyze and power to give a conclusion. And business enunciate expanding of big data as progressive fresh product in the market.

Formulation of the problem. Big Data area calls for managing the solutions of applications. Operations being a part of any IT-product provide the ideal way to do this. XOps [1, 2] is a combined total of plenty different positions/methodologies that provide support for the application in the delivery of the final product, monitoring, version control, fault tolerance, process automation (Fig 1). The requirements that emerge from the Big Data area include knowledge of machine learning, the ability to improve data analysis, and experience of work with cloud technologies.

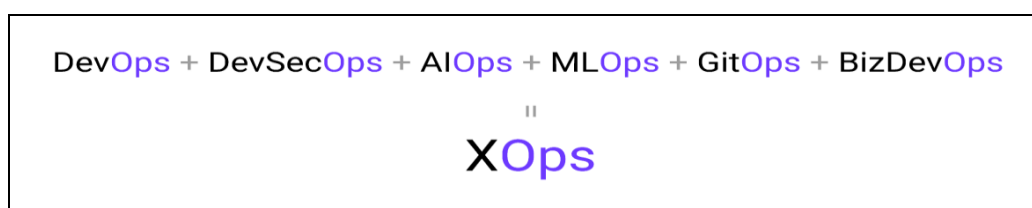


Figure 1. Structure of XOps

Each of the positions/methodologies has its 'Ops' ending meaning that a person should be involved in the operation, the main aim of which is to connect three processes: development, testing and maintenance. Therefore, only the environment and the direction change, and the "universal soldier" is the combination of different positions into one methodology.

The main content of the work. DevOps (Development Operations) is the most common position, as is SecOps (Fig. 1). The main work of such specialist is to set up pipelines, continuously deliver the product to the end user, automate the process of continuous testing (both in pipelines and in the form of regression testing), support services and project builds (Fig. 2). However, in the case of big data, this specialization requires a deep dive into the cloud services and the use of resources provided by another company for data storage and analysis.

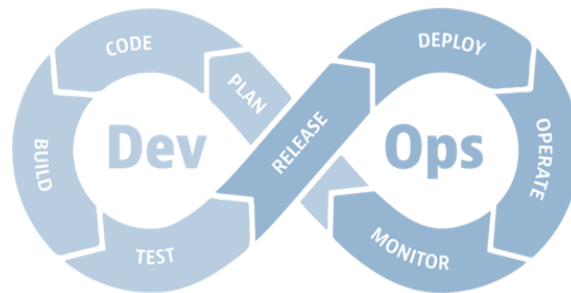


Figure 2. DevOps lifecycle [3]

The purpose SecOps (Security Operations) unlike DevOps is to protect and secure the project. The responsibility of SecOps is to provide the safety of the data on any digital media, create a safe space for the project, configure the analysis of suspicious activities, and eliminate them if necessary.

Both of SecOps and DevOps are in any project, including the ones that do not have a sufficiently large amount of incoming information for processing by the application.

As a rule, GitOps and DevOps have the same responsibilities due to the specificity of the work. However, DevOps' list of duties also includes version control, the overwhelming majority of which is the Git system.

BizDevOps (Business Development Operations) [4] is more about a methodology than specialization, which involves working with business in a collaborative manner, which constantly analyzes the market, the needs of the current client and prioritizes the tasks in the current moment to maximize the income from the product (Fig. 3).

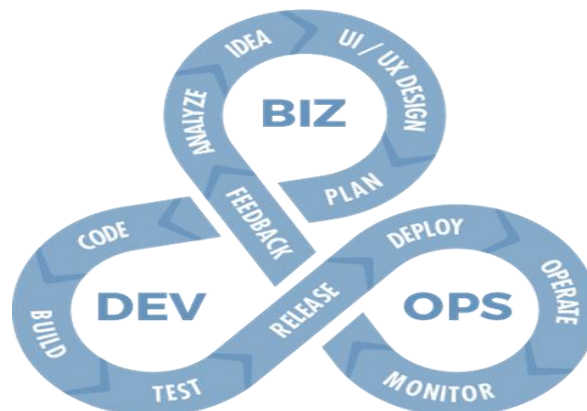


Figure 3. BizDevOps lifecycle [5]

The need for MLOps (Machine Learning Operations) [6] appears on a project with the addition of machine learning. Such specialists go deep into model optimization, training tuning, scaling, diagnostics, and forecasting that demands much more steps for the development and delivery of the product (Fig. 4).

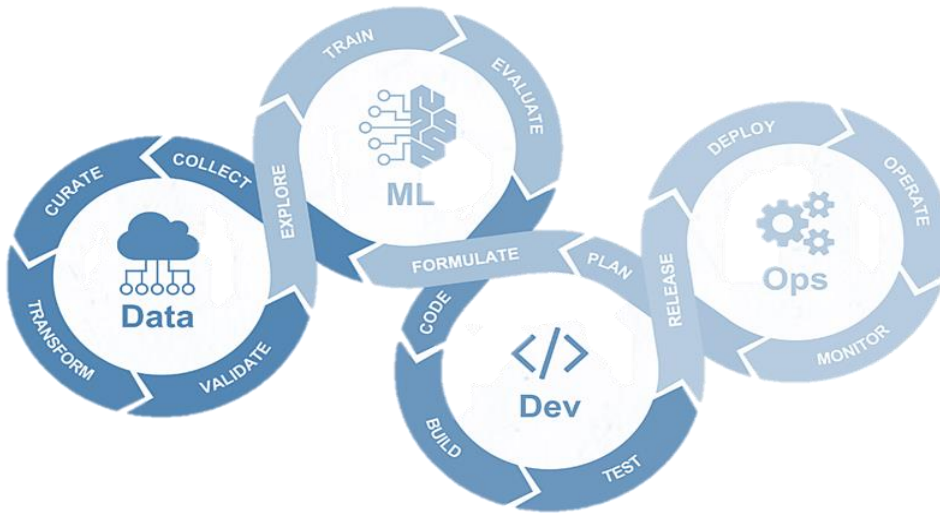


Figure 4. MLOps lifecycle [7]

AI Ops, as its name suggests, is an artificial intelligence that partially takes on the responsibilities of a specialist in the field of monitoring, analyzing, correlating, contextualizing, and forecasting. The task of such specialists is aimed at machine learning in order to process a huge amount of data on the side of the logs and environment metrics, the state of the network and information about the traffic of coming requests. The programs allow you to detect anomalies in the system, analyze the causes and suppress minor warnings grouping them by context.

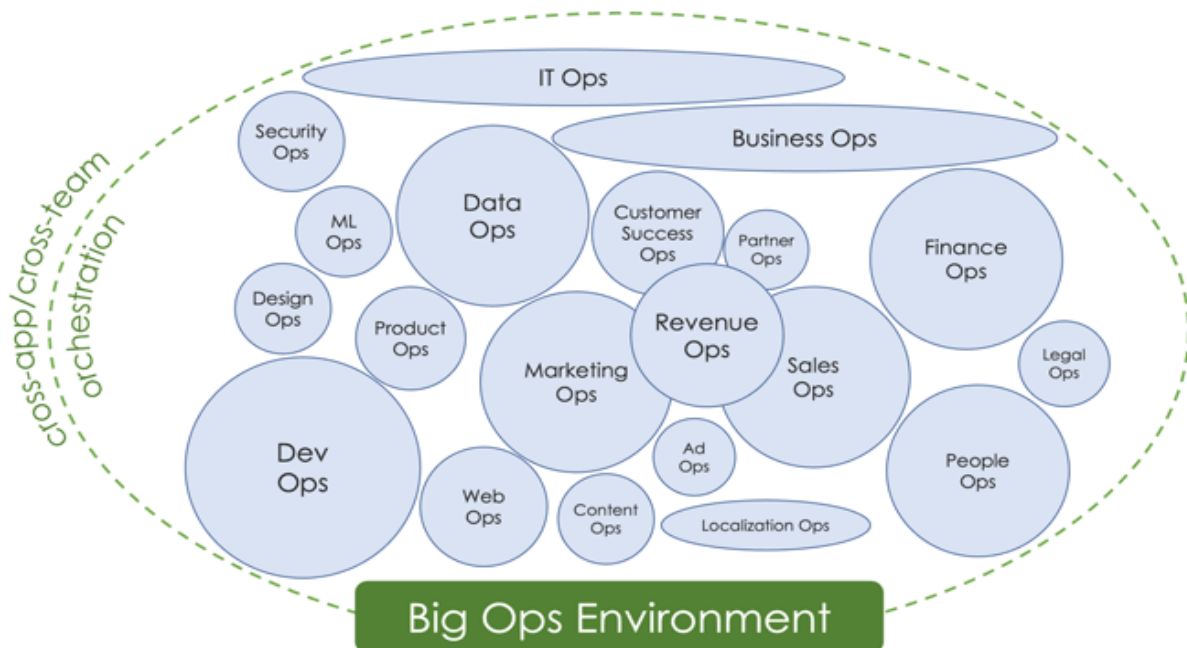


Figure 5. Variety of the 'Ops' [9]

This trend towards creating a huge number of directions among operations has already existed for a relatively long time and is constantly growing (Fig. 5). The concept of XOps became popular only at the end of 2021 - the beginning of 2022 [8].

Conclusions, having so many tasks, XOps cannot be one single person due to the number of responsibilities for configuration, environment maintenance, model analysis, and machine learning. This methodology can be represented by three specializations - DevOps, SecOps and MLOps. The greater the division of responsibilities between a large number of specialists, the more quality there will be in the work performed and in the product support.

ПЕРЕЛІК ПОСИЛАНЬ

1. Arvind P. XOps: The Rise of Smarter Tech Operations [Електронний ресурс] / Prabhakar Arvind. – 2022. – Режим доступу до ресурсу: <https://insidebigdata.com/2022/03/15/xops-the-rise-of-smarter-tech-operations/>.

2. Bhattacharyya S. XOps has emerged as a popular term—what is it? [Електронний ресурс] / Sreejani Bhattacharyya. – 2022. – Режим доступу до ресурсу: <https://analyticsindiamag.com/xops-has-emerged-as-a-popular-term-what-is-it/>.

3. Saif G. What is DevOps? Unpacking the purpose and importance of an IT cultural revolution [Електронний ресурс] / Gunja Saif. – 2021. – Режим доступу до ресурсу: <https://www.dynatrace.com/news/blog/what-is-devops/>.

4. Valerie S. BizDevOps (Business, Development and Operations) [Електронний ресурс] / Silverthorne Valerie. – 2016. – Режим доступу до ресурсу: <https://www.techtarget.com/searchsoftwarequality/definition/BizDevOps-Business-Development-and-Operations>.

5. Rizvan S. What are BizDevOps and DevSecOps? [Електронний ресурс] / Saatov Rizvan – Режим доступу до ресурсу: <https://rizvansaatov.medium.com/what-are-bizdevops-and-devsecops-7526ab043c6e>.

6. GOLED S. MLOps Vs DevOps: A Comparative Analysis [Електронний ресурс] / SHRADDHA GOLED. – 2021. – Режим доступу до ресурсу: <https://analyticsindiamag.com/ml-ops-vs-devops-a-comparative-analysis/>.

7. Gupta S. C. MLOps: Machine Learning Lifecycle [Електронний ресурс] / Satish Chandra Gupta. – 2022. – Режим доступу до ресурсу: <https://www.ml4devs.com/articles/ml-ops-machine-learning-life-cycle/>.

8. Gill N. S. 10 Latest Trends in Big Data Analytics for 2022 | Ultimate Guide [Електронний ресурс] / Navdeep Singh Gill. – 2022. – Режим доступу до ресурсу: <https://www.xenonstack.com/blog/latest-trends-in-big-data-analytics>.

9. van Rijn J. 65+ Email Marketing Trends 2022: The Future of Email Marketing & Marketing Automation [Електронний ресурс] / Jordie van Rijn. – 2022. – Режим доступу до ресурсу: <https://www.emailmonday.com/email-marketing-automation-trends-predictions/>.

М.О. Алексєєв¹, О.М. Алексєєв¹, В.В. Кремньов¹, П. Шчешневській²

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

²Технічний університет «Краківська гірничо-металургійна академія ім. Станіслава Сташиця» (AGH), м. Краків, Польща

КОНТРОЛЬ ПАРАМЕТРІВ ОБ'ЄКТІВ КЕРУВАННЯ З ВИКОРИСТАННЯМ ПЕРЕБУДОВАНИХ МАТРИЧНИХ СПЕКТРАЛЬНИХ ОПЕРАТОРІВ

Анотація. В роботі пропонується метод контролю параметрів об'єкту керування на основі застосування субоптимальних по Карунену-Лоеву перебудованих спектральних операторів, пристосованих до еталонів класів сигналів, які супроводжують функціонування об'єктів керування. Досліджено можливість визначення параметрів ядер матриць спектрального оператора через мінімізацію середньоквадратичної помилки при представленні вихідного вектора його оцінкою.

Ключові слова: об'єкт керування, контроль параметрів, перебудовані оператори.

Вступ. Для формування інформативних ознак сигналів при контролі параметрів об'єктів управління використовуються методи, засновані на застосуванні перетворень, які при відповідному виборі базисної системи забезпечують адекватність аналізованої інформації при високому ступені декореляції інформативних компонентів [1]. У [2] викладено підхід до формування матричних спектральних операторів, заснований на їх поданні через узагальнене спектральне ядро, що дозволяє отримати безліч базисних систем функцій з алгоритмами швидких перетворень. В основі матрично-ядерного представлення спектрального оператора лежить можливість синтезу базису з елементарних мікроструктур-ядер, узагальнена форма яких має вигляд:

$$V_{rl} = \sqrt{2} \begin{bmatrix} \cos \varphi_{rl} & e^{j\Theta_{rl}} \sin \varphi_{rl} \\ \sin \varphi_{rl} & -e^{j\Theta_{rl}} \cos \varphi_{rl} \end{bmatrix},$$

де $\varphi \in [0, 2\pi]$; $\Theta_{rl} \in [0, 2\pi]$; $r = \overline{1, n}$; $l = \overline{1, 2^{n-1}}$

Кути параметри є ступенями свободи ядра і визначають конкретні елементи спектрального оператора.

Концепція ядерного представлення спектральних операторів як спрощує процедуру їх синтезу, а й дозволяє проводити обґрунтований за комплексом критеріїв вибір базису. Однак на жаль ці базиси не є в загальному випадку оптимальними за Каруненом-Лоевом.

Постановка задачі. Для досягнення поставленої мети в роботі пропонується метод контролю параметрів об'єктів управління, на основі використання ортогональних систем базисних функцій субоптимальних по Карунену-Лоеву з алгоритмами швидкого перетворення, що враховують

особливості аналізованих сигналів, мають можливість параметричної перебудови.

Основний зміст роботи. Спектральний оператор, що перебудовується [1], пристосований до еталона класу m $\mathbf{X}_{\text{г}}$ в тому сенсі, що у спектральній області базису, що перебудовується, еталону зіставляється всього один коефіцієнт, відмінний від нуля. Базис, що перебудовується, є оптимальним для еталона в сенсі критерію ентропії спектральних ознак. За процедурою побудови базису, що перебудовується $\mathbf{X}_{\text{г}}$ є перший рядок матриці спектрального оператора. Позначимо матрицю спектрального оператора, що перебудовується, для класу m як $\mathbf{V}^m = [\mathbf{V}_1^m, \mathbf{V}_2^m, \dots, \mathbf{V}_N^m]^T$. Для рядків \mathbf{V}_i^m матриці \mathbf{V}^m виконуються умови ортонормування.

Для формування вектора інформативних ознак за вихідними даними, представленими у вигляді вектора \mathbf{X}^m пропонується визначити дискретні спектри $\{ \mathbf{Y}^m \}$, $m = \overline{1, M}$:

$$\mathbf{Y}^m = \frac{1}{N} \mathbf{V}^m \mathbf{X}.$$

Вектор \mathbf{Y}^m в (1) являє собою вектор спектральних коефіцієнтів по системі ортогональних функцій \mathbf{V}^m , що перебудовується для класу m . У запропонованому методі класифікації вектора інформативних ознак паралельно формуються m систем інформативних ознак по N ознак в кожній системі.

Для кожного вектора \mathbf{X}^m , що належить класу m векторів вихідних даних, отримаємо $\mathbf{Y}^m = \mathbf{V}^m \mathbf{X}^m$, де $\mathbf{X}^{mT} = [x_1^m, x_2^m, \dots, x_N^m]$; $\mathbf{Y}^{mT} = [y_1^m, y_2^m, \dots, y_N^m]$. Вираз для обчислення \mathbf{X}^m можна подати у такому вигляді

$$\mathbf{X}^m = y_1^m \mathbf{V}_1^m + y_2^m \mathbf{V}_2^m + \dots + \mathbf{V}_N^m y_N^m = \sum_{i=1}^N y_i^m \mathbf{V}_i^m.$$

Для отримання мінімальної кількості інформативних ознак необхідно, щоб вони адекватно представляли вектор вихідних даних \mathbf{X}^m . Для оцінки оптимальності перетворення використовуємо середньоквадратичний критерій. Отримаємо оцінку $\tilde{\mathbf{X}}^m$ вектора \mathbf{X}^m , представивши його M членами. Решта $N-M$ координат y_i^m замінимо константами c_i^m . Тоді

$$\tilde{\mathbf{X}}^m = \sum_{i=1}^M y_i^m \mathbf{V}_i^m + \sum_{i=1}^N c_i^m \mathbf{V}_i^m.$$

Помилка у представленні вектора \mathbf{X}^m його оцінкою $\tilde{\mathbf{X}}^m$ може бути представлена у вигляді вектора помилки:

$$\Delta \mathbf{X}^m = \mathbf{X}^m - \sum_{i=1}^M y_i^m \mathbf{V}_i^m - \sum_{i=1}^N c_i^m \mathbf{V}_i^m .$$

Після ряду перетворень середньоквадратична помилка при представленні вектора \mathbf{X}^m вектором $\tilde{\mathbf{X}}^m$ визначиться так:

$$\sigma^m = \sum_{i=M+1}^N \mathbf{V}_i^{mT} \mathbf{K}_x^m \mathbf{V}_i^m ,$$

де \mathbf{K}_x^m – коваріаційна матриця \mathbf{X}^m .

Елементи векторів \mathbf{V}_i^m визначають параметри ядер спектрального оператора \mathbf{V}^m . Оскільки в деякій частині ядер параметри не залежать від вихідного еталона, можна провести подальшу оптимізацію базису під вимогу конкретної задачі.

Наукова новизна розробки складається в удосконаленні метод контролю параметрів технологічних об'єктів керування, що базується на використанні перебудованих матричних спектральних операторів, кожен з яких синтезований за еталоном технологічної ситуації.

Висновок. Запропонований підхід з використанням перебудовуваних субоптимальних по Карунену-Лоеву базисів дозволяє вибрати побудувати базиси, що враховують як глобальні властивості аналізованого сигналу на всьому інтервалі визначення, так і вейвлетоподібні базиси, які враховують локальні зміни сигналу подібно до вейвлетів Хаара. Це дозволяє в рамках єдиного підходу поєднати переваги спектрального аналізу та вейвлет-аналізу сигналів під час оперативного контролю параметрів об'єктів керування.

ПЕРЕЛІК ПОСИЛАНЬ

1. Power Engineering, Control and Information Technologies in Geotechnical Systems / [під ред. Genadiy Pivnyak, Oleksandr Beshta, Mykhaylo Alekseyev]. - Leiden: CRC Press/Balkema, 2015. - 220 с.

2. Energy Efficiency Improvement of Geotechnical Systems: International Forum on Energy Efficiency / [під ред. Genadiy Pivnyak, Oleksandr Beshta, Mykhaylo Alekseyev]. - Leiden : CRC Press/Balkema, 2013. - 244 с.

І.М. Удовик¹, Л.І. Мещеряков¹, Є.С. Попова¹, О.І. Сироткіна²

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

²School of Computer Science University of Windsor, Windsor, Ontario, Kanada

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ФРЕЙМВОРКУ FLUTTER ПРИ РОЗРОБЦІ ВИСОКОДИНАМІЧНИХ МОБІЛЬНИХ ДОДАТКІВ

Анотація. Описано процес дослідження ефективності фреймворку Flutter при розробці високодинамічних мобільних додатків. Дослідження проводиться за рахунок декларативного підходу фреймворку Flutter при розробці програмного забезпечення для мобільних пристроїв на основі єдиної бази коду.

Ключові слова: мобільні додатки, операційні системи, фреймворк, iOS, Android, Flutter, Dart, React Native.

Вступ. Мобільні додатки на сьогоднішній день беруть безпосередню участь у повсякденному житті. Існує цілий ряд мобільних додатків, які використовуються для полегшення діяльності людини: соціальні мережі, електронна пошта, онлайн покупки, транспортні додатки, додатки для обміну текстовими та відеоповідомленнями тощо.

Існують мільйони додатків, доступних у різних магазинах мобільних додатків, таких як Google Play Store та App Store. Постійно зростаюче їх використання спонукає компанії та індивідуальних розробників інвестувати свій час та гроші в розробку додатків, орієнтованих на цих користувачів. Такий високий попит змусив компанії та розробників шукати нові способи розробки мобільних додатків (МД) для охоплення великої кількості користувачів з меншими витратами на розробку.

Різноманітність мобільних платформ була основною перешкодою для бізнесу та розробників для пошуку нових шляхів розробки мобільних додатків, а не слідування економічно неефективним методам розробки мобільних додатків орієнтованих під конкретну платформу. Крім того, розробка мобільних додатків потребує частих удосконалень та адаптацій, щоб відповідати швидкозмінним вимогам зручності використання.

Виробники портативних пристроїв все частіше адаптують нові дизайни, які варіюються від зміни розмірів екрану пристрою до поліпшення можливостей пристрою. Тож розробка мобільних додатків нині є однією з найпопулярніших завдань в сфері інформаційних технологій. Зростає потреба в розробці якісного і відповідного сучасним тенденціям продукту.

Успіх і відповідно і якість МД залежить від вибору платформи та технологій для його розробки. Саме вони зумовлюють життєздатність і конкурентоспроможність додатку, його функціонал, масштабованість і складність обслуговування. Також це впливає і на вартість розробки. Сьогодні існують різні методи і фреймворки, які допомагають створювати МД. Таким

чином, можливості сучасних технологій розробки дозволяють створювати мобільні додатки різної складності.

Тож вибір методу розробки мобільного додатка – це дуже важливий етап в його розробці, на який впливають кілька факторів, таких як технічна оцінка розробників, потреба в доступі до інформації на пристрої, вплив швидкості інтернету на додаток. Вибір тієї чи іншої платформи також залежить від бюджету і вимог, що пред'являються до майбутнього додатку.

Отже, мобільний додаток повинен адаптуватися до всіх цих нових змін, щоб зберегти своїх користувачів. Хоча існують мільйони мобільних додатків на різних ринках додатків, їх загалом можна розділити на три категорії: додатки для конкретних платформ, адаптивні веб-додатки та крос-платформні нативні додатки [2].

Постановка задачі. Для досягнення мети дослідження необхідно розв'язати наступні задачі:

- Провести аналіз існуючих технологій, які дозволяють займатися одночасною розробкою під різні операційні системи. (здійснити вибір інструментальних засобів розробки мобільного додатку, дослідивши особливості розробки додатків для операційних систем Android та iOS.).

- Проаналізувати нове рішення від Google – фреймворк Flutter, як засіб для розробки мобільних додатків.

- Здійснити порівняння та побудувати структурні моделі усіх можливих рішень для розробки програмного забезпечення для мобільних операційних систем.

- Розробити додаток за допомогою фреймворку Flutter для визначення ефективності програмного забезпечення для мобільних пристроїв.

- Підвести підсумки щодо вибору найкращої технології для розробки мобільного програмного забезпечення.

Основний зміст роботи. На даний момент швидко розвивається кросплатформний підхід, який дозволяє розробляти додатки для різних платформ. Переваги при створенні кросплатформних додатків такі:

- суттєве зменшення витрат на адаптацію існуючого програмного забезпечення або розробку нового;

- висока швидкість роботи завдяки компіляції JavaScript в нативний код;

- зменшення кількості фахівцем при створенні мобільного додатку під 2-3 мобільні операційні системи на кросплатформному фреймворку;

- можливість інтеграції нативних модулів.

Найбільш популярними такими фреймворками є: React Native, Xamarin, Flutter тощо. Розглянемо детальніше такі фреймворки як React Native та Flutter, як найбільш популярні на даний час.

Фреймворк React Native. React Native не використовує HTML. Натомість повідомлення з потоку JavaScript використовуються для маніпулювання нативними View. Принцип роботи React Native дуже схожий з React, за винятком одного важливого моменту – React Native не маніпулює DOM за

допомогою Virtual DOM, тому що в мобільних додатках він відсутній. Virtual DOM працює у фоновому процесі, який інтерпретує JavaScript, що написаний безпосередньо мобільними розробниками, на кінцевому пристрої і спілкується із нативною кодовою базою за допомогою се-ріалізації, пакетного та асинхронного мостів. Наявний код обертається за допомогою реактивних компонентів, які взаємодіють з нативними application programming interface (API) за допомогою декларативної парадигми JavaScript та інтерфейсу користувача.

Це надає можливість розробляти нативні програми за допомогою невеликих команд розробників, а в деяких випадках можна обмежитися одним розробником.

Фреймворк Flutter. Flutter представляє фреймворк від компанії Google, який дозволяє створювати кросплатформні програми, які можуть використовувати один і той же код. Спектр платформ, в яких він застосовується, широкий – це веб-додатки, мобільні програми під Android та iOS, графічні програми під настільні операційні системи Windows, MacOS, Linux тощо.

Дебютна версія Flutter, яка працювала на операційній системі Android, була відома під кодовою назвою «Sky». SDK був висвітлений на саміті розробників Dart у 2015 року, із заявкою на можливість послідовно відображати зображення з частотою більш ніж 120 кадрів в секунду. Під час виступу розробників з Google у Шанхаї було оголошено Flutter Release Preview 2 – потужний реліз перед стабільним релізом Flutter 1.0. На заході Flutter Live, 4 грудня 2018 року, було оголошено о Flutter 1.0. На заході Flutter Interactive, 11 грудня 2019 року, з'явився Flutter 1.12 та було оголошено, що Flutter – це перша платформа інтерфейсу, що призначена для обчислення навколишнього середовища.

Для створення додатків Flutter використовується мова Dart, яка містить багато вдосконалень. Під час розробки програм, Flutter транслює код написаний на Dart у нативний код програми за допомогою Dart AOT (компіляція програми перед його запуском), яку можна запускати на Android або iOS або іншій платформі. Однак при розробці програми для її прискорення Flutter використовує JIT (компіляція програми під час його запуску).

Що використовується для розробки під Flutter? Для написання коду програми можна використовувати будьякий улюблений текстовий редактор і потім за допомогою утиліт командного рядка з Flutter SDK компілювати програму. Однак для таких середовищ, як Android Studio та IntelliJ IDEA, а також текстового редактора Visual Studio Code, компанія Google випустила спеціальні плагіни, які дозволяють спростити розробку. Тому найчастіше для розробки під Flutter використовуються саме Android Studio та IntelliJ IDEA.

Таким чином, набір інструментів представлених фреймворків React Native та Flutter надають можливість створення, інтеграції і впровадження продуктів із використанням однієї мови програмування для кросплатформних мобільних

додатків у багатьох сферах: в медицині та охороні здоров'я, для навчання студентів, впровадження технологій читання по губах, миттєвих перекладачів, для моделювання та проектування у виробництві, комерційних цілях, використання машинного зору для запровадження технологій самокеруючих автомобілів, для підвищення показників продажів і спілкування з споживачами, налагодження алгоритму продажів тощо.

Наукова новизна полягає у запропонованому та обґрунтованому способі удосконалення та пришвидшення роботи над високодинамічними мобільними додатками за допомогою фреймворку Flutter, який, в свою чергу, гарантує ефективність роботи системи завдяки декларативному підходу до побудови мобільних інтерфейсів.

Висновки. В результаті цієї роботи теоретично і на практиці були досліджені фреймворки для розробки мобільних додатків – Flutter та React Native, а також було проведено порівняння поміж ними.

Були досліджені різниці в підходах, процесі розробки та у підтримці існуючих додатків. Наразі Flutter є сучаснішим за фреймворк React Native, і може похвалитися вищою продуктивністю, можливістю налаштування та кращою документацією. Flutter здатний забезпечити розробку більш масштабованих, швидких та високопродуктивних додатків, сумісних із численними платформами.

З наведеного вище можна зробити такий висновок, що фреймворк Flutter випереджає React Native за популярністю серед розробників, та багатьма іншими критеріями, але в бізнесі, наразі, все ще частіше використовують React Native через відносно молодий вік Flutter.

ПЕРЕЛІК ПОСИЛАНЬ

1. Офіційна документація Flutter [Електронний ресурс]. – Режим доступу : <https://flutter.dev/>.
2. Офіційна документація React Native [Електронний ресурс]. – Режим доступу : <https://reactnative.dev/>.
3. Vladimir Novick. React Native – Building Mobile Apps with JavaScript. – Chapter 4 – 2017 – P. 114.
4. Rob Whitaker. Developing Inclusive Mobile Apps: Building Accessible Apps for iOS and Android. United States of America: Apress, 2020. 359 с.
5. Windmill E. Flutter in Action Л.: Manning Publications, 2019. – 368 с.
6. Abshak P., Abshishek N. React Native for Mobile Development (2nd ed.) Л.: Apress, 2017. – 648 с.
7. Frank Zammetti. Practical Flutter: Improve your Mobile Development with Google's Latest Open-Source SDK, 2020. 364 с.
8. Wargo J. Apache Cordova API Cookbook. Л.: Gardner Books, 2017. – 328 с.
9. Best Language For Mobile App Development. [Електронний ресурс]. – Режим доступу : <https://www.webiotic.com/best-language-for-mobile-app-development-what-youneed-to-know/>.
10. Shaun Lewis, Mike Dunn. Native Mobile Development: A Cross-Reference for iOS and Android. – Chapter 1 – 2019 – P. 1-3.
11. «Android. Збірник рецептів » від Яна Ф. Дарвіна. По суті книга – покрокова інструкція з прикладами для вирішення завдань по роботі з web-службами для досвідчених знавців Java і інтерфейсу Java SE API.

12. Eisenman, B. Learning React Native [Text] / Bonnie Eisenman. ¾ 2nd edition. ¾ Sebastopol: O'Reilly Media, 2017. ¾ 242 p.
13. Garbade, M. Native vs. cross-platform app development: pros and cons [Електронний ресурс] / Dr. Michael J. Garbade. ¾ Режим доступу: <https://codeburst.io/native-vs-cross-platform-app-development-prosand-cons-49f397bb38ac>.
14. Brian Kayfitz. Flutter Cookbook: Over 100 proven techniques and solutions for app development with Flutter 2.2 and Dart, 2020. 376 с.
15. Flutter, Flutter for iOS developers [Електронний ресурс] / Flutter. ¾ Режим доступу: <https://flutter.dev/docs/get-started/flutter-for/ios-devs>.
16. Napoli, M. Beginning Flutter: A Hands On Guide to App Development [Text] / Varco L. Napoli. ¾ 1st edition. ¾ Birmingham: Wrox, 2019. ¾ 528 p.

УДК 519.876.5

Є.О. Антоненко¹, Л.С. Коряшкіна¹, Т. Гец²

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

²Університет Кобленц-Ландау, Кобленц, Німеччина

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ЕПІДЕМІЇ З УРАХУВАННЯМ МОБІЛЬНОСТІ

Анотація. Запропоновано математичну модель поширення інфекції з врахуванням маятникової міграції населення на прикладі COVID-19. Представлений аналіз результатів обчислювальних експериментів, який підтверджує вплив мобільності населення на швидкість розповсюдження інфекції.

Ключові слова: Інфекційна модель, маятникова міграція, моделювання поширення інфекції, система диференціальних рівнянь, мобільність населення.

Вступ. За багатовікову історію людству доводилось неодноразово стикатися з різноманітними захворюваннями, деякі з них перетворювались на епідемії. Залежно від природи вірусу, кількості людей, їх місцезнаходження, клімату, санітарних умов, це розповсюдження інфекції набувало різних масштабів, найкрупніші з яких уносили життя більше десятків або навіть сотень мільйонів людей.

Зараз, зі стрімким розвитком медицини, більшість вірусів, що провокували інфекції сторіччя тому, не являє собою загрози. Майже від усіх знайдені ліки, або вони взагалі припинили своє існування через покращення життя людства. Але від деяких інфекцій і досі немає ліків, хоча були знайдені способи сповільнення розвитку зараження людства. Яскравим прикладом може стати ВІЛ, початком епідемії якого вважається 1980 рік, та антиретровірусна терапія, яка дозволяє пацієнтам продовжити строк життя після зараження.

У сучасному світі дуже важко уявити собі вірус, який зможе похитнути життя не тільки на якійсь окремій частині Землі, але розповсюдиться на всі

континенти нашої планети, фундаментально змінить спосіб життя, спілкування та кардинально вплине на взаємодію та переміщення населення. Заради уповільнення поширення інфекції, яка потім носитиме назву COVID-19 та переросте у пандемію, будуть прийняті різноманітні протиепідемічні заходи, але одним із першим та найбільш очевидних та дієвих способів перешкоджанню розповсюдження інфекції стало обмеження контакту осіб між собою.

Метою даної роботи є обґрунтування доцільності врахування маятникової міграції (регулярного переміщення населення між двома і більше населеними пунктами, пов'язаного з метою роботи) під час математичного опису розповсюдження захворювань.

Вихідні дані для дослідження: 1) інформація про переміщення населення між сімома містами Німеччини землі Рейнланд-Пфальц (так звана матриця мобільності) у періоди вересень – жовтень 2020-го року та січень – лютий 2021-го року; статистика нових щоденних виявлених захворювань на COVID-19 з Інституту Роберта Коха у ті самі часові проміжки; 3) загальна кількість населення кожного міста, за які прийнято кількість індивідуумів, які пересувалась або, навпаки, залишалась у місті, а не реальну кількість населення кожного з міст через невелику похибку (в середньому 2%).

Постановка задачі. Для досягнення поставленої мети цієї роботи, були поставлені наступні задачі:

- Реалізувати класичну SIR модель без урахування міграції, використовуючи науково встановлені коефіцієнти поширення інфекції та одужання на основі вихідних даних;
- Побудувати модифіковану SIR модель, включивши до неї вплив міграції населення із використанням тих самих вже відомих коефіцієнтів;
- Налаштувати параметри модифікованої SIR моделі, підібравши коефіцієнт інтенсивності контактів індивідуумів з подальшим їх інфікуванням, спираючись на реальні дані про кількість людей, що захворіли;
- Порівняти отримані результати, зробити висновки.

Основний зміст роботи. Математичне моделювання швидкості поширення інфекції здійснено на основі детерміністичної популяційної SIR моделі, в якій населення певного регіону розглядається як сукупність груп, що відображають різний статус індивідів по відношенню до захворювання: 1) кількість сприйнятливих до хвороби осіб (Susceptible – S); 2) інфікованих (Infected – I); 3) «видалених» осіб із системи, маючи на увазі як тих, хто одужав, так і тих, хто помер від хвороби (Removed – R), і, таким чином, виключаючи можливість повторного для них захворювання. У середині кожної групи індивіди вважаються нерізними між собою. Чисельності груп змінюються згодом внаслідок переходу індивідів з однієї групи до іншої. Вибір цієї моделі обумовлений її простотою в побудові й використанні, можливістю аналітичного дослідження й легкістю настройки на реальні дані, хоча вона й не враховує інкубаційний період коронавірусу, який прийнято вважати 2 тижнями.

Отже, в даній роботі розглядається найбільш базовий випадок поширення інфекції. Зазначену модель узагальнено на випадок врахування зміни численності груп, пов'язаної з міграцією населення. Модифікована модель отримана додаванням складових, що описують взаємодії осіб із різних міст, тобто обмін інфекцією відбувається не тільки в рамках закритого суспільства, а з-поміж низки міст.

В роботі порівняно результати моделювання поширення інфекції коронавірусу за допомогою класичної SIR-моделі та її узагальнення на випадок врахування міграції, коли параметри моделі (коефіцієнт інтенсивності контактів індивідуумів з подальшим інфікуванням та коефіцієнт інтенсивності ви здоровлення інфікованих індивідів) є відомими, отриманими з наукових джерел. Також проведено налаштування параметру моделі на реальні статистичні дані без урахування міграції та з введенням до неї матриці мобільності.

Наукова новизна цієї роботи – відому SIR модель узагальнено на випадок врахування маятникового переміщення осіб з одного міста до інших, що відбуваються кожного дня; досліджено вплив міграції населення на швидкість поширення інфекції.

Висновки. За результатами проведених експериментів можна зробити наступний висновок про те, що мобільність населення між містами, яка має назву маяткової міграції, може суттєво впливати на швидкість поширення інфекцій, і під час математичного опису епідемій варто враховувати такі міграційні процеси.

УДК 004.415.3:681.6

В.В. Спірінцев¹, А.Л. Ширін¹, А.А. Сарібекян¹, Мачей Потемпа²

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

²Сілезький університет, факультет Науки і техніки, Польща

ВИКОРИСТАННЯ REACT ЯК ФРЕЙМВОРКУ ДЛЯ РОЗРОБКИ ВЕБ-УТИЛІТ

Анотація. Розкрито переваги використання React як фреймворку для розробки веб-утиліт. Розкрито принципи роботи Virtual DOM.

Ключові слова: веб-фреймворк, Virtual DOM, ізоморфність, веб-утиліта, dirty checking, нативність.

Вступ. За останнє десятиліття спостерігається стійка тенденція розвитку веб-технологій та інструментів для їх розробки. Однією з платформ для створення веб-інтерфейсів користувача на основі UI компонентів є React. На сьогодні він впевнено входить до трійки лідерів серед інших фреймворків. Більшість розробників використовують React як базу при розробці односторінкових,

мобільних або серверних додатків. За проведеними дослідженнями утиліти створені на базі React працюють швидше, за базовий DOM.

Основний зміст роботи. React - це бібліотека для створення інтерфейсів користувача. Однією з її відмінних рис є можливість використовувати JSX, мову програмування з близьким до HTML синтаксисом, який компілюється в JavaScript. Розробники можуть досягати високої продуктивності додатків за допомогою вбудованого у React способу подання та взаємодії з об'єктами – Virtual DOM. [1]

За допомогою React ви можете створювати ізоморфні програми, які допоможуть вам позбутися неприємної ситуації, коли користувач з нетерпінням очікує, коли ж нарешті завершиться завантаження даних і на екрані його комп'ютера нарешті з'явиться щось окрім анімації завантаження. Створені компоненти можуть бути легко змінені і використані заново в нових проектах. Високий відсоток перевикористання коду спрощує його тестування, що, своєю чергою, призводить до вищого рівня контролю якості.

Нижче наведено переваги, які може отримати замовник при використанні React:

- Virtual DOM може підвищити продуктивність високонавантажених додатків, що може знизити ймовірність виникнення можливих повторень в коді та покращує користувацький досвід (рис.1);

- Використання ізоморфного підходу допомагає робити рендерінг сторінок швидше, тим самим дозволяючи користувачам почуватися комфортніше під час роботи з вашим додатком. Пошукові системи індексують такі сторінки краще. Оскільки один і той же код може бути використаний як у клієнтській, так і в серверній частині додатка, немає необхідності дублювати один і той же функціонал. В результаті час розробки та витрати знижуються;

- Завдяки перевикористанню коду стало набагато простіше створювати мобільні програми. Код, який був написаний під час створення сайту, може бути використаний для створення мобільного додатка. Якщо ви плануєте використовувати не тільки сайт, але й мобільний додаток, немає необхідності наймати дві великі команди розробників.

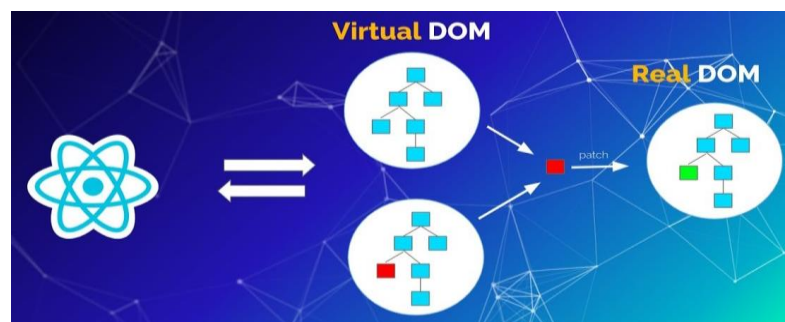


Рис. 1. Принцип використання роботи Virtual DOM у React

Ізоморфні програми. Коли ми говоримо про ізоморфні додатки або про ізоморфний JavaScript, ми маємо на увазі, що ми можемо використовувати той самий код як у серверній, так і в клієнтській частині програми. Коли користувач відкриває сайт у своєму браузері, вміст сторінки має бути завантажений із

сервера. У випадку зі SPA-додатками (Single Page Application), це може тривати деякий час. Під час завантаження користувачі бачать порожню сторінку або анімацію завантаження. Враховуючи, що за сучасними стандартами очікування протягом більш ніж двох секунд може бути помітною незручністю для користувача, скорочення часу завантаження може виявитися вкрай важливим. А ще одна вагома проблема: пошукові машини не індексують такі сторінки так добре, як нам хотілося б. Виконання JavaScript коду на стороні сервера допомагає виправити такі проблеми [2].

Якщо ви створюєте ізоморфні програми, ви можете отримати помітну вигоду, роблячи рендерінг на стороні сервера. Після завантаження сторінки ви все ще можете продовжувати рендерінг компонентів. Така можливість рендерінгу сторінок як на сервері, так і на клієнті призводить до помітних переваг, таких як можливість кращого індексування сторінок пошуковими машинами та поліпшення досвіду користувача. Більше того, такий підхід дозволяє знизити час, що витрачається на розробку. При використанні деяких сучасних фреймворків, ви повинні створювати компоненти, які повинні рендеруватись на стороні сервера, а також шаблони для клієнтської сторони програми. Розробники React можуть створювати компоненти, які працюють на обох сторонах.

Virtual DOM. Document Object Model, або DOM - це спосіб подання та взаємодії з об'єктами в HTML, XHTML та XML документах. Відповідно до цієї моделі, кожен такий документ є ієрархічним деревом елементів, зване DOM-деревом. Використовуючи спеціальні методи, ми можемо отримати доступ до певних елементів нашого документа і змінювати їх так, як ми хочемо. Коли ми створюємо динамічну інтерактивну веб-сторінку, ми хочемо, щоб DOM оновлювався так швидко, як це можливо після зміни стану певного елемента. Для даної задачі деякі фреймворки використовують прийом, який називається *dirty checking* і полягає в регулярному опитуванні стану документа і перевірці змін у структурі даних. Подібне завдання може стати справжнісіньким головним болем у разі високонавантажених додатків. Virtual DOM, своєю чергою, зберігається у пам'яті. Саме тому в момент, коли «справжній» DOM змінюється, React може змінювати Virtual DOM миттєво. React «збирає» такі зміни, порівнює їх зі станом DOM, а потім перемальовує компоненти, що змінилися (рис. 2) [3].

При цьому підході не треба робити регулярне оновлення DOM, можна здійснювати рендерінг за сервер зовсім як за клієнта. Саме тому може бути досягнуто вищу продуктивність React-додатків.

Перевикористання коду. Мобільні програми мають деякі переваги в порівнянні з сайтами. Їх можна використовувати без підключення до Інтернету. Вони мають доступ до таких можливостей пристрою, як спливаючі сповіщення. Також вони дають змогу бути в контактi з вашими користувачами в режимі 24/7. React Native – це фреймворк, який дозволяє створювати мобільні програми за допомогою React. Логіка програми пишеться на JavaScript, таким чином програмісту не потрібно відмовлятися від звичних прийомів веб-розробника. Все, що потрібно, навчитися писати специфічний для пристрою код, який адаптує компоненти, раніше створені для веб-сайту до нового середовища проживання.

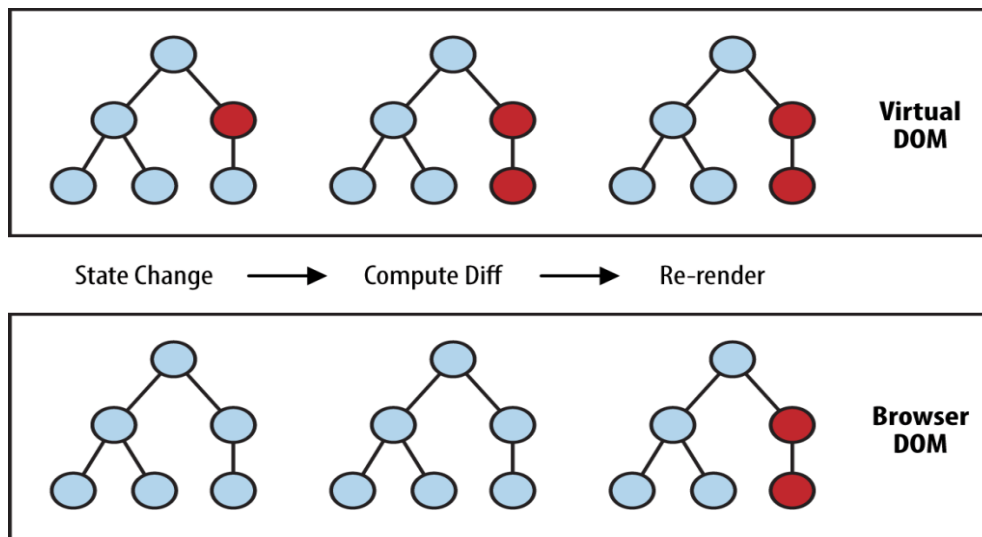


Рис. 2. Принцип роботи Virtual DOM та Browser DOM.

Якщо ви плануєте створити корпоративний веб-додаток і не цілком впевнені, чи буде розробка мобільної версії цього ж додатку гарною ідеєю, ось що ви повинні пам'ятати. React Native дозволяє використовувати вже наявну логіку веб-програми при створенні мобільного додатка. Це означає, що команда розробників може використовувати той же код, який був використаний у процесі створення сайту, замість того, щоб починати з чистого аркуша.

Крім швидшої розробки, перевикористання коду дозволяє уникнути великої кількості помилок. Якщо ви створюєте добре спроектовані компоненти, які потім використовуєте знову, вам потрібно буде писати менше коду, коли ви вирішите створити з їх допомогою новий інтерфейс користувача. Чим менше нового коду вам потрібно, тим менша ймовірність виникнення нових помилок. До того ж ви знаєте ваші компоненти. Ви вже використовували та тестували їх при роботі над реальним проектом, а отже, при виникненні помилок зможете передбачити причину їх появи [4].

Якщо взяти таку потужну платформу, як Amazon, для якої затримка в 100мс коштує 1% продаж, то ми можемо зрозуміти, що найчастіше швидкість – це дуже важливий параметр, який треба брати до уваги. [5]

Висновки. Компонентно-орієнтований підхід, можливість легко змінювати наявні компоненти і перевикористовувати код перетворюють React розробку в безперервний процес поліпшення. Компоненти, створені під час роботи над тим чи іншим проектом, немає додаткових залежностей. Таким чином, ніщо не заважає використовувати їх знов і знов у проектах різного типу. Весь попередній досвід може бути легко застосований при роботі над новим сайтом або навіть при створенні мобільного додатка. Використовуючи передові можливості, такі як Virtual DOM або ізоморфний JavaScript, розробники React можуть з високою швидкістю створювати високопродуктивні програми, незважаючи на рівень їх складності. Можливість з легкістю заново використовувати вже наявний код підвищує швидкість розробки, спрощує

процес тестування, і, як наслідок, знижує витрати. Той факт, що ця бібліотека розробляється та підтримується висококваліфікованими розробниками та набирає все більшої популярності з кожним роком, дає підстави сподіватися, що тенденція до подальших покращень продовжиться.

ПЕРЕЛІК ПОСИЛАНЬ

1. Herbert D., What is React.js (Uses, Examples, & More), 2022 [Електронний ресурс]. URL: <https://blog.hubspot.com/website/react-js> (дата звернення: 27.10.2022).
2. Adhikary Yu., Isomorphic Implementation of React, 2019 [Електронний ресурс]. URL: <https://yudhajitadhikary.medium.com/isomorphic-implementation-of-react-fa6e129c246f> (дата звернення: 27.10.2022).
3. Ravichandran A., React Virtual DOM Explained in Simple English, 2019 [Електронний ресурс]. URL: <https://adhithiravi.medium.com/react-virtual-dom-explained-in-simple-english-fc2d0b277bc5> (дата звернення: 27.10.2022).
4. Sathananthan S., How To Reuse React Components, 2021 [Електронний ресурс]. URL: <https://medium.com/codezillas/how-to-reuse-react-components-851ffcc68a9c> (дата звернення: 27.10.2022).
5. Amazon study: Every 100ms in Added Page Load Time Cost 1% in Revenue, 2021 [Електронний ресурс]. URL: <https://www.contentkingapp.com/academy/page-speed-resources/faq/amazon-page-speed-study/> (дата звернення: 27.10.2022).

УДК 004.415.3:681.6

В. Футтерлинг², К.С. Родна¹, М.В. Ларикова¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

²Haas Schleifmaschinen GmbH, Троссінген, Німеччина

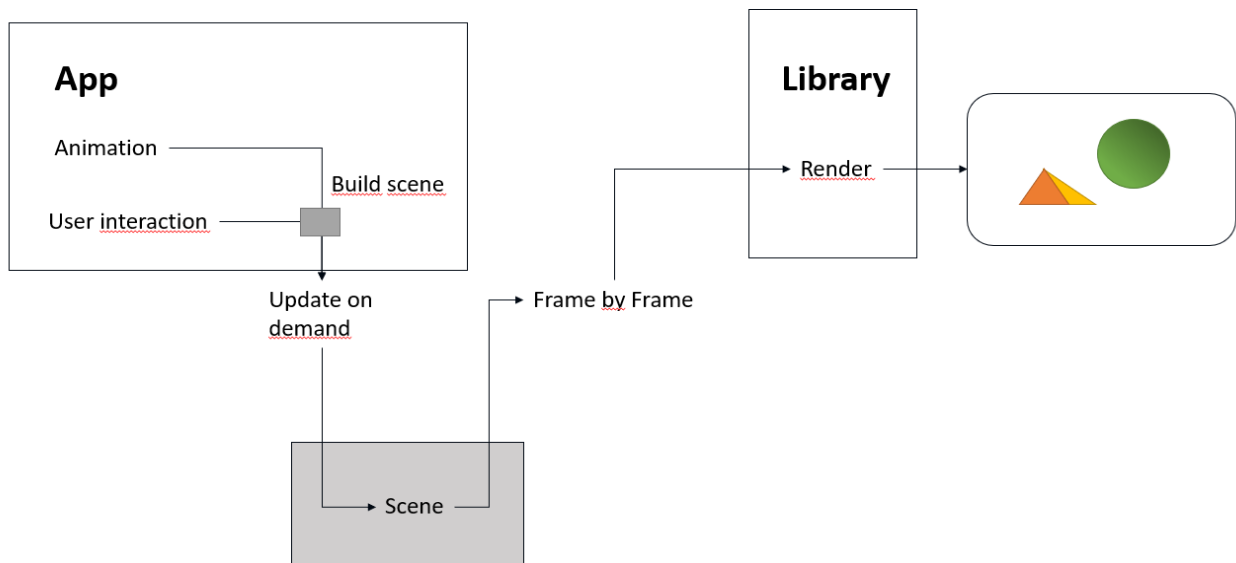
IMGUI. DEAR IMGUI: LIBRARY FOR APPS IN C++

Abstract. The article considers the creation of a graphical user interface for programs that process information in real time frame by frame. Practical implementation based on the ImGui pattern.

Keywords: *rendering, graphical user interface, immediate display mode, design pattern, shaders.*

Introduction. Game developers are often encountering such difficulties as finding the best approach for application design. The main problem is that application should be able to interact not only with game objects and models in real time, but also with GUI. Since some data can be changed on rendering of each frame, these changes should be displayed immediately on user side.

Main Content. To solve this problem, developers use immediate mode graphics approach for designing GUI. ImGui (also known as immediate mode graphics user interface) is a graphical user interface design pattern that requires to use external library with should follow certain standard.



Pic. 1. ImGui pattern structure

Among them, the most important are sync with game scene and graphic, be overlaid on scene and have ability for customization all basic elements abided by whole game design. That's why, such GUI should be much more flexible and should have frame-by-frame render mode. All elements of such GUI must be drawn by primitives with API and widget hierarchy – every child of widget have relative positioning by default. Same thing for positioning widgets in window space. Besides that, they also should have functionality with layers. However, you should remember about constant workload on the CPU. That's why all such patterns use shaders that are loaded on GPU. All the calculation and rendering are running in parallel mode. Which means that rendering each element (pixel by pixel) and rebuilding scene with cleaning vertex buffers doesn't affect CPU performance.

What are shaders? If trying to explain simple, they are small (in most cases) programs that gives commands for GPU how every element should be displayed. There are a lot of types of them, but for GUI the most common are pixel and vertex shaders, because most GUI elements are simple and don't use complicated graphics to render interface. The first one used for adding colors, so you can easily apply some filter to texture. For example, creating bloom or blur effect for 2D objects (images or screenshots). The second one is for positioning objects inside the scene. By default, the scene is divided with axes. Their max values bound to one – left top edge is (-1; 1) and right bottom is (1; -1).

The most popular open library that implements ImGui approach is Dear ImGui that written in C++. It uses several graphics APIs and additionally supports different languages and back-ends. It was presented with Allegro 5, DirectX, GLFW, iOS, Marmalade, Metal, OpenGL, SDL 2 and Vulkan APIs, so developers should not be worried about developing their applications under different platforms.



Pic. 2. Dear ImGui debugging possibilities demonstration

“Dear ImGui is particularly suited to integration in games engine (for tooling), real-time 3D applications, full screen applications, embedded applications, or any applications on consoles platforms where operating system features are non-standard.”

But from mine personal experience with this library, I can say that it is not so flexible for creating complected and customizable GUI. That’s why, you need to rewrite library methods for rendering or create a new one by copy-pasting. By the way, this library needs additional tools for rendering such things like reading files or svg-paths directly from file. Moreover, there are not presented default methods for blurring or creating shadow. Thereby, you need to write your own shaders using your API.

Conclusion. But this library has a lot of advantages. For example, it is envisaging optimization, have a lot of functions and elements for creating simple interface (has demo windows that shows all its power and demonstrate all elements).

Also, it has gamepad and keyboard controls, and this library can be easily integrated. But the coolest things are that it is provided with multiplatform support and fantastic debugging features. So, as for me Dear ImGui one of the best opensource solutions for game GUI development.

REFERENCES

1. "Realistic Lighting in Justice with Mesh Shading". NVIDIA Developer Blog. May 21, 2021.
2. "GLSL Tutorial – Vertex Shader". June 9, 2011.
3. "GLSL Tutorial – Fragment Shader". June 9, 2011
4. Cornut, Omar. "ocornut/imgui: Dear ImGui: Bloat-free Immediate Mode Graphical User interface for C++ with minimal dependencies". December 22, 2019.
5. Radich, Quinn. "Retained Mode Versus Immediate Mode". Win32 apps. Microsoft. December 21, 2019.
6. "Sample IMGUI implementation and demo program". silverspaceship.com. June 15, 2021.

РОЗДІЛ 2

ЗАСТОСУВАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ У СФЕРІ ОСВІТИ, НАУКИ І УПРАВЛІННЯ ВИРОБНИЦТВОМ

УДК 004.9: 528.854

В.В. Гнатушенко¹, Д.Д. Грищак¹, О.С. Шевцова¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

РОЗПІЗНАВАННЯ ЗЕЛЕНИХ НАСАДЖЕНЬ ІЗ ЗАСТОСУВАННЯМ ГЕОІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Анотація. У статті розглядається можливість застосування ГІС-технологій для оцінки площі зелених насаджень. Озеленені території є важливим індикатором якості міського середовища. Геоінформаційні системи дозволяють створювати карти, бази даних, підтримувати актуальну інформацію. У роботі наведено приклад використання даних аерофотозйомки для розпізнавання й розрахунку площі озеленення. Використання запропонованого підходу дозволяє значно скоротити час та трудовитрати для знаходження показників озеленення міського середовища.

Ключові слова: розпізнавання, зображення, рослинність, геоінформаційні системи, дистанційне зондування.

Вступ. Організація зеленого господарства міста вимагає розумного підходу як до вибору, так і розміщення зелених насаджень і, звичайно, їх обліку. При цьому слід уточнити, що поняття обліку зелених насаджень має на увазі дві різні процедури. Це, по-перше, інвентаризація дерев, встановлення їх якісного стану. По-друге, це оцінка площі, що займають всі зелені насадження, що включає не тільки дерева і чагарники, а й газони і т.п. На відміну від розвинених країн світу, в Україні не впроваджено концепції зеленого простору, зеленої інфраструктури, відсутнє нормування загальної кількості міських зелених насаджень [1]. Однак оцінка рівня комфортності об'єктів озеленення урбанізованих територій у наші дні є важливим завданням, особливо великих мегаполісів. А практичні результати зазначеної оцінки мають використовуватися в областях, пов'язаних із територіальним плануванням, плануванням території, кадастровою оцінкою земель населених пунктів, індивідуальною оцінкою об'єктів нерухомості, а також при ухваленні різних управлінських рішень.

Процес впорядкування зелених насаджень, як система заходів щодо вирішення вищевказаних проблем, набуває важливого значення для зеленого будівництва. Реалізація таких заходів можлива лише за умови наявності достовірної, оперативної та актуальної інформації про кількісний і якісний стан зелених насаджень населених пунктів. Проте, цей процес, як і будь-яка сучасна методика отримання, обробки, обліку, аналізу та збереження даних, не може функціонувати без залучення переваг новітніх технологій. Отже розробка

технології інвентаризації зелених насаджень території з використанням сучасних ГІС-технологій є актуальною задачею сьогодення [2].

У представленому дослідженні здійснено спробу відпрацювати методику оцінки площ озеленення міста на основі отриманих даних аерофотозйомки з безпілотних літальних апаратів (БПЛА).

Основна частина. З появою та розвитком цифрових технологій процес обліку зелених насаджень удосконалюється. На основі таксації створюються атрибутивні бази даних, цифрові карти. Все це дає широкі можливості статистичної обробки інформації, складання вибірок, аналізу стану деревостою, планування заходів щодо посадок насаджень. Найбільш складним є завдання обліку площ насаджень обмеженого користування, що знаходяться на прибудинкових територіях, при школах, дитячих садках, медичних, освітніх та інших закладах. Аерофотозйомка з літаків або БПЛА дозволяє отримати зображення більш високої роздільної здатності (до 5 см) та оцінити не лише кількість, а й стан насаджень. При цьому можливе проведення лазерного сканування, що дає більш точні просторові зображення. У цьому дослідженні основним завданням роботи було відпрацювання методики визначення площ озеленення.

Експериментальні дослідження проводилися на матеріалах з БПЛА декількох ділянок земної поверхні. Автори використовували власну програмну реалізацію розробленого алгоритму з використанням на окремих етапах результатів попередніх досліджень [3, 4].

Методика досліджень.

В роботі запропоновано підхід, що містить наступні основні етапи:

1. Завантаження вихідного мультиканального знімка. Приклад візуалізації фрагменту RGB-синтезованого зображення подано на рис.1.

2. Виділення рослинності з використанням ІЧ-діапазону. Бінаризований результат цього етапу представлено на рис 2.

3. Виділення газонів з використанням навчальної вибірки (рис. 3).

4. Семантична сегментація (рис. 4) [4].

5. Використання маски рослинності та подальше просіювання замаскованого зображення (рис.5).

6. Побудова вектору та подальше накладення його на растр (рис.6).

7. Підрахунок площ зелених насаджень.



Рис. 1. Вихідне RGB-зображення

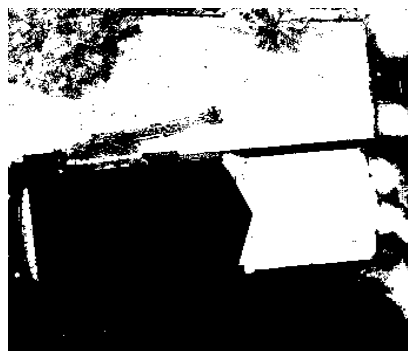


Рис. 2. Виділення рослинності з використанням ІЧ-діапазону



Рис. 3. Виділення газонів з використанням навчальної вибірки



Рис. 4. Результат сегментації

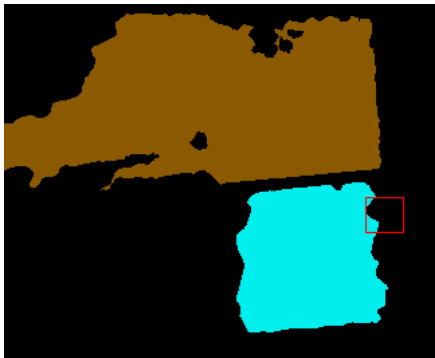


Рис. 5. Результат просіювання замаскованого зображення

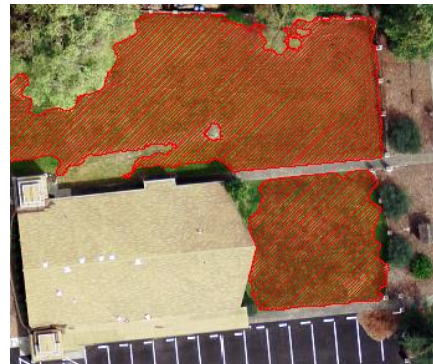


Рис. 6. Накладення вектора на растр

Висновки. В результаті проведених досліджень було розроблено метод розпізнавання рослинності та подальшої оцінки площ озеленення міста на основі отриманих даних аерофотозйомки з безпілотних літальних апаратів. Верифікація на місцевості підтвердила працездатність та високу точність запропонованого методу.

ПЕРЕЛІК ПОСИЛАНЬ

1. Yukhnovskyi V.Yu., Zibtseva O.V. Normalization of green space as a component of ecological stability of a town. *Journal of Forest Science*. 2019. № 65 (11). P. 428–437.
2. Пілічева М.О., Маслій Л.О., Анопрієнко Т.В. Технологія виконання геодезичних робіт при інвентаризації зелених насаджень території з використанням безпілотних апаратів. *Комунальне господарство міст*, 2022, том 3, випуск 170, с.263-270.
3. Mozgovoy D.K., Hnatushenko V.V. and Vasyliiev V.V. Automated recognition of vegetation and water bodies on the territory of megacities in satellite images of visible and IR bands, *ISPRS Ann. Photogramm. Remote Sens. Spatial Inf. Sci.*, IV-3, 167–172, <https://doi.org/10.5194/isprs-annals-IV-3-167-2018>, 2018.
4. Hnatushenko Volodymyr, Zhernovyi Vadym, Udovik Iryna, Shevtsova Olga. Intelligent System for Building Separation on a Semantically Segmented Map. 2 International Workshop on Intelligent Information Technologies & Systems of Information Security (IntelITSIS-2021) Khmelnytskyi, Ukraine. <http://ceur-ws.org/Vol-2853/keynote1.pdf>

Д.В. Субач¹, І.А. Ткаченко¹, Г.М. Коротенко¹, В.В. Гнатушенко¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

РОЗРОБКА ІНФОРМАЦІЙНОЇ ТЕХНОЛОГІЇ СТВОРЕННЯ ТА ВИКОРИСТАННЯ КОМПОНЕНТІВ NFT МАРКЕТПЛЕЙСА

Анотація. Розроблено децентралізований маркетплейс, який призначений для карбування унікальних токенів NFT, виставлення їх до маркетплейсу, а також доступу до інформації транзакцій та власника токенів.

Ключові слова: інформаційні технології, блокчейн, децентралізований додаток, non-fungible token, NFT маркетплейс, смарт-контракти, крипто-гаманець, Truffle Suite Studio, Ganache, Solidity, Typescript.

Вступ. Авторами проведено дослідження ефективності застосування інформаційних технологій і відповідних програмних засобів підтримки інноваційних технологій блокчейну та незмінних токенів, які продовжують розвиватися на базі розширення галузей їхнього застосування. Було розроблено децентралізований маркетплейс, який призначений для карбування унікальних токенів NFT, виставлення їх до маркетплейсу, а також доступу до інформації щодо проведених транзакцій для власників токенів у вигляді електронних сертифікатів права власності щодо фізичних або віртуальних активів.

Постановка задачі. Для досягнення поставленої мети в роботі сформовані і потребують вирішення такі завдання:

- дослідити принципи побудови та роботи децентралізованих додатків та засобів для їх побудови;
- розробити програмне забезпечення децентралізованого маркетплейсу на базі комплексу обраних інструментів;
- забезпечити повноцінне функціонування заявлених функцій, таких як: карбування унікальних токенів NFT; можливість виставлення їх до маркетплейсу; доступ до повної інформації токenu.

Основний зміст роботи полягає у популяризації та описі особливостей роботи сучасної технології блокчейну, на прикладі розробки інформаційної технології NFT маркетплейса з використанням криптовалют та незмінних токенів.

Інформаційна технологія NFT маркетплейса розроблювалася за допомогою наступних інструментів:

- об'єктно-орієнтована мова програмування Solidity;
- типізована мова програмування TypeScript;
- служба керування мультимедійними файлами NFT Pinata;
- середовище розробки Truffle;
- персональний блокчейн Ganache;
- колекція інтерфейсних бібліотек Drizzle.

Було проведено аналіз існуючих маркетплейсів для вибору актуальної платформи блокчейну для реалізації поставленої задачі. Аналіз показав, що більшість ринків NFT використовують Ethereum. Ethereum містить тисячі колекцій NFT. У мережі Ethereum токени створюються з використанням стандарту ERC-721, який зберігає метадані NFT у однойменному блокчейні. Предметом розробки є децентралізований маркетплейс, який призначений для карбування унікальних токенів NFT, їх виставлення до маркетплейсу, а також доступ до інформації токенів.

Обрано локальний блокчейн Ganache, який є симулятором блокчейну від Ethereum на основі Ethereum Virtual Machine. Інструмент для налаштування власного локального блокчейну Ethereum, який можливо використовувати для розгортання смарт-контрактів перед їх запуском в автентичному ланцюжку, розробки децентралізованих додатків та тестування. Ganache є компонентом структури Truffle Suite разом з іншими компонентами Truffle і Drizzle [1].

В якості середовища для розробки програми запропоновано використати Truffle – середовище розробки, конвеєр активів і платформа тестування екосистеми Truffle Suite. Фреймворк Truffle використовує віртуальну машину Ethereum (EVM), для забезпечення гнучкої розробки смарт-контрактів [2].

Використано колекцію інтерфейсних бібліотек Drizzle. Ядро Drizzle базується на основі сховища Redux. Інструментів розробки Redux роблять написання інтерфейсів децентралізованих додатків простішим і передбачуванішим. Drizzle покладається на магазин Redux як на основний елемент для гнучкого доступу до багатьох функціональних інструментів розробки, пов'язаних із Redux. Це допомагає в синхронізації даних контрактів, даних транзакцій і багатьох інших факторів [3].

Для того, щоб забезпечити розміщення та зберігання токенів у блокчейні Ganache, використано платформу керування мультимедійними файлами NFT – Pinata на основі служби IPFS. Inter Planetary File System (IPFS) є протоколом, який дозволяє розміщувати занадто великий вміст для блокчейнів у децентралізованій одноранговій мережі, використовуючи криптографію, щоб гарантувати, що вміст неможливо змінити [4].

Для розробки децентралізованого маркетплейсу, була використана об'єктно-орієнтована мова програмування високого рівня Solidity, яка призначена для реалізації смарт-контрактів та роботи з віртуальною машиною Ethereum Virtual Machine (EVM). Solidity діє як інструмент для створення коду машинного рівня та його компіляції на віртуальній машині Ethereum. Програми транслюються в байткод EVM. Solidity дозволяє вам «відображати» структури даних за допомогою переліків, операторів і хеш-значень, щоб повертати значення, що зберігаються в певних місцях зберігання [5].

Для розробки інтерфейсу додатку та роботи з фреймворками для JavaScript - React.js та Next.js, прийнято рішення використати типізовану мову програмування, яка базується на JavaScript під назвою TypeScript. TypeScript розширює JavaScript і надає різноманітні інші функції, такі як інтерфейси, псевдоніми типів, абстрактні класи, перевантаження функцій, кортежі,

генерики тощо. Він розроблений для розробки великих додатків і компілюється на JavaScript. TypeScript пропонує всі функції JavaScript і додатковий рівень на додаток до них: систему типів TypeScript [6].

Для вирішення поставленої задачі розроблено додаток у вигляді децентралізованої програми, яка існує і працює у блокчейні на платформі Ethereum. Схема архітектури децентралізованих додатків у середовищі розробки Truffle Suite представлена на рис. 1.

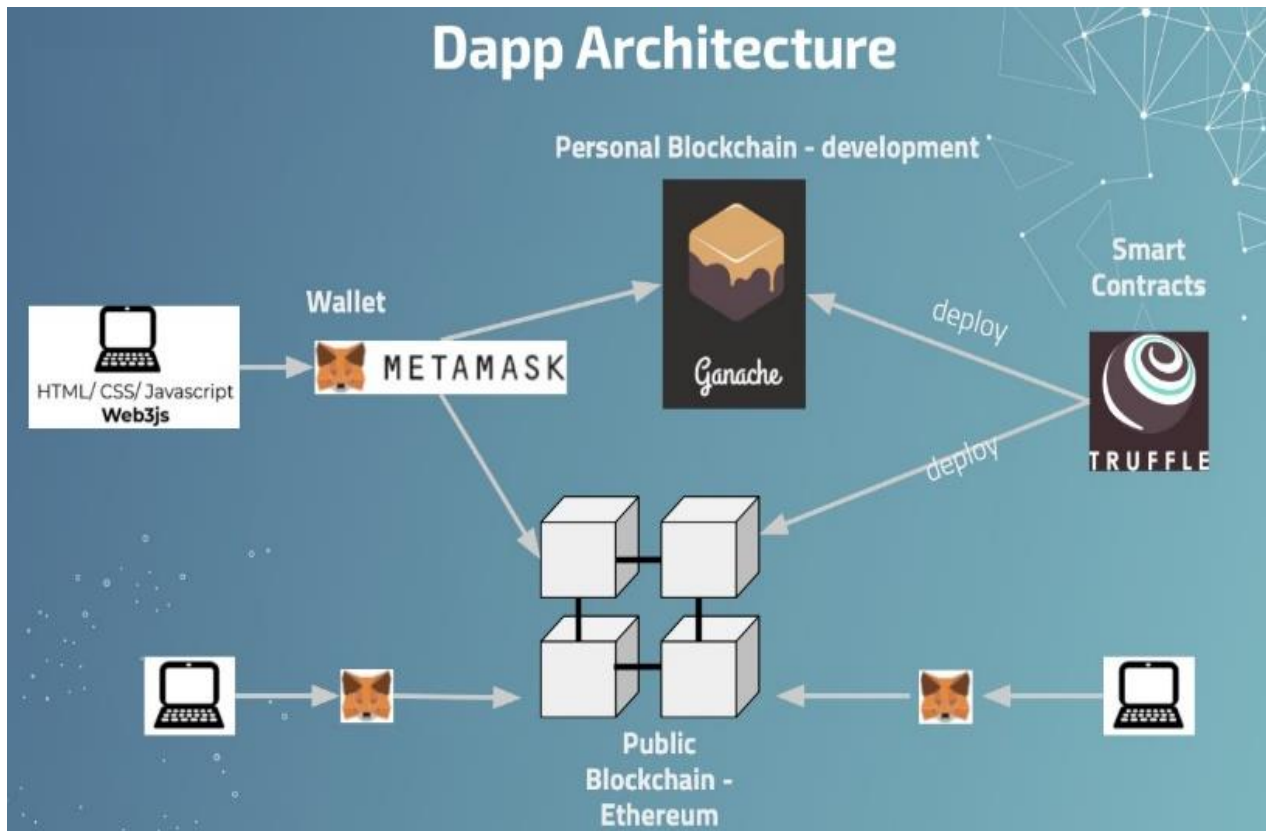


Рис. 1. Архітектура децентралізованих додатків у середовищі розробки Truffle Suite

Результатом розробки є децентралізований додаток – маркетплейс для карбування незамінних токенів за стандартом ERC721, на мові програмування Solidity з використанням пакету розробника Truffle Suite Studio.

Наукова новизна визначається тим, що вперше розроблено і описано інформаційну технологію розробки децентралізованого маркетплейсу для створення власних NFT, який побудовано на блокчейні Ganache та реалізовано зберігання метаданих токенів у сервісі Pinata.

Висновки. Сформовано ідею проекту та створена структура децентралізованого додатку на мові програмування Solidity з використанням локального блокчейну Ganache; інструментами для розробки комплексу розробника Truffle Suite Studio та служба керування мультимедійними файлами NFT – Pinata. Було розроблено інформаційну технологію створення децентралізованого маркетплейсу, який призначений для карбування

унікальних токенів NFT, виставлення їх до маркетплейсу, а також доступу до інформації щодо проведених транзакцій для власників токенів у вигляді електронних сертифікатів права власності щодо фізичних або віртуальних активів.

ПЕРЕЛІК ПОСИЛАНЬ

1. What is Ganache? (Електрон. Ресурс) / Спосіб доступу: URL: <https://trufflesuite.com/docs/ganache/>
2. What is Truffle? (Електрон. Ресурс) / Спосіб доступу: URL: <https://trufflesuite.com/docs/truffle/>
3. Drizzle overview. (Електрон. Ресурс) / Спосіб доступу: URL: <https://trufflesuite.com/docs/drizzle/>
4. What is IPFS? (Електрон. Ресурс) / Спосіб доступу: URL: <https://docs.pinata.cloud/what-can-i-learn-here/what-is-pinata/what-is-ipfs>
5. Solidity. (Електрон. Ресурс) / Спосіб доступу: URL: <https://docs.soliditylang.org/en/v0.8.17/>
6. What is TypeScript and why should you use it? (Електрон. Ресурс) / Спосіб доступу: URL: <https://www.contentful.com/blog/what-is-typescript-and-why-should-you-use-it/>

УДК 004.9

В.І. Олевський¹, В.С. Крутько¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

РОЗРОБКА ПІДСИСТЕМИ ВІЗУАЛЬНОЇ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧА НА ОСНОВІ ВИКОРИСТАННЯ БІБЛІОТЕК OPENCV TA FACE RECOGNITION

Анотація. З використанням спеціалізованих бібліотек OpenCV та Face Recognition створена підсистема розпізнавання обличчя користувача для додаткового захисту інформаційних систем. При цьому досягається можливість враховувати особливості інформаційної системи та індивідуальні налаштування процедур, що дає змогу уникнути недоліків існуючих систем.

Ключові слова: інформаційні системи, захист від несанкціонованого доступу, методи візуальної аутентифікації, система розпізнавання обличчя, OpenCV, Face Recognition.

Вступ. Сучасні інформаційні системи широко використовуються у виробництві і суспільному житті, зберігаючи при цьому критично важливу інформацію користувачів. Тому питання забезпечення захисту таких систем від несанкціонованого доступу є край важливим.

Традиційні методи ідентифікації, засновані на використанні логіну і паролю в цілому вирішують цю проблему, але є вразливими до зламу та методів соціальної інженерії. У зв'язку з значним поширенням зловмисних атак, особливо в період війни, розробники використовують для додаткового захисту інформаційних систем методи візуальної аутентифікації користувачів [1]. Такі методи більш надійні, тому що є невід'ємними від особистості користувача.

Існують декілька програмних систем, що реалізують візуальну ідентифікацію. Наприклад, Face ID – це об'ємно-просторовий сканер обличчя, який працює за принципом Time-of-flight розпізнаючи не просто плоский портрет людини, а його об'ємну модель [2]. Face ID є відгалуженням технології LiDAR, вона самонавчається, запам'ятовує зміни в особі за допомогою нейронних мереж у процесорі смартфона. Але ця має повільну швидкість роботи, не є надійною в плані захисту даних; не розпізнає дітей віком до 12 років, не розпізнає користувача поки він не подивиться на фронтальну камеру і т.д. Користувачі зауважують на нестабільну роботу цієї технології, тому вона використовується більше для простоти аутентифікації, ніж для надійного захисту даних.

Ще однією такою системою є DeepFace – система розпізнавання обличчя з глибоким навчанням, створена дослідницькою групою Facebook [3]. Вона здатна розпізнавати обличчя на цифрових фотографіях, але не здатна відокремлювати особу на фото.

Сервіс Amazon для виявлення та розпізнавання зображень на основі нейронних мереж з глибоким навчанням. Недоліками цієї технології є те, що при використанні зображень з роздільною здатністю QVGA (320x240) і нижче значно підвищується ймовірність помилок при розпізнаванні осіб.

Постановка задачі. Метою роботи є створення підсистеми розпізнавання обличчя користувача для додаткового захисту інформаційних систем з використанням спеціалізованих бібліотек OpenCV та Face Recognition. Для досягнення поставленої мети в роботі сформовані і вирішені такі завдання:

- викласти принципи захисту інформаційних систем з використанням спеціалізованих бібліотек OpenCV та Face Recognition;
- визначити основні необхідні компоненти системи;
- спроектувати та розробити програмну частину підсистеми розпізнавання обличчя;
- зробити експериментальні випробування розробленої системи і зробити висновки щодо її ефективності.

Основний зміст роботи. OpenCV – це бібліотека комп'ютерного зору з відкритим кодом та досить широким функціоналом, яка призначена для аналізу, класифікації та обробки зображень. Вона надає засоби для обробки і аналізу вмісту зображень, у тому числі розпізнавання об'єктів на фотографіях (наприклад, осіб і фігур людей, тексту тощо), відстежування руху об'єктів, перетворення зображень і застосування методів машинного навчання. Бібліотека використовується для різних мов програмування (C++, Python, Java, Ruby, Matlab та ін.) і може вільно використовуватися в академічних та комерційних цілях.

Face Recognition – це бібліотека мови Python, що використовується для розпізнавання обличчя. Вона відрізняється високою точністю, дуже зручна у використанні, її легко розгорнути на сервері навіть без дорогого GPU і вона має досить прості вимоги.

Завданням даної роботи є створення підсистемі візуальної аутентифікації користувача веб додатку на основі використання бібліотек OpenCV та Face Recognition. Для створення веб додатку використовувався найбільш поширений фреймворк Django.

Схема роботи розробленої підсистемі приведена на рис 1.

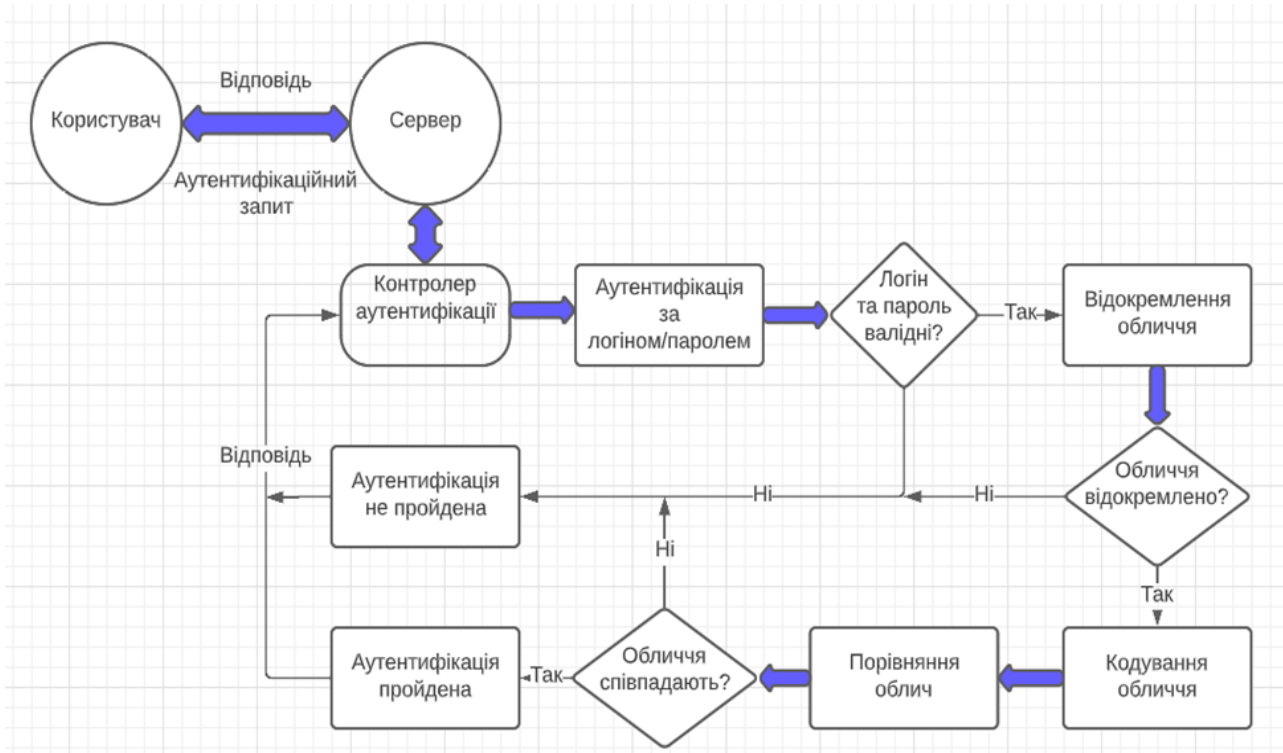


Рис. 1. Схема роботи підсистемі аутентифікації користувача

Робота підсистемі здійснюється наступним чином:

1. Користувач за допомогою візуального інтерфейсу формує аутентифікаційний запит. Спочатку користувач заповнює поля "login" та "password", далі, якщо в нього активована функція візуальної аутентифікації, за допомогою веб-камери він робить знімок свого обличчя. Після того як всі поля заповнені, запит відправляється до серверу.

2. Сервер отримавши запит, перенаправляє його до підсистемі аутентифікації.

3. Першим етапом аутентифікації є стандартна аутентифікація за допомогою полів login_field та password. На цьому етапі система надсилає запит до бази даних (БД), для отримання даних користувача за полем login_field. Якщо користувач не знайдений, повертається помилка аутентифікації. В іншому випадку, порівнюється пароль надісланий у запиті, з паролем цього користувача, збереженим в БД. Якщо паролі співпадають, відбувається перехід до другого етапу аутентифікації - підсистемі візуальної аутентифікації, до якої передається запит, та вже отримані дані користувача з БД.

4. Підсистема отримує зображення, з поля "image", та починає процес пошуку обличчя на знімку за допомогою функцій "imread" та "face_locations" бібліотек Face Recognition та OpenCV. Якщо обличчя не відокремлено, повертається помилка аутентифікації. Якщо обличчя знайдено, починається обробка обличчя.

5. Використовуючи функцію "face_encodings" бібліотеки Face Recognition обличчя кодується.

6. За допомогою функції "compare_faces" бібліотеки Face Recognition закодоване обличчя порівнюється з обличчям користувача в БД. Якщо обличчя співпадають, то аутентифікація успішно завершується; якщо ні - повертається помилка аутентифікації.

Ініціалізація системи візуальної аутентифікації:

1. За допомогою візуального інтерфейсу, користувач реєструється в системі. Для цього він заповнює поля "login" та "password", та натискає кнопку "Зареєструватись" в візуальному інтерфейсу. Після того, як користувач відправив запит на реєстрацію, система зберігає в БД вказані логін та пароль. Реєстрація завершується.

2. Наступним кроком є аутентифікація користувача в системі за допомогою логіну та паролю, які були вказані при реєстрації. Користувач формує аутентифікаційний запит за допомогою візуального інтерфейсу - заповнює поля "login" та "password" та відправляє запит до серверу. Якщо вказані дані, відповідають даним користувача в БД, аутентифікація в системі завершується, користувач отримує доступ до інтерфейсу системи.

3. За допомогою інтерфейсу користувач переходить на сторінку налаштування свого профілю та вмикає функцію візуальної аутентифікації.

4. Після застосування функції візуальної аутентифікації, користувач переходить до інтерфейсу налаштування аутентифікації. За допомогою інтерфейсу він додає від 3-х фотографій свого обличчя. При цьому користувач має можливість завантажити їх або зі своєї файлової системи, або зробити і додати їх за допомогою веб-камери. Після того, як користувач додав свої світлини, він відправляє запит на завершення налаштування.

5. Як тільки запит на завершення налаштування надійшов до серверу, система починає обробляти цей запит. Система отримує надіслані зображення, відокремлює на них обличчя та порівнює їх між собою. Якщо після порівняння обличчя на фотографіях відсоток їх схожості менше 70%, то система вважає ці обличчя різними та надсилає користувачу відповідь, в якій просить користувача завантажити інші зображення. При цьому налаштування системи візуальної аутентифікації не є закінченим і вона не буде застосовуватися при аутентифікації в систему. У випадку, коли відсоток схожості обличчя більше 70%, система зберігає їх в БД, разом з іншими даними цього користувача. Налаштування системи вважається завершеним, користувачу надходить відповідь про успішне завершення налаштування.

б. Після успішного налаштування, візуальна аутентифікація для користувача активована, при наступній аутентифікації в систему, вона буде застосовуватися.

Система візуальної аутентифікації була протестована з використанням фото користувача, зробленого в різних умовах:

- з гарним освітленням;
- з поганим освітленням;
- з аксесуарами;
- з нестандартним ракурсом.

Загалом було відправлено 80 аутентифікаційних запитів лише 8 з яких були невдалими. Виходячи з даної статистики можна зробити вивід, що точність системи - 90%, тому дану систему можна вважати високоточною та надійною. Також, беручи до уваги результати окремих тестів, можна сказати, що на точність системи не впливає освітленість фото, наявність аксесуарів та ракурс обличчя на фото.

Наукова новизна розробки складається в удосконаленні системи візуальної аутентифікації за рахунок використання новітніх бібліотек OpenCV та Face Recognition та отриманні подальшого розвитку інформаційних система з візуальною аутентифікацією шляхом можливості враховувати особливості інформаційної системи та індивідуальні налаштування процедур.

Висновки. В результаті була спроектована ефективна інформаційна система з візуальною аутентифікацією. Система була протестована для різних умов використання; випробовування продемонстрували її стабільну роботу і надійність. Система має потенціал розвитку, наприклад, за рахунок заміни нейронної мережі, яка застосовується для розпізнавання та порівняння обличь. В порівнянні зі стандартною системою аутентифікації за логіном та паролем, розроблена система є більш надійною та зручною в використанні.

ПЕРЕЛІК ПОСИЛАНЬ

1. Євсєєв С. П. Дослідження загроз методів двофакторної автентифікації / С. П. Євсєєв, О. Г. Король // Вісник Національного університету "Львівська політехніка". – 2014. – № 806 : Комп'ютерні системи та мережі. – С. 62–71.
2. Face ID: [Electronic resource] – Access mode: https://uk.wikipedia.org/wiki/Face_ID
3. Face Detection Algorithms and Techniques [Electronic resource] – Access mode: <https://facedetection.com/algorithms/>
4. Rosebrock Adrian Face recognition with OpenCV, Python, and deep learning [Electronic resource] – Access mode: <https://pyimagesearch.com/2018/06/18/face-recognition-with-opencv-python-and-deep-learning/>
5. OpenCV: OpenCV Tutorials [Electronic resource] – Access mode: <https://docs.opencv.org/2.4/doc/tutorials/tutorials.html>

РОЗРОБКА СИСТЕМИ КЕРУВАННЯ КОМБІНОВАНОГО ВЕЛОТРЕНАЖЕРА ДЛЯ РЕАБІЛІТАЦІЇ ВІЙСКОВИХ СЛУЖБОВЦІВ

Анотація. Розроблено систему керування велотренажером для фізичної терапії та програмний додаток для платформи Android, завдяки якому відбувається налаштування параметрів тренування користувачем.

Ключові слова: комбінований велотренажер, контролер системи керування, електронна схема, датчик, параметри тренування, мобільний застосунок.

Вступ. Фізична терапія із застосуванням різного виду тренажерів допомагає відновити функціональний стан організму [1, 2]. Авторами [3] було розроблено комбінований велотренажер, який поєднує функціональні можливості стаціонарного велотренажеру та велосипеда, що дозволяє отримати точне, фіксоване і контрольоване навантаження під час їзди по ділянках місцевості із різними типами покриття, та змінним рельєфом. Крім того, на сьогоднішній день було розроблено лише концепт комбінованого велотренажера. Тому, залишається актуальним розробка системи керування комбінованим велотренажером.

Постановка задачі. Для досягнення поставленої мети в роботі сформовані і вирішені такі завдання:

- розробити систему керування комбінованим велотренажером;
- обґрунтувати компоненти системи керування комбінованим велотренажером;
- розробити електронну схему контролеру системи керування комбінованим велотренажером;
- розробити програмний код контролера системи керування комбінованим велотренажером;
- розробити мобільний додаток для користувача комбінованого велотренажера, який забезпечує управління тренажером, з телефону пацієнту, дозволяє ввести параметри тренування та відслідковувати прогрес та статистику тренувань лікарю.

Основний зміст роботи полягає в аналізі існуючих систем керування велотренажерів, та розробці й оптимізації системи керування для комбінованого велотренажера.

Система керування комбінованого велотренажера складається із таких компонентів:

- контролер, побудований на мікроконтролері arduino mini;
- датчик натиску на педалі;
- датчик швидкості обертання шатуна педалей;

- датчик кута нахилу;
- дисплейний модуль;
- інтернет модуль;
- контролер прямопривідного мотор-колеса;
- акумуляторна батарея.

Було проведено аналіз датчиків для реалізації поставленої задачі. Аналіз показав, що у лінійці мікро-контролерів Arduino існують такі плати як Arduino Uno, Arduino mini, Arduino micro. Всі вони побудовані на одній й той самій мікросхемі ATmega328P та відрізняються кількістю пінів та вбудованої пам'яті. Перевагу було надано саме платі Arduino UNO зважаючи на оптимальну кількість логічних входів/виходів та оптимальну кількість вбудованої пам'яті.

Для датчика натиску на педалі було запропоновано використати FSR датчик. FSR датчик – (абр. Force Sensitive Resistor) – датчик резистивного типу, опір якого зменшується під час натиску на його робочу сторону. За формою датчики діляться на круглі, прямокутні та квадратні. Було вирішено використовувати датчики квадратної форми через необхідність великої площі натиску.

Для того, щоб визначити швидкість обертання шатуна педалей було використано датчик магнітного типу PAS. Принцип його роботи полягає у зчитуванні електромагнітного сенсору, який встановлений нерухомо на рамі велотренажера магнітного імпульсу. Датчик складається із двох частин: одна частина кріпиться на місце прикріплення втулки шатуна педалей, інша – кріпиться на вал шатуна педалей на відстані до 5мм до відповідної нерухомої частини. На рухомій частині датчика розташовано до 16 неодимових магнітів по колу. Чим більша кількість магнітів тим більша точність зчитування швидкості обертання шатуна педалей.

Для визначення кута нахилу комбінованого велотренажера було використано датчик MPU-6050 GY-521, який містить вбудований акселерометр й гіроскоп.

Дисплейний модуль у комбінованому велотренажері використовується для можливості керування режимами тренувань та введення потрібного значення навантаження за рекомендаціями лікаря. Через те, що дисплейний модуль використовується як пристрій для виведення інформації користувачу комбінованого велотренажера у вигляді графічної інформації немає суттєвої різниці для розробки системи керування яка модель дисплейного модулю буде використана. Але зважаючи на необхідність тестування роботи нашої системи керування комбінованого велотренажера було вирішено використовувати наявний в нас дисплейний модуль LCD1602.

Для того, щоб віддалено керувати комбінованим велотренажером було використано технологію Wi-Fi для мікроконтролеру Arduino UNO R3–NodeMCU ESP8266, на базі мікроконтролеру ESP8266.

Враховуючи характеристики мотор-колеса та необхідні для нього номінали напруги та струм живлення, було запропоновано використати

акумулятор номінальною напругою у 48В та ємністю у 12 А/год, що у перерахунку у Вт/год становить 576 Вт/год, а саме акумуляторну батарею, що складається із окремих модулів LiFePo4. Ці властивості зумовлені хімічною структурою елементів живлення.

Після проведеного аналізу датчиків та складових компонентів системи керування комбінованого велотренажера було вирішене питання, щодо регулювання навантаження на педалі комбінованого велотренажера. Структурна схема для рекупераційного (електромагнітного) гальмування представлена на рис. 1.

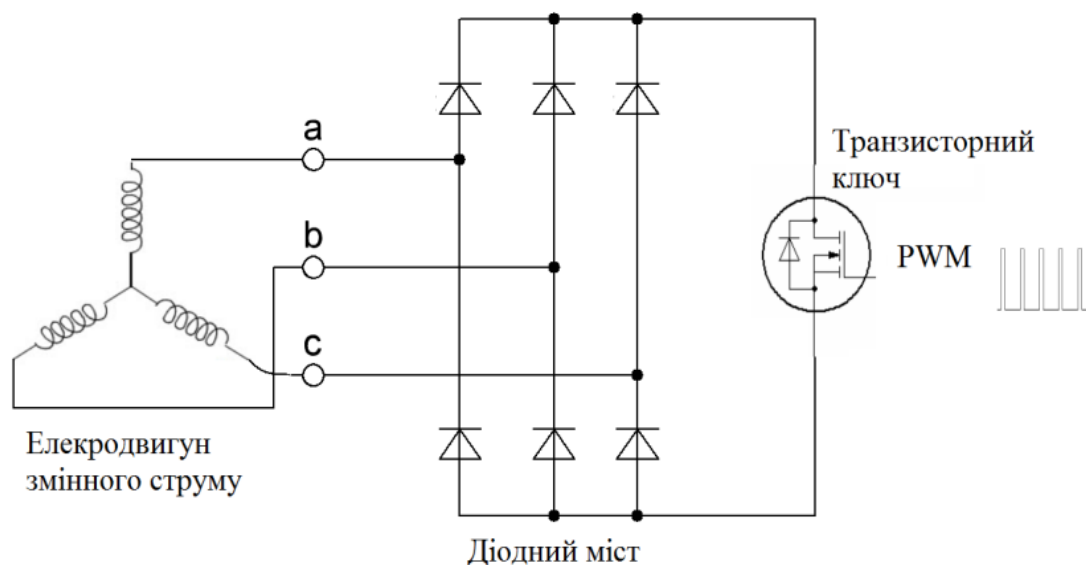


Рис. 1. Електрична схема для електромагнітного гальмування

Далі, для того, щоб керувати всіма компонентами системи керування та об'єднати їх в одне ціле, було розроблено та спроектовано електронну схему контролеру та написано програмний код під мікроконтролер Arduino UNO R3, на мові Arduino C, що є оптимізованої під платформу Arduino мовою C++ із фреймворком Wiring.

Наукова новизна полягає у розробці системи керування та програмного забезпечення, а також в проведенні аналізу схем системи керування велоергометрами та їх оптимізації під конструкцію комбінованого велотренажера, що побудований на базі велосипеда та стаціонарного велоергометра.

Висновки. Було розроблено систему керування комбінованого велотренажера. Було проведено аналіз існуючих схем систем керування, шляхом виявлення їх позитивних та негативних характеристик. Завдяки проведеному аналізу роботи велотренажера, та медичних програм тренувань, було підібрано компоненти системи керування та забезпечено їх взаємозв'язок завдяки контролеру системи керування. Було розроблено програмний код контролеру на мові Arduino C та програмний додаток для платформи Android, який забезпечує взаємозв'язок між комбінованим велотренажером, пацієнтом

та лікарем. Це дає можливість контролювати параметри навантаження на органи людини та надає змогу лікарю віддалено слідкувати за перебігом та статистикою тренування, за потреби вносити зміни до роботи системи.

ПЕРЕЛІК ПОСИЛАНЬ

1. Хоменко І.П. Визначення провідного виду уражень військовослужбовців у сучасних військових конфліктах // І. П. Хоменко , К. В. Гуменюк, С. О. Король, Є. В. Цема, Р. М. Михайлусов , В. П. Майданюк , С. В. Тertiшний , О. М. Попова. – Харківська хірургічна школа. – 2021;2(107), с.128-129.

2. Клапчук В.В., Фетісова В.В. Обґрунтування методики відновлювальних тренувань на велоергометрі висококваліфікованих плавців після гострих респіраторних захворювань // В.В. Клапчук, В.В. Фетісова. – Медичні перспективи. – 2005;10(1), с.95-99.

3. Заболотний К.С., Манін Б.М. Комбінований велотренажер на основі велосипеда *specialized hardrock v 26 2015* як прилад для фізичної терапії і спорту. Проекти фіналістів всеукраїнського конкурсу-захисту науково-дослідницьких робіт учнів-членів Дніпропетровського відділення Малої академії наук України. 2021. с. 50-51.

УДК 004.415.3:681.6

Л.І. Мещеряков¹, А.Л. Ширін¹, О.М. Галушко¹, В.В. Слесарев¹, С.А. Шолойко¹
¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ПОБУДОВА ПРАВИЛЬНОЇ ТОПОЛОГІЧНОЇ СІТКИ ОБ'ЄКТІВ ТА МОДИФІКАТОР RETOPOLOGY В AUTODESK 3DSMAX

Анотація: В роботі наведено реалізацію автоматизованого виправлення топологічної сітки за допомогою модифікатора Retopology у середовищі Autodesk 3ds Max. Розкрито процес створення карти нормалей для побудови нових об'єктів на основі існуючої моделі.

Ключові слова: топологія, топологічна сітка, автоматизоване виправлення, ретопологія, карти нормалей, 3D-модель, модифікатор Retopology.

Вступ. 3D-моделювання, як напрямок, з'явилося ще в минулому столітті. Ще у 1995 році відбулася прем'єра повноцінного 3D мультфільму «Історія іграшок». Розробка 3d-об'єктів, їх анімація, моделювання та реалізація в різних наукових та освітніх проектах, повсякденно використовуються в нашому житті. Але вдосконалення технологій та інструментів щодо створення 3d-моделей, експорту та імпорту між різними системами їх реалізації відбувається щорічно, що дозволяє вирішувати ще більше різних задач та автоматизувати процеси при побудові складних об'єктів. Однією з таких складних графічних систем є Autodesk 3ds Max – програмне забезпечення для 3D-моделювання, анімації та візуалізації під час створення ігор і проектування.

Основний зміст роботи. При створенні кожний об'єкт має свою топологію. Топологія це сітка з полігонів, які і формують 3D-модель. Але, у зв'язку з методикою моделювання, ця сітка може формуватися неправильно, що

приведе до зайвого навантаження системи й неможливості обробки моделі у подальшому (наприклад, в анімації) (рис. 1).

Правильна топологія служить двом цілям:

- правильні деформації під час анімації;
- використання мінімальної кількості полігонів для опису форми.

У першому випадку, полігони повинні розташовуватися на 3D моделі (зазвичай живої істоти) так, щоб при русі кісток або м'язів, було легко повторити реальні опуклості від цих м'язів або кісток. Найпростіший приклад, це необхідність ущільнення сітки на колінах та ліктях. Адже коли моделюються персонажа у Т-позі, то лікті та коліна випрямлені. А коли їх потрібно зігнути на 90 градусів, то вийде, що полігонів не вистачає (якщо модельєр зробив сітку рівномірною), щоб сформувати гарне закруглення ліктя чи коліна.

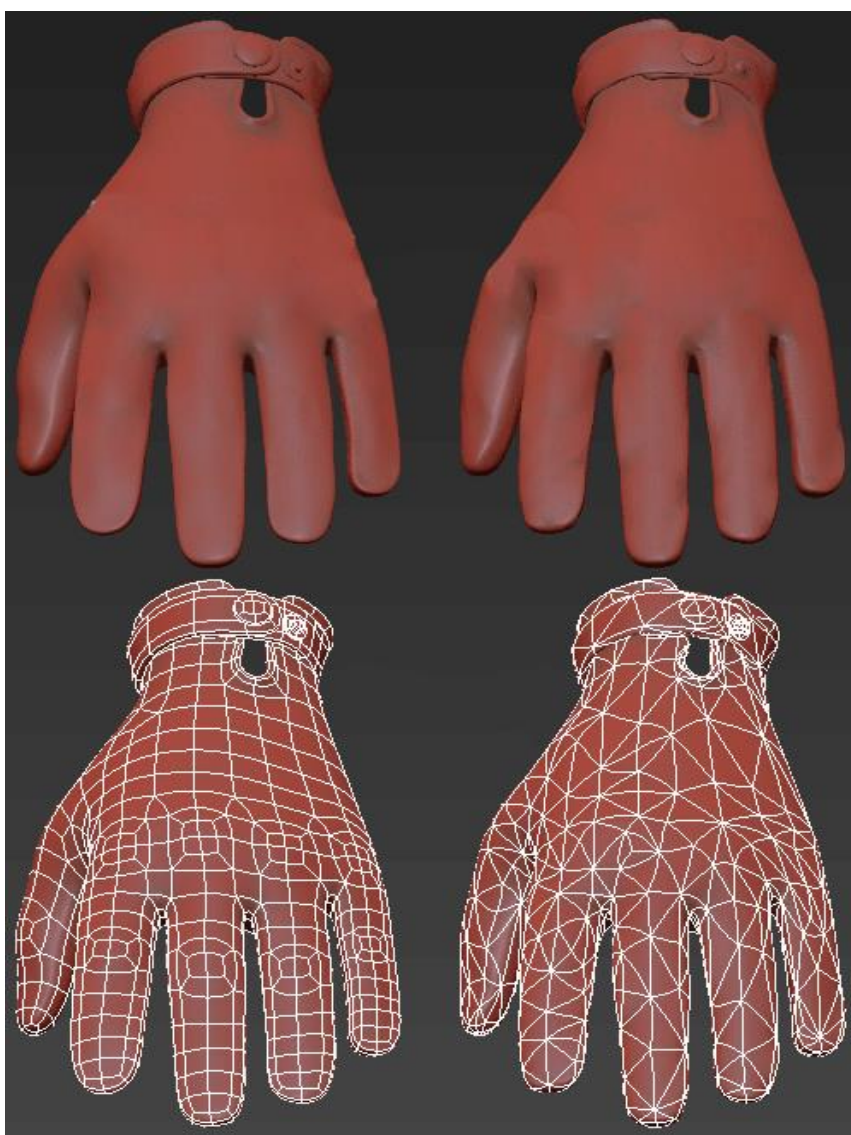


Рис. 1. Приклад правильної та неправильної топології

Другий випадок більш складний. Один з популярних методів створення 3D-моделей, має великий недолік – після завершення роботи та перенесення у іншу програму, модель має неймовірно велику кількість полігонів (деколи цифра може сягати десяти мільйонів). Жоден комп'ютер не зможе за адекватну кількість часу обробити рендер такої великої кількості даних. Для того, щоб уникнути таких неприємних ситуацій, було розроблено спеціальні інструменти, які допомагають спеціалістам «полегшити» 3D-модель, тобто знизити кількість полігонів з декількох мільйонів до двадцяти-вісімдесяти тисяч полігонів – це середнє обмеження у сучасних студіях. На допомогу приходить ретопологія.

Ретопологія – це створення повністю нової топологічної сітки на основі існуючої моделі, вона потрібна, щоб зменшити кількість полігонів на моделі та зберегти якісну сітку. Вручну виконувати цей процес довго та складно, але у 3ds Max 2022 автоматизували це й додали модифікатор Retopology. Розглянемо його більш детально.

Меню модифікатору розбито на два розділи:

Розділ Retopology містить у собі режими ретопології: ReForm, QuadriFlow та Instant Mesh. Рекомендується використовувати саме ReForm, оскільки він дає найкращий результат у більшості випадків. Також можна знайти лічильник Face Count, він показує кількість полігонів, яка буде у моделі після ретопології. За умовчанням воно дорівнює 5000, але часто його доводиться змінювати. Наприклад, для об'єкту типу дивана краще підходить 10000. Чим більше це значення, тим краще збережеться форма моделі. Далі розташована кнопка Compute, вона запускає розрахунок та перебудову сітки. Сам процес займає певний час, тому після натискання доведеться трохи зачекати. Reset – скидання створеної сітки та повернення до вихідної форми.

Розділ ReForm Settings. Цей розділ налаштувань має декілька покажчиків:

Quad Tolerance – значення, на яке модифікатор може відхилитися від значення у Face Count, але потрібно зауважити, що він ніколи не може точно повторити число Face Count.

Subdivision Factor – вказує, чи потрібно використовувати згладжування під час створення нових полігонів. Якщо стоїть 1, то функцію вимкнено. При значенні 2 генеруватиметься не тисяча полігонів, а всього двісті п'ятдесят. Але після генерації всі полігони додатково згладяться, як при TurboSmooth, в результаті чого можливо отримати назад тисячу. Треба бути обережним та не завишувати значення цього параметру, оскільки форма 3D-моделі може спотворитися (рис. 2).

Regularize – показує, наскільки однаковим буде розмір полігонів. Чим ближче до 1, тим більше можливо буде порівняти полігони за розміром і формою (рис.3). Але це не завжди добре. На гранях і кутах концентрація полігонів має бути вищою, ніж на плоских поверхнях, це краще передає форму.

Anisotropy – функція налаштування квадратної форми полігонів. Чим ближче до 0, тим більше полігони будуть схожі на квадрати. Чим ближче до 1, тим сильніше витягуватимуться. На практиці це стосується граней, оскільки на плоских поверхнях майже завжди будуть квадрати, тому помітити результат не завжди просто.

Adaptivity – показник сили зміни розміру полігону, залежить від кривизни поверхні.

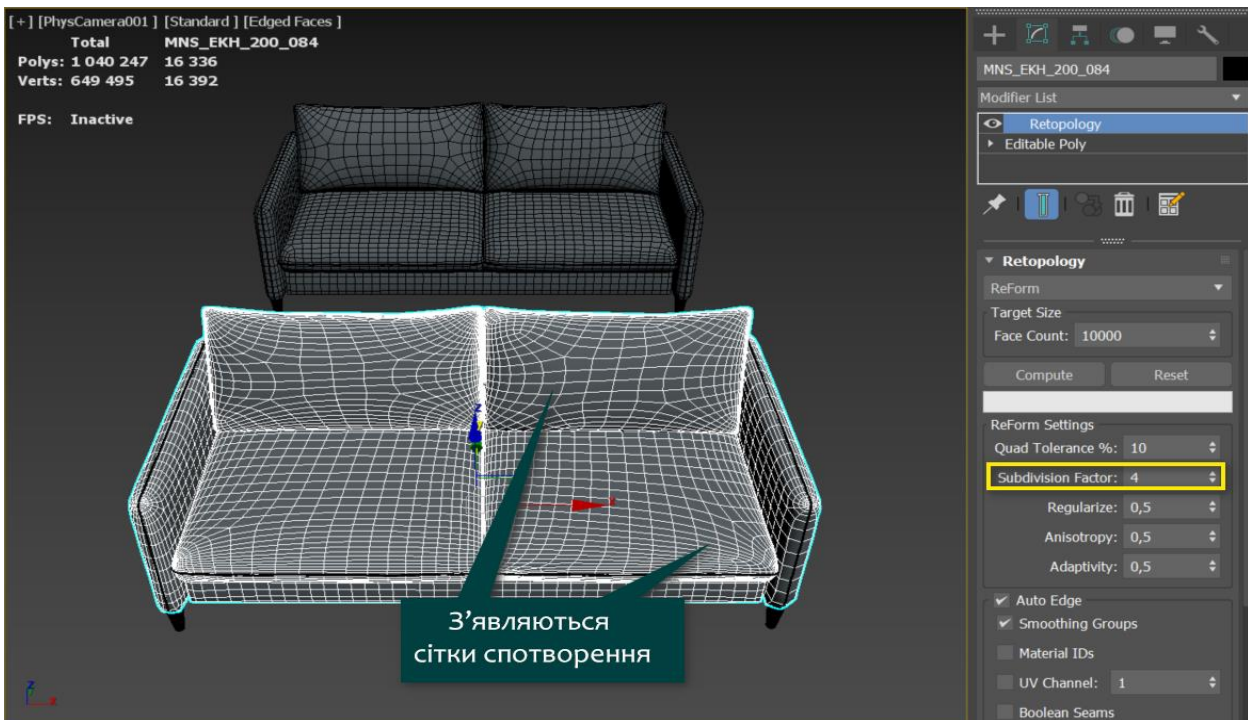


Рис.2. Приклад спотворення 3D-моделі при завищенні значення Subdivision Factor

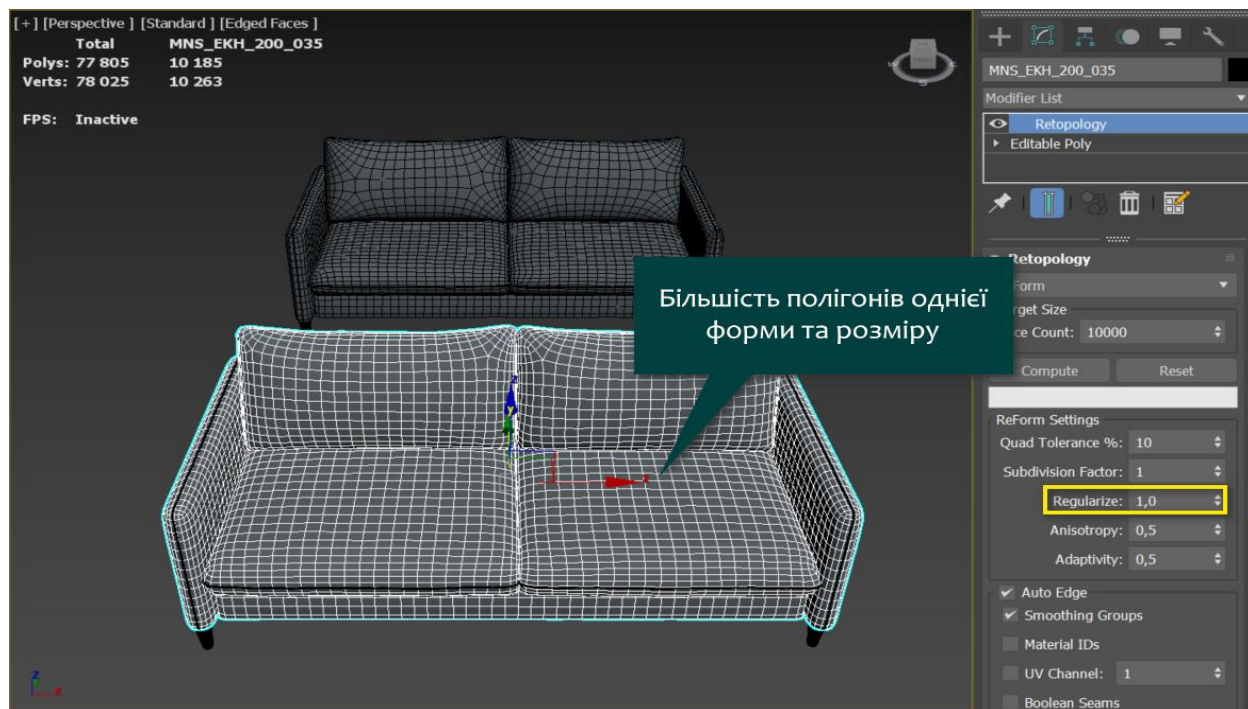


Рис. 3. Приклад впливу функції Regularize на форму та розмір полігонів

Але є суттєвий недолік – після ретопології може зникнути потрібна деталізація. Щоб цього уникнути використовують карти нормалей, це RGB-зображення, де кожен із каналів (червоний, зелений, синій) інтерпретується в X, Y та Z координати нормалей поверхні відповідно. Червоний канал простору дотичних карт нормалів відповідає за вісь X, зелений канал за вісь Y і синій канал за вісь Z. За допомогою цієї технології можна імітувати нерівності на поверхні та зробити 3D-модель більш схожою на її високополігонільну версію. З її допомогою можна доробити різні деталі, які не можна передати через геометрію за допомогою обмеження полігонажу на проєкті, і змусити модель виглядати більш заокругленою для кращої передачі освітленості та більшої реалістичності.

Тобто, після спрощення моделі до низькополігонального типу (lowpoly, двадцять-вісімдесят тисяч полігонів на весь об'єкт), її деталізовану копію, створену за допомогою скульптингу, «запікають». Так називається процес створення карт нормалей, на ній відбувається відображення яким чином світло падає на модель і який виходить при цьому відблиск. Після накладення цієї карти створиться світлова ілюзія деталізації при мінімальному полігонажі. Наглядно показано усі перетворення на рис. 4.

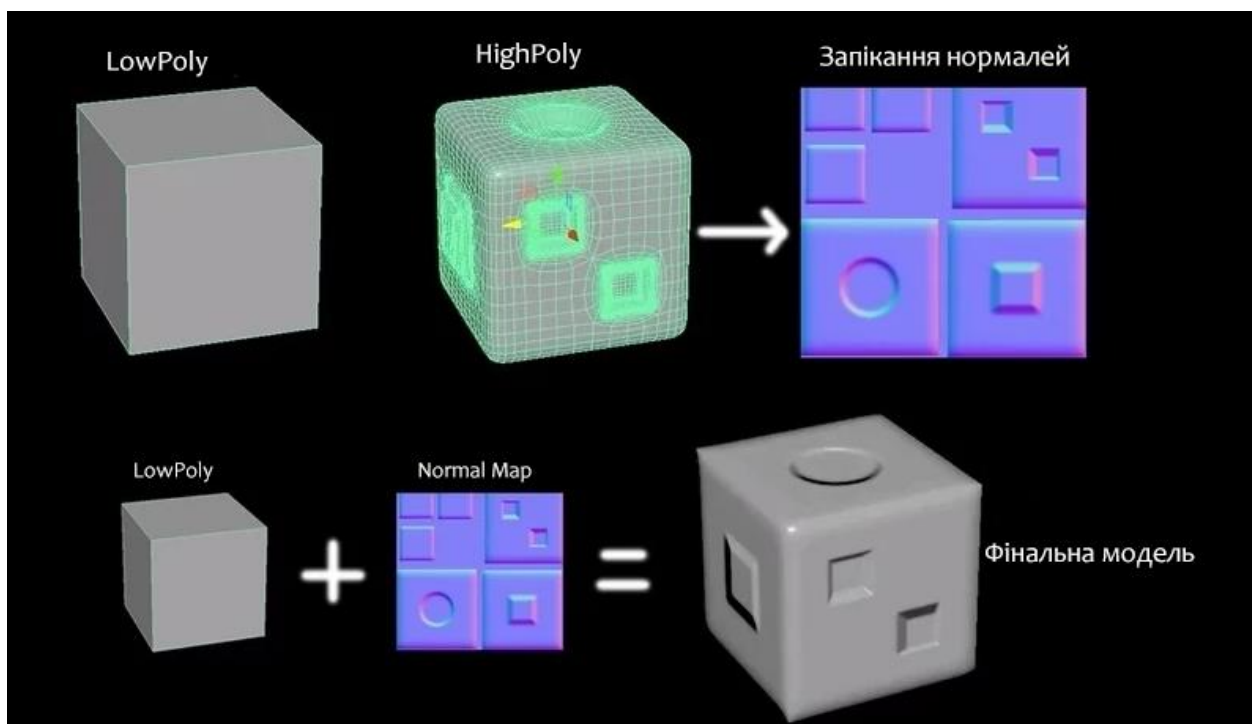


Рис. 4. Приклад використання карт нормалей

Висновок. Отже, ретопологію потрібно проводити якщо 3D-модель після створення потрібно буде анімувати або використовувати у подальшому, наприклад, у іграх. Для полегшення цього процесу було розроблено спеціальних інструмент – модифікатор Retopology, при правильних налаштування якого, топологія автоматично вирівнюється потрібним чином.

ПЕРЕЛІК ПОСИЛАНЬ

1. Retopology Tools for 3ds Max: електронний збірник документації. URL: <https://knowledge.autodesk.com/support/3ds-max/learn-explore/caas/CloudHelp/cloudhelp/2022/ENU/3DSMax-Retopology/files/GUID-57B77A00-5300-47CB-99E1-22B9C536B060-html.html> (дата звернення: 13.11.2022)
2. Retopology tools in 3ds Max: веб-сайт. URL: <https://www.stateofartacademy.com/retopology-tools-in-3ds-max/> (дата звернення: 13.11.2022)
3. 3dsMax 2021.3 Retopology Modifier Quick Start – Part.1: електронний гід. URL: <https://cganimator.com/3dsmax-2021-3-retopology-modifier-quick-start-part-1/> (дата звернення: 13.11.2022)

УДК 004.93

А.В. Малієнко¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

АНАЛІЗ СУЧАСНОЇ ПЛАТФОРМИ ІНФОРМАЦІЙНИХ ПОТОКІВ ТА МЕТОДІВ УПРАВЛІННЯ ПРОЕКТАМИ В УМОВАХ РОЗВИТКУ ЦИКЛІЧНОЇ ЕКОНОМІКИ

Анотація. Описано основні вимоги до інформації що формується в базу знань при переходу до циклічної економіки сучасного світу. Сформульовані основні принципи отримання, формування, обробки та прийняття даних для загального користування. Визначено постійне інформування зацікавлених осіб про склад продукту для вторинного використання. Зазначено, що актуальна інформація про місце збору, збереження, сортування відходів, їх склад повинна бути зібрана та передана зацікавленим підприємствам, які в режимі реального часу мають доступ до загальної, відкритої бази даних про товар. Зазначені основні вимоги до методів управління

Ключові слова: управління, проектний аналіз, циклічна економіка, інформаційні потоки, економіка замкнутого циклу.

Вступ. В економіці замкнутого циклу економічна діяльність створює та відновлює загальний стан системи. Концепція визнає важливість того, щоб економіка працювала ефективно у всіх масштабах — для великих та малих підприємств, для організацій та окремих осіб глобально та локально.

Перехід до економіки замкнутого циклу — це не лише коригування, спрямовані на зниження негативної дії лінійної економіки. Швидше, це системне зрушення, яке забезпечує довгострокову стійкість, створює ділові та економічні можливості та приносить користь навколишньому середовищу та суспільству.

Основний зміст роботи. Перехід до циклічної економіки потребує чіткої координації потоків матеріальних та інформаційних ресурсів. Інформацію про кількість, якість продукту та сировину, що міститься в них, необхідно збирати, опрацьовувати та зберігати. Щоб «відходи» ставали цінною сировиною, ця

інформація має бути основою кругообігу «відходів». Головним питанням є формування великого масиву інформації про матеріальний склад кожного окремого продукту, про характер його використання, його місце в ланцюгу роботи з «відходами» або виробничого циклу. Інформаційний потік з запитам та зворотній потік отриманих даних необхідно ефективно збирати, проводити його обробку та сформований потік актуальної інформації надати у розпорядження відповідним фахівцям для аналізу. Все це необхідно для нової системи розвитку «циклічної економіки» – створювання працюючих ринків виробничих циклів. Тоді в ринкових умовах, замість чистого лінійного регулювання, виникають ефективні рішення, засновані на сучасних законах циклічного ринку. Велика кількість даних в інформації потоках «циклічної економіки» ще навіть не сформовані, не наведена їх структура та не сформована база даних для актуального заповнення. Але це питання до цифрової трансформації «циклічної економіки».

В умовах циклічної економіки інформація про склад продукту, його використання та повторну переробку повинна супроводжувати продукт протягом усього терміну його експлуатації, а також включати всі екологічно/економічні важливі дані продукту [1].

Використання сучасної Технології Індустрії 4.0 дозволить отримувати та надалі зберігати дані в режимі реального часу. Актуальна інформація про місце збору, збереження, сортування відходів, їх склад повинна бути зібрана та передана зацікавленим підприємствам, які в режимі реального часу вже зможуть провести планування свої виробничих планів та технологічних процесів. Нові методи аналізу даних - на базі штучного інтелекту (Big Data) – дадуть змогу прийняти, ухвалити рішення про повторне використання сировини, та провести оптимальні розрахування логістичних маршрутів доставки.

Злагоджену роботу для формування пропозицій та попиту на вторинну сировину можна прискорити за допомогою формування єдиної-відкритої бази в системі промислового товарообігу. Робота автоматизованої ринково-орієнтованої, логістичної загальної платформи значно скоротить час на пошук та доставку сировини, що обумовить значну економію фінансової складової підприємства. Використовуючи відкриту загальну систему, буде сформована тенденція широкого продукту, який створить попит для себе на ринку. Завдяки загальному доступу через інтернет до платформи промислового товарообігу, товар продає себе сам на основі інформації про свій склад та можливості застосування його в технологічному процесі. На сьогодні сировина вторинного використання дешевша, ніж первинна сировина, та цю перевагу можна буде ще використовувати при описі та можливості запропонувати нові рішення, нові методи використання [2]. Можливість доступу до вторинної сировини та технології переробки стануть конкурентною перевагою та сформує конкурентний ринок переробки, створюючи нові робочі місця з новими вимоги та сучасними систем та методами управління проектами циклічної економіки [3].

Зазначимо, що в ставленні формування систем управління, інноваційних змін в роботі з відповідними «базами даних циклічної економіки», як новій світовій економіці є велика кількість перешкод.

На сьогодні, для досягнення конкурентних переваг недостатньо успішно виконати окреме завдання або окремих проект, оскільки досягнення конкурентної переваги є не лише наслідком успішної реалізації одного проекту, це є досягнення мети всього загального проекту інновацій. Основна проблема для системи управління є формування зв'язків вектору розвитку визначеної системи з критеріями та пріоритетами при прийнятті відповідальних рішень виконання поставлених задач при управлінні проектом. Сучасні методи і засоби управління проектами дозволяють здійснювати контроль проектів, забезпечують їх виконання на різних рівнях управління – на стратегічному і оперативному, тому є потреба у вирішенні першочергових задач управління на підприємствах для переходу до циклічної економіки :

- здійснення аналізу управління розвитком підприємства за умовами переходу до циклічної економіки;
- розробки сучасної моделі управління розвитком окремого підприємства в умовах циклічної економіки та обґрунтувати методологію досліджень та впровадження;
- удосконалення професійного управління розвитком підприємства в умовах глобальної економіки шляхом адаптації понять і категорій циклічної економіки із категоріями системної синергетики;
- розробки методів для підготовки актуальних рішень для додаткових проектів розвитку підприємства на основі співставлення внутрішніх організаційних процесів та процесів зовнішнього середовища та їх вплив;
- розробки методологічної основи з синергетичного підходу управління розвитком підприємств в умовах циклічної економіки;
- провести узагальнення методологічних основ діяльності з циклічно-орієнтованого напрямку управління розвитком підприємств.

Висновки. Таким чином – аналіз ринкового стану та формування вимог показує перехід до циклічної економіки є сучасним розвитком та економічного стану країни, але при цьому потрібно провести велику кількість роботи не тільки на економічному але й науковому поприщі. Новітні розробки систем інформаційних технологій формують сучасний вигляд та доступ до необхідної та актуальної інформації в онлайн режимі для всіх зацікавлених осіб. Підприємства потребують зміни в управлінні та розвитку для переходу до циклічної економіки. На сьогодні вже класичний підхід до охорони навколишнього середовища, управлінні відходами та науковим підходом до лінійної економіки не є ефективними. Потрібні сучасні методи, комплексний підхід, який сформує та надасть переваги циклічної економіки. Основою такого підходу може стати методологія управління проектами, оскільки проектна методологія є потужним інструментом здійснення змін в умовах обмежень часу та ресурсів

ПЕРЕЛІК ПОСИЛАНЬ

1. MU o. J.: Umweltpolitische Digitalagenda[BMU б.г.: <https://www.bmu.de/digitalagenda/>.
2. European Commission; Directorate-General for Communication: Circular economy action plan: For a cleaner and more competitive Europe, [https://op.europa.eu/en/publication-detail/-/publication/45c-c30f6-cd57-11ea-adf7-01aa75ed71a1/languageen/format-PDF/source-170854112.European Environment Agency 2018: Waste prevention in Europe. Policies, status and trends in reuse in 2017. 4/2018 \[EEA Report\], <https://doi.org/10.2800/15583>.](https://op.europa.eu/en/publication-detail/-/publication/45c-c30f6-cd57-11ea-adf7-01aa75ed71a1/languageen/format-PDF/source-170854112.European%20Environment%20Agency%202018%3A%20Waste%20prevention%20in%20Europe.%20Policies,%20status%20and%20trends%20in%20reuse%20in%202017.%204/2018%20%E2%80%9C%E2%80%9A%20%E2%80%9C%E2%80%9A%20Report%5D,%20https://doi.org/10.2800/15583)
3. А.В. Малієнко. Аналіз та формування методологічних основ управління циклічною економікою./ Управління проектами. Перспективи розвитку проектного та нейроменеджменту, інформаційних технологій управління, технологій створення та використання об'єктів права інтелектуальної власності: збірник наукових праць за матеріалами IV Міжнародної науково-практичної інтернет конференції (24-25 березня 2022 р.). – УДУНТ, УКРНЕТ, НДІПВ НАПрН України, Дніпро: Юрсервіс, 2022. С 66- 71.

УДК 621-926

М.В. Рогоза¹, В.А. Бородай¹, О.Ю. Нестерова¹, Є.В. Кошеленко¹, С.І. Федоров¹
¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ШВИДКІСНЕ ОБСЛУГОВУВАННЯ ТЯГОВИХ АКУМУЛЯТОРІВ СУЧАСНИХ АВТОТАРНСПОРТНИХ ЗАСОБІВ

Анотація. Запропоновано оригінальний спосіб швидкісного обслуговування тягового акумулятора сучасного автотранспортного засобу способом стаціонарного-мобільного заряду та надано рекомендації щодо можливої реалізації способу, який забезпечить надійну та тривалу роботу електромеханічної системи цілком.

Ключові слова: *електромобілі, стаціонарний швидкісний заряд, заряд авто у русі, надійність та якісне обслуговування акумуляторів.*

Вступ. Сучасна проблема зменшення карбонового сліду, який виникає внаслідок роботи автотранспорту на двигунах внутрішнього горіння, спонукала провідні держави світу до впровадження у широкий загаль автомобілів на електричній тязі. Крім того такий перехід пов'язаний, зокрема з обмеженою кількістю вуглеводневого палива, якого, за приблизними розрахунками, залишилося на планеті не більше ніж на 100 років. За очевидних переваг електричного привода (компактність двигуна і системи керування, кращі динамічні властивості, підвищена екологічність) слід відмітити, що йому притаманні й деякі недоліки [1]. Перш за все це тривалість заряду, який у номінальному режимі може сягати п'яти годин, а зважаючи на те, що ємність заряду акумуляторів досить сильно залежить від температури навколишнього середовища, термін та його циклічність має велике значення.

Мета роботи – розробка оригінального способу швидкісного обслуговування тягового акумулятора сучасного автотранспортного засобу

способом стаціонарного-мобільного заряду задля забезпечення його високої надійності та тривалості служби.

Основний зміст роботи. Досвід використання сучасної електромобільної техніки показав, що вона є найбільш придатною для експлуатації в міських умовах. Поясненням причин цього є недостатня ємність заряду, яка лімітує дальність поїздки, а досить тривале його поновлення ставить часові обмеження при використанні такого транспортного засобу. У випадку поїздок, що можуть перевищувати 400 кілометрів, системи живлення такого типу є мало придатними.

Аналіз відомих систем поновлення заряду свідчить, що для швидкого заряду потрібно забезпечити струм, який є десятою частиною від паспортної ємності акумулятора. У цьому випадку його тривалість може коливатись у межах від п'яти до дев'яти годин. За швидкісного заряду використовують підвищене значення струму [2-3], який одночасно супроводжується досить інтенсивним зносом пластин акумулятора. Відомі імпульсні методи заряду, хоча і знижують час процесу, але повністю уникнути від зносу пластин не дозволяють [4]. Висновок - існуючі методи швидкісного заряду акумуляторів не дають змоги поліпшити ситуацію.

На відміну від попередніх підходів авторами пропонується комбінована система стаціонарного-мобільного заряду. Сутність стаціонарного способу полягає в побудові мережі зарядних станцій з кваліфікованим персоналом, який забезпечить професійне обслуговування гальванічних батарей, а споживач буде користуватися орендною акумуляторною батареєю, яку на такій станції можливо демонтувати /монтувати протягом декількох хвилин. Тобто забезпечити умови подібні заправці авто з двигуном внутрішнього горіння. Таким чином, може бути вирішена потрібна задача:

- перша – зняття проблеми обмеження по часу використання транспортного засобу;
- друга – встановлення станцій обслуговування акумуляторів і систем безконтактного живлення із енергосистемою на відновлюваних джерелах забезпечить якісне і професійне їх обслуговування;
- третє – беручи до уваги те, що поширення таких станцій має досить великі масштаби, дана розробка дасть суттєве зростання робочих міць, що відповідно поліпшить соціальну ситуацію у масштабах держави.

За вирішення проблеми у мобільному вигляді пропонується задіяти на магістральних трасах трансформатори безконтактної передачі енергії [5] із підвищеною частотою роботи. Особливістю такого вирішення проблеми є те, що такі системи заряду не потрібно встановлювати протягом усієї протяжності шляху, а лише у тих місцях, де за розрахунками попередній заряд батарей повинен закінчуватись. Таким чином, зарядні системи запропонованого типу дозволять здійснювати періодичний підзаряд у заздалегідь визначених місцях і в потрібний розрахунковий термін, а циклічність їх розташування дозволить зберегти ресурси при їх побудові.

Наукова новизна роботи полягає у розробці оригінального комбінованого способу стаціонарного та мобільного підзаряду акумуляторних батарей, який дозволить покращити користувацькі якості електромобілів, підвищить їх надійність та довговічність, а також дозволить забезпечити населення робочими місцями у масштабах держави.

Висновки. Запропонований оригінальний спосіб заряду акумуляторних батарей електромобілів дозволить уникнути тривалого процесу заряду і часового обмеження використання транспортних засобів, що у кінцевому рахунку підвищить їх користувацькі якості.

ПЕРЕЛІК ПОСИЛАНЬ

1. Електромобіль: очікування та реальність. URL: <https://consumerhm.gov.ua/2854-elektromobil-ochikuvannya-ta-realnist>
2. Що найбільше впливає на знос батареї електромобіля: названо основні причини. URL: <https://autogeek.com.ua/shcho-najbilshe-vplyvaie-na-znos-batarei-elektromobilia-nazvano-osnovni-prychyny/>
3. Типи зарядних пристроїв та їх особливості повідомлення. Класифікація типів зарядних пристроїв. Як довго триває заряджання різних пристроїв. URL: <https://oborudow.ru/uk/salon/tipy-zaryadnyh-ustroistv-i-ih-osobennosti-soobshchenie-klassifikaciya/>
4. Швидке зарядження: що працює вже сьогодні і що чекає нас в майбутньому. URL: <https://www.imena.ua/blog/fast-charging-part-1/>
5. Пивняк Г.Г., Ремизов И.П., Саратикянц С.А. и др. Транспорт с индуктивной передачей энергии для угольных шахт / Под общей ред. Г.Г.Пивняка. М.: Недра, 1990. 245 с.

УДК 004.415.3:681.6

Д.Д. Чумичов¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ОГЛЯД ТЕХНОЛОГІЇ ІНТЕРНЕТУ РЕЧЕЙ ТА ЙОГО КЛЮЧОВИХ КОМПОНЕНТІВ

Анотація. Дана коротка характеристика інтернету речей, описано його значення і зростаюча необхідність в сучасному світі. Розкрито поняття інтернету речей та зазначені його ключові компоненти: датчики, приводи та контролери. Наведено приклади застосування цієї технології в різних сферах діяльності людини і варіанти її подальшого розвитку.

Ключові слова: інтернет речей, датчик, привод, контролер, туманні обчислення.

Вступ. Інтернет речей (англ. internet of things, IoT) – це концепція мережі передачі між фізичними об'єктами («речами»), оснащеними вбудованими засобами і технологіями взаємодії один з одним чи із довкіллям.

Інтернет речей (IoT) є всюди навколо нас і швидко розповсюджується. Ми тільки починаємо отримувати переваги IoT. Він допомагає людям об'єднувати (з'язувати) речі для покращення якості життя. Постійно розробляються нові способи використання пов'язаних речей. Мільйони людей

зараз використовують портативні пристрої для відстеження своєї фітнес-активності. Прикладом такого пристрою є сорочка, призначена для відстеження вашого рівня фізичної підготовки. IoT також допомагає організаціям і галузям покращити управління ресурсами, щоб стати ефективнішим, і використовується в розумних містах для підключення речей для більш комфортного життя громадян.

Основна частина. Існує багато різних пристроїв із підтримкою IoT. Більшість цих пристроїв використовують датчики, контролери та виконавчі механізми для виконання своїх функцій. Петлі зворотного зв'язку використовуються пристроями IoT для надання інформації в реальному часі його контролеру на основі поточної поведінки. Замкнутий цикл існує, коли контролер постійно отримує зворотний зв'язок від своїх датчиків. Контролер аналізує та обробляє інформацію, і при необхідності може використовувати виконавчі механізми для зміни умов. Цей процес постійно повторюється і коригується.

Наприклад, на рис. 1 показано компоненти IoT у розумному термостаті. Датчик (або кілька датчиків) безперервно надсилає показання температури до контролера. Контролер використовує цей зворотний зв'язок для моніторингу умов і регулювання температури відповідно до вимог користувача. Контролер також може використовувати бездротовий зв'язок з користувачем.

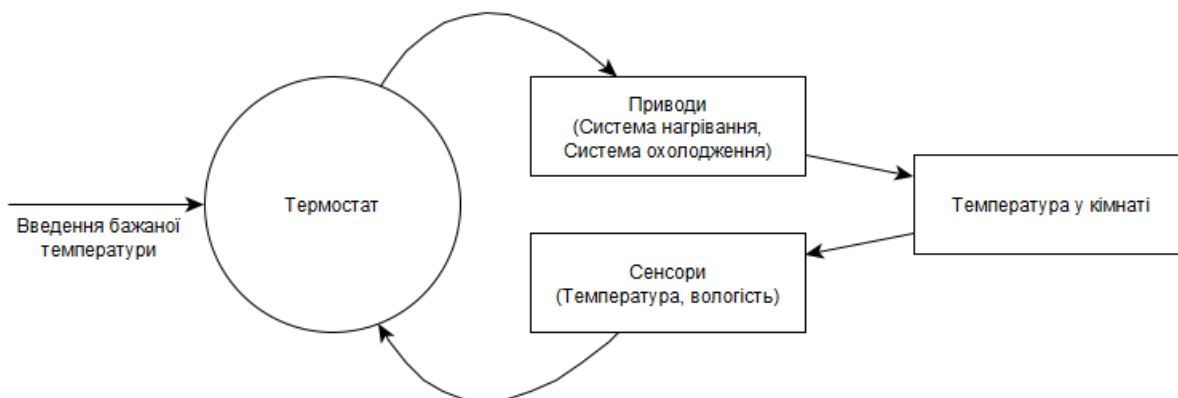


Рис. 1. Компоненти IoT у розумному термостаті

Ключові компоненти систем Інтернету речей включають датчики, які підключаються через бездротове або дротове з'єднання до приводів або контролерів. Розглянемо ці компоненти докладніше:

Датчик.

Датчик – це пристрій, який можна використовувати для вимірювання фізичної властивості шляхом виявлення певного типу інформації з фізичного світу. Ця інформація може бути світлом, вологістю, рухом, тиском, температурою або будь-якими іншими умовами навколишнього середовища. Наприклад, кавова ферма може використовувати датчики для збору різноманітної інформації, як-от сонячне світло, температура та вологість

грунту. Потім ці дані можна проаналізувати, щоб допомогти максимізувати врожайність і якість кавових зерен.

Датчик може бути підключений до контролера безпосередньо або віддалено. Датчики та контролери зазвичай з'єднуються за допомогою аналогових або цифрових схем. Датчики передають дані на контролер. Контролер негайно реагує на дані датчиків та змінює налаштування приводу. Як приклад, мережа контролера (CAN) – це стандарт, який використовується в більшості сучасних автомобілів для забезпечення зв'язку між датчиками та контролерами без будь-яких керуючих комп'ютерів. Більшість нових автомобілів оснащено відеокамерами та датчиками запобігання зіткненням. Ці камери та їхні датчики можуть попередити контролер, якщо на шляху є об'єкт, і контролер автоматично змусить автомобіль видавати попереджувальний сигнал або застосовувати гальма.

Цей контролер також може діяти як шлюз до IP-мережі та передавати дані датчиків для зберігання та аналізу на серверах у тумані чи хмарі. Зібрані дані та аналізи можна використовувати для ініціювання дій людей, систем або машин. Наприклад, кількість автомобілів на дорозі може спричинити зміну системи світлофора.

Привід.

Привід – це пристрій, який можна використовувати для переміщення або керування механізмом чи системою на основі певного набору інструкцій. Як правило, в промисловому Інтернеті речей існує три типи приводів:

- електричний – працює від двигуна, який перетворює електричну енергію в механічну;
- гідравлічний – використовує тиск рідини для виконання механічного руху;
- пневматичний – використовує стиснене повітря для виконання механічних операцій.

В інших областях виконавчий механізм може відповідати за перетворення електричного сигналу у фізичний вихід. Цей фізичний вихід може надавати інформацію користувачеві через світлодіоди або змінювати інший пристрій чи середовище. Наприклад, нагрівач у контурі зворотного зв'язку термостата є приводом, оскільки він змінює стан контрольованого середовища (температуру) у відповідь на електричний сигнал.

Контролер.

Контролери відповідають за збір даних із датчиків і забезпечення підключення до мережі чи Інтернету. Контролери можуть мати можливість негайно приймати рішення або вони можуть надсилати дані на більш потужний комп'ютер для аналізу. Цей потужніший комп'ютер може бути в тій самій локальній мережі, що й контролер, або може бути доступним лише через підключення до Інтернету.

Топологія IoT складається з датчиків, контролерів, маршрутизаторів і центрів обробки даних. Щоб отримати доступ до потужніших комп'ютерів у

центрі обробки даних, контролер спочатку надсилає дані на локальний маршрутизатор. Цей маршрутизатор є інтерфейсом між локальною мережею та Інтернетом. Він може пересилати дані між ними.

Контролери також використовуються в туманих обчисленнях. Наприклад розумні світлофори, які містять датчики та виконавчі механізми. Датчики виявляють рух транспорту та повідомляють контролеру. Контролер здатний обробляти ці дані локально та визначати оптимальні моделі трафіку. Використовуючи цю інформацію, контролер надсилатиме сигнали виконавчим механізмам світлофора для регулювання потоків транспорту. Це приклад міжмашинного зв'язку (M2M). У цьому сценарії датчики, виконавчі механізми та контролер знаходяться у тумані. Це означає, що інформація не надсилається за межі локальної мережі кінцевих пристроїв.

Контролери можуть мати IP-підключення. Контролер із підтримкою IP передає інформацію через мережу IP і дозволяє окремим особам отримувати віддалений доступ до контролера. Окрім пересилання основної інформації в конфігурації M2M, деякі контролери можуть виконувати більш складні операції. Деякі контролери можуть консолідувати інформацію з кількох датчиків або виконувати базовий аналіз отриманих даних.

Мікроконтролер Arduino та Raspberry Pi (RaPi) є прикладами контролерів. Вони можуть також працювати без Інтернету і використовуються любителями та професіоналами. Ключовою відмінністю між ними є фізичний розмір, доступна обчислювальна потужність, пам'ять і ОС. Зазвичай Arduino потребує менше енергії, ніж RaPi, і більше підходить для аналогового входу. Додаток повинен диктувати, який контролер найкраще використовувати. Обидва контролери зазвичай використовуються разом. Наприклад, ви можете отримувати дані за допомогою Arduino, а потім обробляти їх за допомогою Raspberry Pi.

Незалежно від того, як виконавчий механізм викликає рух, його основною функцією є отримання сигналу від контролера та виконання заданої дії на основі цього сигналу.

Як і багато компонентів у будь-якій системі, деякі пристрої можуть мати більше ніж одну функцію. Це стосується контролера в системі IoT. Контролер може збирати дані з датчиків без втручання людини чи підключення до мережі. Наприклад, датчик визначає, що температура в квартирі впала нижче заданого рівня. Контролер обробить дані та надішле вихідні дані, щоб увімкнути нагрівач (привід).

Контролер також може діяти як шлюз до локальної мережі. У попередньому прикладі, якщо система IoT розроблена для реєстрації змін температури в кожній квартирі будинку, контролер може передавати дані для зберігання або аналізу на серверах у локальній або периферійній мережі. Місце обробки даних впливає на швидкість, з якою можуть відбуватися зміни в системі. Дані можна зберігати й обробляти на пристроях, розташованих поблизу межі мережі або навіть ближче до датчиків. Цей тип обробки називається туманим обчисленням. Вузли туману, які створюють зони для обробки, є частиною цієї системи.

Контролер також може бути шлюзом до хмарної мережі. Якщо система IoT з нашого початкового прикладу розроблена для агрегування даних термостатів з кількох різних багатоквартирних будинків, що належать одній компанії, кілька контролерів і пристроїв можуть зберігати й обробляти дані датчиків у різних вузлах туману. Дані з різних вузлів туману будуть зберігатися, агрегуватися та аналізуватися. Потім ці проаналізовані дані можна використовувати для прийняття обґрунтованих рішень.

Висновки. Були розглянуті поняття інтернету речей та його основні компоненти: датчик, привід та контролер. Наведені приклади можливого виростання IoT у повсякденному житті людини. В найближчі десятиліття IoT стане таким же звичним явищем, як електрика чи Інтернет. Це підтверджується зростаючою кількістю пристроїв, підключених до мережі Інтернет, які роблять життя людей комфортнішим.

ПЕРЕЛІК ПОСИЛАНЬ

1. Cisco Networking Academy: [Cisco Networking Academy Builds IT Skills & Education For Future Careers \(netacad.com\)](https://www.netacad.com/learning-paths/networking/for-future-careers)
2. <https://www.i-scoop.eu/internet-of-things-guide/>
3. <https://www.ibm.com/blogs/internet-of-things/what-is-the-iot/>
4. <https://www.zdnet.com/article/what-is-the-internet-of-things-everything-you-need-to-know-about-the-iot-right-now/>

УДК 004.75+004.67:681.5

С.М. Ткаченко¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ОБґРУНТУВАННЯ ТЕХНІЧНОГО РІШЕННЯ ДЛЯ СИСТЕМ ЦИФРОВОЇ ТЕРМОМЕТРІЇ ЗЕРНА НА ОСНОВІ КОНТРОЛЕРА ARDUINO

Анотація. Обґрунтовано вимоги й рекомендації до обладнання системи термометрії зерна, яка використовує підвіси на цифрових датчиках. Обрано серійну модель контролера Arduino, розроблено на його основі блок збирання інформації, надано рекомендації щодо функціоналу програмного забезпечення блоку. Вказано на можливості щодо реалізації варіантів топологій системи.

Ключові слова: система термометрії, ds12b20, цифровий датчик, термонідвіс, блок збирання інформації, інтерфейс I-wire, Arduino, збірковий контролер, АРМ термометрії, бітгандінг, паразитне живлення I-wire, зовнішнє живлення I-wire.

Вступ. На сьогоднішній день на елеваторах України широко застосовуються системи термометрії в силосах, які побудовані на основі цифрових датчиків фірми Dallas Semiconductor. Ці датчики, на відміну від термісторів, що також використовуються, не потребують періодичного

калібрування. Хоча внаслідок недосконалості конструкції термopідвісів та проблем зі статичною електрикою цифрові датчики частіше виходять з ладу[2], перевагу надають все ж їм через нижчу вартість придбання та обслуговування. На ринку існує досить багато фірм, які пропонують власні технічні рішення для отримання даних з термopідвісів на цифрових датчиках. Ознака, яка об'єднує подібні розробки – використання обладнання з порівняно низьким ступенем стандартизації за умови порівняно невеликих обсягів поставок на об'єкти. Для прикладу, збірні контролери можуть навіть мати власний закритий протокол для зв'язку з АРМ термометрії. У результаті, якщо фірма-поставник системи термометрії іде з ринку, часто виникає необхідність повної заміни обладнання, яке збирає інформацію від термopідвісів. Застосовувати існуючі засоби стандартних промислових контролерів замість унікальних розробок економічно недоцільно, оскільки здебільшого їх номенклатура не передбачає організацію територіально розподіленої мережі з сотень, а у деяких випадках тисяч датчиків температури. Більше того, мережа 1-wire, яку використовують датчики температури Dallas Semiconductor, не реалізована як промислова засобами згаданих контролерів. Таким чином, виникає актуальна задача розробки системи термометрії для використання сумісно з цифровими термopідвісами, такої, що використовує серійні апаратні та відомі програмні засоби. З іншого боку, на ринку України на сьогодні існують готові стандартні рішення – контролери Arduino. Вони мають порівняно невисоку вартість і при цьому можуть використовуватись із датчиками ds18b20[1]. На додачу є готові розробки програмних бібліотек підтримки цих датчиків.

Мета представленої роботи полягає в розробці технічного рішення для організації системи термометрії зерна на базі термopідвісів з цифрових датчиків ds18b20 на базі контролерів Arduino та з використанням стандартних промислових мереж. Це забезпечить можливість обслуговування впроваджених систем альтернативними організаціями, можливість модернізацій шляхом перенесення апробованих програмних рішень на інші технічні засоби, масштабування систем термометрії, можливо, з використанням обладнання, відмінного від вже встановленого.

Постановка задачі. Для досягнення поставленої мети в роботі сформовані і вирішені такі завдання:

- проаналізувати схеми розташування термopідвісів та можливу кількість датчиків на один термopідвіс для сховищ зерна різного типу;
- визначити кількість точок підключення термopідвісів до блоку збирання інформації функціональну схему системи термометрії;
- розрахувати обмеження на вибір та обґрунтування обладнання Arduino з точки зору струму споживання та потрібних обсягів пам'яті;
- розробити рекомендації для використання обладнання Arduino з точки зору методів його програмування для системи термометрії;

- запропонувати обґрунтоване технічне рішення для системи термометрії зерна на основі цифрових термопідвісів, сумісне із типовими промисловими мережами та обладнанням;

- зробити висновки щодо переваг запропонованого рішення.

Основний зміст роботи. Під час роботи над даною роботою були використані наступні методи та інструменти.

- аналіз існуючих технологічних схем вимірювання температури зерна і технічних засобів;

- обґрунтування критеріїв до вибору типового обчислювального обладнання, яке повинне використовуватись у складі систем термометрії;

- застосування визначених критеріїв до вибору обладнання системи термометрії та розробка рекомендацій з програмних методів керування вимірюванням температури;

- синтез технічного рішення системи термометрії зерна для цифрових термопідвісів на основі контролерів Arduino.

Для обґрунтування критеріїв, необхідних для вибору обладнання системи термометрії необхідно розглянути будову термопідвісів та схеми їх розташування на зерносховищах різного типу. Кількість датчиків у термопідвісі може сягати 35 [2] у залежності від висоти і типу ємності зберігання зерна із розрахунку 1 датчик на 1,3...2 метри довжини підвісу [3]. Насправді, кількість датчиків на підвіс визначається особливостями об'єкту. Так, сховища складського типу з плоским дном використовують мають висоту насипу до 5,5 м, але у термопідвісі використовується до 7 датчиків, а з конічним – до 11,5 м з 11 датчиками на один термопідвіс. При цьому в обох випадках склад може мати до 24 ряди комор по 6 у кожному ряді [3]. Отже, потрібно до 144 термопідвісів. Якщо ж узяти металеві силоси, наприклад, GSI на елеваторі ТОВ «Елеватор-Агро» у селі Червонозаводське Полтавської області то тут є 6 силосів на 24 термопідвісів по 17 датчиків, та 6 силосів на 3 термопідвіси по 10 датчиків [4]. Тобто 162 точки. Наостанок, розглянемо силосний корпус зберігання шроту заводу по переробці масляних культур, де була встановлена система термометрії на термопідвісах з датчиками ds18b20 [5]. Тут застосовано 78 термопідвісів довжиною 15 м по 6 датчиків у кожному за аналогією до термісторних термопідвісів, які використовувались для залізобетонних силосних корпусів раніше. Переважна більшість елеваторів України залізобетонні, тому представлений склад шроту є більш-менш типовою конструкцією, хоча на зернових елеваторах таких корпусів може бути й до трьох-чотирьох під загальною галерею. Тобто для таких елеваторів потрібно близько 300 термопідвісів.

Для подальшого дослідження цікавим є факт, що тут використане обладнання однієї з фірм-виробників систем термометрії для цифрових термопідвісок на основі контролерів власного виробництва [2]. Виходячи з представленої функціональної схеми, до блоків збору інформації можна комутувати до 12 термопідвісів [5], що зумовлено не лише поточним проектом,

але й типовими для таких споруд рішеннями з монтажу шаф систем термометрії й прокладкою до них вимірювальних кабелів. Тобто для залізобетонного силосного корпусу як правило, встановлюється одна шафа комутаційна або збору інформації на 12 силосів. Варто зазначити, що для збору інформації навіть у металевих силосах з 18-24 термopідвісами більшість виробників систем термометрії використовують ті ж самі блоки збирання інформації на 12 термopідвісів, імовірно, через техніко-економічну доцільність саме такого варіанта. Тому візьмемо це за один з критеріїв для обґрунтування вибору готового контролера для термометрії й способів його використання. Також приймемо функціональну схему на рис. 1 за основу для вибору й обґрунтування обладнання для нового технічного рішення для системи термометрії.

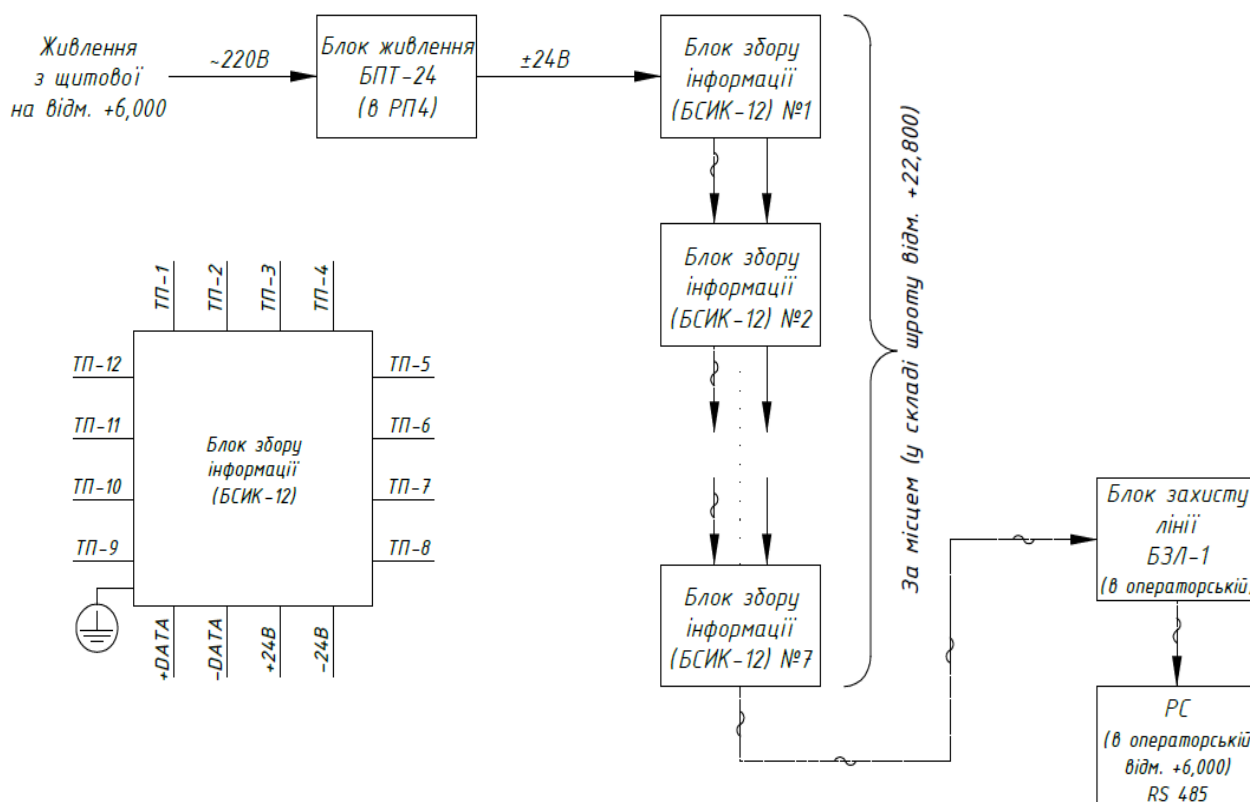


Рис. 1. Функціональна схема системи термометрії сховища шроту

Далі розглянемо сам цифровий термopідвіс. Інтерфейс 1-wire, використаний для зв'язку з датчиками температури має довжину, визначену довжиною термopідвісу та комутаційного проводу від монтажної колодки у голівці термopідвісу до колодки контролера блоку збирання інформації. Передати дані по шині 1-wire від кожного датчика безпосередньо до АРМ термометрії технічно неможливо через велику кількість самих датчиків (сотні чи до тисячі), так і відстань (до сотень метрів). Тому другим критерієм вибору контролера – використання промислового інтерфейсу.

Датчики ds18b20 для живлення використовують схему паразитного або схему зовнішнього живлення [6]. Реальні термopідвіси використовують обидва

рішення, це зумовлено не лише обмеженням у 30 датчиків на шині паразитного живлення, але й загальною довжиною термopідвісу та комутаційного кабелю, що у металевих силосах можуть сягати ще десятків метрів [4]. На додачу, на деяких підприємствах посилений рівень електромагнітних перешкод, що робить інтерфейс 1-wire з паразитним живленням датчиків неефективним. Тому надалі вважаємо основною схемою живлення термopідвісу із зовнішнім живленням. Паразитна схема є опціональною, і перехід на неї за умови використання того ж самого контролера не складає проблеми.

Визначимося з режимами роботи датчиків термopідвісу. Під час вимірювання температури датчик ds18b20 споживає максимум струму до 4 мА, а у режимі зупинки до 1 мА напругою від 3 до 5 Вольт [6]. Розрахуємо сумарний струм, що споживається 12 термopідвісами на 35 датчиків (беремо пiдвіс максимальної довжини для металевих силосів). Для активної роботи він складе 1680 мА, для роботи у режимі зупинки 420 мА. Обидва ці значення значно перевищують струм, що може видати навіть порівняно високовартісний Arduino Due [7]. Але цю проблему можна вирішити схемотехічно, якщо просто відключати живлення непотрібних для поточного вимірювання термopідвісів. Для цього необхідно передбачити окреме живлення кожного термopідвісу через дискретні виходи. Дійсно, термopідвіска на 35 датчиків споживає $35 \cdot 4 = 140$ мА при їх одночасному активному вимірюванні, або $34 \cdot 1 + 1 \cdot 4 = 38$ мА якщо активний лише один датчик. Навіть найменш потужний Arduino Nano здатен видати до 40 мА на порт виведення [7], тому таке рішення здійсненне.

Як результат передбаченого використання дискретних виходів Arduino для живлення термopідвісів, виникає обмеження на алгоритм збору даних: недопустимо використовувати команди групового запуску датчиків ds18b20 по протоколу 1-wire [8], оскільки їх одночасне спрацювання призведе до виходу з ладу обладнання збору інформації. Як наслідок, тривалість опитування всіх датчиків у такого модуля зростає, оскільки один датчик активно вимірює температуру до 750 мс [6]. З урахуванням затримок на передачу й обробку даних отримаємо приблизно 1 секунду на датчик, 35 секунд на термopідвіс, 420 секунд (7 хвилин) на всі термopідвіси блоку збирання інформації. Вважаючи, що вимірювання температури зерна виконуються 1 раз на день, або 2-3 рази у особливо несприятливих чи відповідальних випадках, така затримка несуттєва. Але бажане більш оперативне надходження даних з точки зору зручності для оператора системи термометрії, особливо з урахуванням вірогідності помилок протоколу.

Узявши за основу структурну схему на рис 1., необхідно визначитись з прикладними протоколами вимірювання температури системою. Існуючі впровадження тут використовують три підходи:

- зберігання адрес датчиків у базі даних АРМ з передачею кожної індивідуальної адреси датчика для читання раніше отриманих даних з температури спочатку через загальні канали зв'язку, а потім через 1-wire;

- зберігання адрес датчиків у базі даних не АРМ, а стандартного промислового контролера, використаного як збірковий контролер більш високого рівня. Такий контролер автономно й цілодобово взаємодіє зі збірковими контролерами термopідвісів, аналогічно п.1;

- збіркові контролери термopідвісів самі зберігають адреси датчиків і автономно цілодобово отримують від них значення температури.

Підхід 1 видається найбільш простим і дешевим, але має суттєві недоліки. Повна процедура керування вимірюванням температури має бути реалізована на АРМ, що не завжди просто реалізувати наявними засобами SCADA, і тому доведеться використовувати універсальні інструменти програмування, що є відступом від концепції стандартних засобів, оскільки програма у результаті буде унікальною і прив'язаною до конкретного виробника. Окрім того, взаємодія зі збірковими контролерами буде послідовною, тому на 10 збіркових контролерів матимемо відповідно не 7, а 70 хвилин, що знижує оперативність отримання інформації. Вихід – тримати АРМ термометрії постійно включеним, що не завжди можливо.

Підхід 2 дозволяє інтегрувати систему термометрії до засобів SCADA і стандартного промислового обладнання, інформація по температурам може бути надана відразу по запиті, але у випадку знеструмлення системи проблема низької швидкодії залишається. Потрібно постійно жити тепер вже промисловий контролер. До того ж, додатковий компонент призводить до здорожчання технічного рішення. Тому підхід 2 може бути рекомендований лише як шлях модернізації раніше впровадженої системи, коли повністю проблема стандартизації технічних засобів не вирішується.

Підхід 3 прискорює накопичення даних на збіркових контролерах термopідвісів, оскільки тут вони можуть працювати автономно й паралельно. Тому тут у випадку включення після знеструмлення інформація відновиться за ті ж 7 хвилин. А за цілодобового режиму роботи інформація надається за запитом. Однак, тут є і недолік – адреси відповідних датчиків повинні бути прописані у контролери термopідвісів або на етапі впровадження системи на об'єкті, або безпосередньо під час запуску. Проблема вирішується шляхом використання модулів розширення Arduino для промислових мереж, та створення відповідного програмного забезпечення. Але виникає обмеження: потрібно забезпечити обсяг пам'яті даних на збірковому контролері, достатній для зберігання всіх адрес датчиків, що можуть до нього підключатись, а також всіх значень температури.

Виразуємо необхідний об'єм пам'яті для зберігання адрес. Якщо використано 12 термopідвісів по 35 датчиків, то це 420 адрес. Адреса ds18b20 має довжину 8 байт [6]. Отже, потрібно обрати контролер Arduino, який має обсяг пам'яті 3360 байт або 3,3 КБайт не враховуючи оперативну пам'ять для обчислень та зберігання даних з термометрії. Відразу можна виключити з аналізу моделі Arduino NANO, MINI, UNO R3, Leonardo та аналоги, оскільки тут загальний обсяг пам'яті даних не перевищує 3,5 КБайти [7], а використання

EEPROM невиправдано ускладнює програмне рішення. Натомість Arduino UNO (оригінал), Arduino MEGA R3 та Arduino Due мають, відповідно 8 КБайт, 4 КБайти та 512 КБайт [7] флеш-пам'яті даних, що дозволяє їх використати відповідним чином.

Виразуємо необхідний об'єм пам'яті для зберігання значень температури. Датчики ds18b20 повертають 12-бітні значення, які потребують, відповідно, по 2 байти для зберігання. Оскільки опитуватись має максимум 480 датчиків, потрібно 960 байт або 0,94 КБайт оперативної пам'яті. Варіант з флеш-пам'яттю не розглядаємо, оскільки потрібна оперативна інформація, енергонезалежне зберігання якої опційне. Вважаючи на вартість Arduino UNO (оригінал) та Arduino Due [7], доцільно розглянути саме варіант використання Arduino MEGA R3 (китайська локалізація Arduino MEGA) в якості складової до блоку збирання інформації з термopідвісок. Тут є 8 КБайт оперативної пам'яті, чого цілком достатньо і для накопичення даних з термометрії, і для виконання обчислень [7].

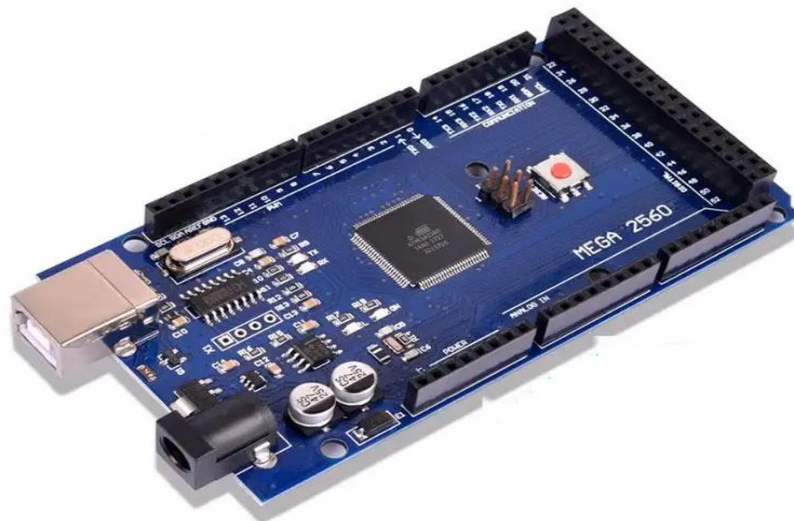


Рис. 2. Зовнішній вигляд плати Arduino MEGA 2560 R3

Виходячи з існуючих розробок і рекомендацій до програмно-технічної взаємодії Arduino з датчиками інтерфейсу 1-wire, існує три способи підключення датчиків до плати: з використанням UART, через спеціальний контролер 1-wire чи бітгандінг [8]. У прийнятій моделі є 4 порти UART[6] за потреби підключення 12 термopідвісів. В цьому випадку підключення по три термopідвіси на порт теоретично хоча й можливе, але може дати сумарну довжину лінії 1-wire більше 100 метрів, що небажано, оскільки це її верхня межа за умови використання кабелю UTP 5-ї категорії. Саме з цього кабелю виготовляється шина датчиків у термopідвісі і саме він використовується далі для комутації з блоком збирання інформації. Розглядати спеціальний контролер 1-wire також немає смислу, оскільки з одного боку це додаткове обладнання, а з іншого боку є ще технологія бітгандінгу.

Бітгандінг – технологія імітації послідовного каналу зв'язку із залученням звичайних дискретних входів і виходів. У Arduino є така можливість і апаратно, і на рівні підтримки готовими програмними бібліотеками [8], що робить такий шлях вирішення проблеми найбільш перспективним.

Arduino MEGA має 54 контакти для введення-виведення, але варто пам'ятати, що вони універсальні й багатофункціональні [9]. Тобто заздалегідь від цієї кількості потрібно відняти 6 контактів на зв'язок з модулем промислової мережі, що також має ввійти у комплект блоку збирання інформації тому маємо 48 контактів.

Визначимо кількість виходів, що мають бути задіяні для живлення термopідвісу на 35 датчиків. Раніше було показано, що під час опитування датчиків термopідвісу буде спожито струм не більше 38 мА. Один вихід Arduino MEGA може видати до 50 мА, чого цілком достатньо для живлення. Таким чином, на кожну з 12 термopідвісок прийдеться по 3 виходи: 1 вихід на живлення термopідвісу, 1 вихід на передачу команд датчику, 1 вхід на прийом даних від датчика. Отже, контактів в Arduino MEGA достатньо, буде задіяно 36 контактів з 48 доступних, лінія нульового провідника для всіх термopідвісів загальна і тут не враховується. У резерві ще залишається 12 контактів, що можуть бути використані для додаткового обладнання у порядку вдосконалення системи.

У результаті можливо запропонувати принципову схему для блоку збору інформації системи термометрії, показано на рис. 3.

Щодо модуля зв'язку (модуля промислової мережі) на рис. 3, то тут можна запропонувати варіант кількох рішень:

- Ethernet-модуль ENC28J60 [9] або Wi-Fi модуль ESP8266 [10], які підключаються через SPI. Їх можливо використати для підключення до мереж Ethernet, Profinet, Ethernet RTU, включаючи безпроводні варіанти;
- Модуль MAX485 UART-RS485 [10] чи подібні, підключається через UART. Можливо використати для підключення блоку збирання інформації до промислових мереж ProfiBUS, ModBus.

Рішення, показане на рис. 3, передбачає певні рекомендації до програми блоку збирання інформації від цифрових термодатчиків, а саме: блоки збирання інформації мають працювати автономно й цілодобово, здійснюючи індивідуальне почергове підключення живлення окремих термopідвісів, а також почергові запуск і отримання даних від кожного датчика термopідвіса. Для цього в енергонезалежну пам'ять кожного Arduino блоків збирання інформації мають бути прописані їх адреси. Для спрощення монтажу й обслуговування системи бажано, щоб прикладний протокол зв'язку між блоком збирання інформації й АРМ термометрії включав не лише команди отримання даних з температури але й команди перевизначення адрес датчиків з відповідною програмною підтримкою на АРМ термометрії та з боку блоку збирання інформації.

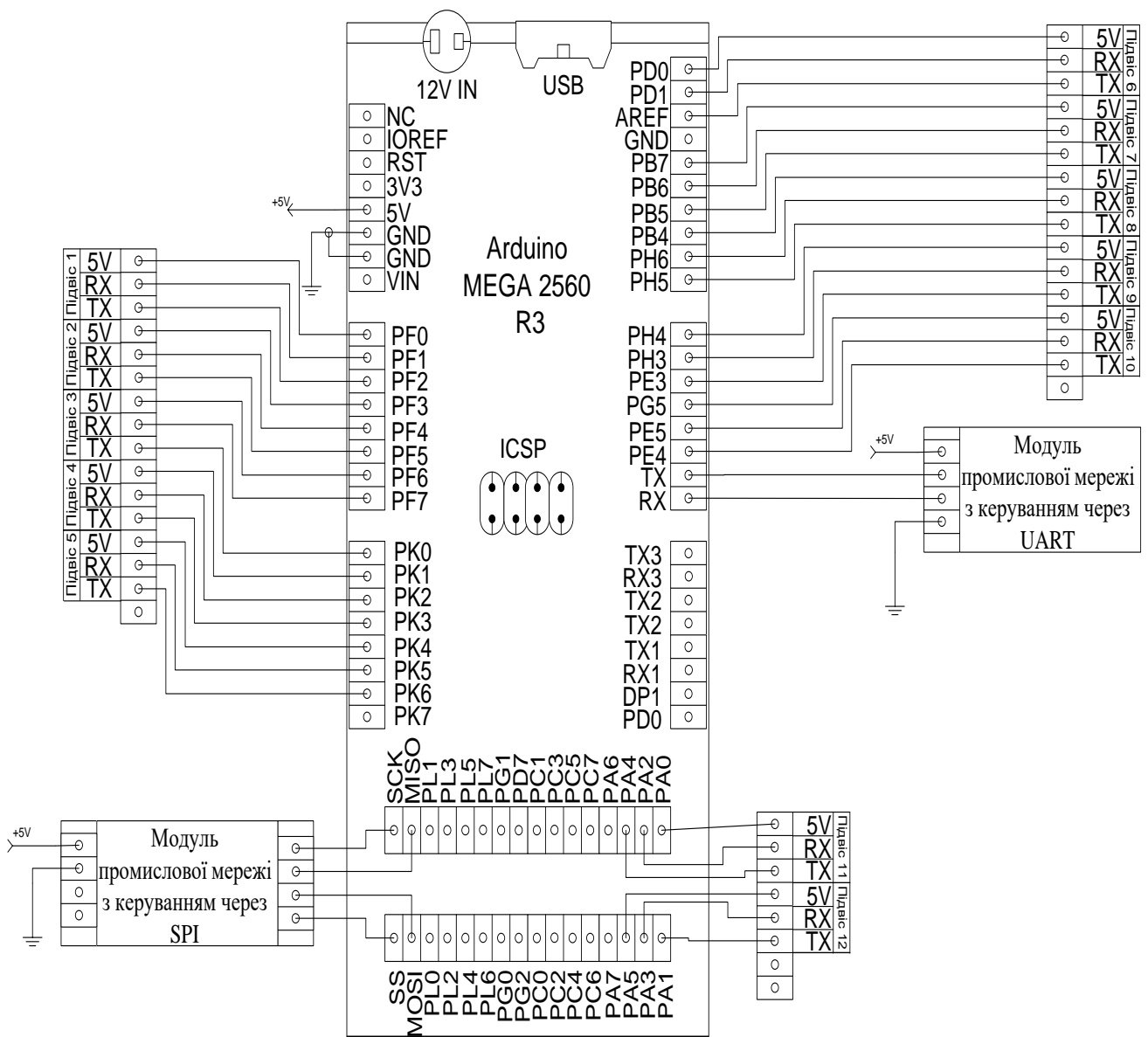


Рис. 3. Використання обладнання Arduino в якості блоку збирання інформації системи термометрії

Наукова новизна полягає в обґрунтуванні можливості заміни унікального обладнання багатоточкової термометрії зерна на зерносховищах типовими технічними засобами й компонентами, придатними для інтеграції у керуючі та обчислювальні системи підприємства. Також обґрунтовано рекомендації з функцій програмного забезпечення засобів збирання інформації від термопідвісів.

Висновки. В результаті було запропоноване загальне технічне рішення для системи цифрової термометрії зерна, придатне для використання у сховищах зерна складського та силосного типів. Його відмінність і перевага порівняно з аналогічними системами – використання серійних контролерів Arduino MEGA, які випускаються у досить великій кількості кількома

зарубіжними виробниками, що в умовах воєнного конфлікту в Україні дозволяє елеватору-споживачу систем термометрії не орієнтуватись на місцевого виробника унікального обладнання, а звернути увагу на інтегратора готових серійних рішень. Застосування у запропонованому рішенні кількох варіантів модулів зв'язку дозволяє гнучко застосовувати шинні, зіркові та бездротові топології в залежності від ситуації на відміну від шини, яка використовувалась раніше. Крім того, інтерфейси цих модулів сумісні з обладнанням і програмним забезпеченням промислових мереж, ідеологіями OPC й SCADA, що полегшує їх інтеграцію у систему керування виробництвом.

ПЕРЕЛІК ПОСИЛАНЬ

1. Просяник А.В., Соснін К.В.. [Основні варіанти вибору системи термометрії елеватора // «Хранение и переработка зерна» № 3 март 2008 стр. 29-30.](#)
2. <http://dnvpeldorado.com/index.php/ua/seeds-elevators> Обладнання для елеваторів зберігання зерна та зернопродуктів на базі сучасних систем термометрії з використанням цифрових датчиків температури фірми Dallas Semiconductor у складі термопідвісок. – ДНВП "Ельдорадо", сайт асоціації "Ельдорадо", (дата звертання 20.11.2022).
3. Просяник А.В., Хом`як В.С.. Практична термометрія в зерносховищах з пласкою і похилою підлогами// «Хранение и переработка зерна» № 4 березень 2003 р. ст.43-45.
4. Ткаченко С.М., Маслов А.В.. [Питання іскробезпеки при вимірюванні температури в силосах елеваторів](#) //Національний гірничий університет. Збірник наукових праць. – 2012. – №39 стор. 136-141.
5. Завод по переробці олійних культур на території ПП "Оліяр", в с. Ставчани, Пустомитівського району, Львівської області. Альбом №4-АТХ.ТМ. Система контролю температури шроту. Робочий проект. – ТОВ "Інтерпроект-Інжиніринг". – 2012. – 29 с.
6. <https://html.alldatasheet.com/html-pdf/433921/MAXIM/DS18B20/150/1/DS18B20.html> DS18B20. Programmable resolution 1-wire digital thermometer. – Maxim Integrated Products Inc. 2008. – 22 р. (дата звертання 20.11.2022).
7. <https://uamper.com> Інтернет-магазин "Мікроампер". (дата звертання 20.11.2022).
8. <https://docs.arduino.cc/learn/communication/one-wire> 1-Wire Protocol. – Arduino Community. – 19.11.2022. (дата звертання 20.11.2022).
9. <https://docs.arduino.cc/static/44e8456396e518cb86482519ac85bc92/A000067-datasheet.pdf> Arduino MEGA 2560 Rev3. – "Arduino". – 2022. – 17 р. (дата звертання 20.11.2022).
10. <https://arduino.ua> Інтернет-магазин "Arduino.ua". (дата звертання 20.11.2022).

АНАЛІЗ ТА ОПТИМІЗАЦІЯ СИСТЕМИ ІНТЕРНЕТ ТОРГІВЛІ В УМОВАХ СУЧАСНОГО РОЗВИТКУ РИНКУ ЮВЕЛІРНИХ ВИРОБІВ

Анотація. Описано основні вимоги до покращення технологій аналізу ринку збуту, збільшення клієнт-орієнтованого продукту за допомогою сучасних методів онлайн аналізу в умовах інтернет торгівлі, розробка та вдосконалення аналітики Інтернет магазину за допомогою сучасних технологій

Ключові слова: управління, проектний аналіз, онлайн аналітика, інформаційні потоки, аналітичний підхід, маркетинг, онлайн торгівля.

Вступ. Умови функціонування сучасної економіки, які характеризується гострою конкуренцією, вимагають від керівництва компаній постійної модернізації бізнес-процесів підприємства, використання інноваційних технологій. Впровадження комплексу Інтернет-маркетингу дозволяє сучасним українським підприємствам автоматизувати процес взаємодії з клієнтами, надаючи індивідуальне обслуговування в зручний час, ефективно організувати зворотний зв'язок, оперативно отримувати маркетингову інформацію і на цій основі приймати ефективні, обґрунтовані маркетингові рішення [1], проводити високоефективні рекламні кампанії. Отже використання новітніх технологій Інтернет-маркетингу сприяє підвищенню конкурентоспроможності підприємства.

При цьому покращення технології аналізу ринку збуту, збільшення клієнт-орієнтованого продукту за допомогою сучасних методів онлайн аналізу в умовах інтернет торгівлі та вдосконалення Інтернет магазину за допомогою сучасних мов програмування, це є основа сучасних підприємств та розвиток економіки країни [2].

Основний зміст роботи. Веб-аналітика – безперервний процес зі збирання та вимірювання даних, необхідних для оптимізації та покращення роботи. Сам собою інструмент не вирішує проблеми, а лише надає інформацію. Її потрібно правильно інтерпретувати: визначити недоліки та слабкі місця ресурсу, вивчити поведінку, потреби цільової аудиторії. На підставі результатів вносяться правки до стратегії та приймаються серйозні бізнес-рішення [3].

Ключові завдання веб-аналітики:

- оцінка якості трафіку;
- виявлення технічних помилок та інших недоліків на сайті;
- формування моделі поведінки та портрета цільового клієнта;
- пошук ефективних методів підвищення конверсії;
- визначення найвигідніших маркетингових каналів;

- зниження вартості залучення покупця;
- відстеження актуальних тенденцій ринку;
- збільшення прибутку з клієнта;
- порівняння показників ресурсу з лідерами у ніші.

Моніторинг дає зрозуміти, що зараз відбувається на сайті, чи окупаються вкладення у розробку, обслуговування, SEO-просування, підтримка репутації, різні види реклами. Без цього інструмента робота здійснюється наосліп. Підприємець не може побачити, від чого йде максимальна віддача, що потребує покращень, у якому напрямку краще рухатися. Тому налаштувати аналітику потрібно з моменту запуску сайту.

Аналітичні дані необхідні:

1. Керівнику бізнесу. Для більшого розуміння принципів роботи, контролю за роботою колективу, управління розвитком сайту.
2. Начальникам відділів продажу. Для розробки та подальшого коригування планів щодо реалізації товару.
3. Інтернет-маркетологів. Для аналізу маркетингових кампаній та каналів трафіку, створення рекламних стратегій, перевірки зручності сайту, пошуку та усунення проблем.

Багато новачків в e-commerce недооцінюють значення веб-аналізу. Якщо в інтернет-магазині немає системи аналітики або інструмент працює некоректно, це позбавляє бізнес реальних вимірних перспектив. Те саме стосується і ситуацій, в яких аудит виконується вкрай рідко або в штаті немає співробітника, який вміє аналізувати дані та робити грамотні висновки.

Основні типи, метрики та базові поняття аналізу.

У web-аналітиці передбачено кілька різних методів збирання інформації. Вони відрізняються цілями та об'єктом дослідження. Метою може бути зростання відвідуваності, підвищення конверсії, збільшення обсягів продажу. Виходячи з цього, встановлюються конкретні KPI для відстеження стану справ та внесення корективів, які покращують роботу всього сайту чи окремих напрямків.

Можливі об'єкти дослідження:

1. Відвідуваність ресурсу – якість трафіку, активність та кількість унікальних відвідувачів, причини спаду та підйому відвідуваності.
2. Цільова аудиторія – поведінка, географія, інтереси, демографічні дані, використовувані браузерери та пристрої.
3. Юзабіліті – шлях користувача, найпопулярніші сторінки, карти кліків, скролінгу.
4. Технічний стан майданчика – наявність сторінок-дублів, швидкість завантаження, помилки під час переходів.
5. Основні показники електронної комерції – найбільш ходові товари, кількість операцій, середній чек, цикл продажу, прибуток та виручка, ліди, дзвінки.

6. Джерела трафіку – продуктивні рекламні канали, популярні ключові фрази та слова, сторінки входу/виходу, вартість залучення покупця.

7. Конкуренти – обсяг ринку, семантичне ядро, джерела залучення нових клієнтів, контент, асортиментна картка.

Процес складається з визначення конкретних завдань, постановки KPI, збору та аналізу інформації, вироблення чітких регламентів та рекомендацій. За цим слідує внесення змін та оцінка результату.

Умовно веб-аналітику поділяють на 3 рівні.

На початковому рівні розраховують та аналізують:

- трафік – кількість відвідувань у межах доби, тижня, місяця;
- перегляди – кількість сторінок ресурсу, які відкрив або оновив користувач;
- глибину перегляду – кількість сторінок, переглянутих за сеанс;
- сеанси – середню тривалість взаємодії з ресурсом;
- відмови – відсоток відвідувачів, які переглянули лише одну сторінку та не вчинили конверсійну дію;
- карти скролінгу, кліків – вимірювання кількості кліків, аналіз глибини прокручування сторінок;
- відвідування цільових сторінок (товарних карток);
- переходи із рекламних каналів – соціальних мереж, контекстної реклами, email, сторонніх майданчиків.

На етапі аналітики бізнесу приділяється увага таким основним параметрам:

- цільові дії на сайті – передплата, дзвінки, продажі;
- товарообіг – обсяг реалізації у грошах за звітний період;
- прибуток – різниця між доходами та витратами компанії;
- рентабельність – співвідношення отриманого прибутку до собівартості продукції;
- виторг – загальна сума виручених коштів;
- середній чек – співвідношення суми всіх замовлень до чеків;
- повернення – користувачі, які додали ресурс до закладок або запам'ятали домен;
- конверсія – співвідношення кількості цільових дій до загальної кількості відвідувань;
- показники продажів по кожному окремому менеджеру.

У процесі аналізу визначають, наскільки прибутковим є сайт і як збільшити фінансові показники.

Завдання останнього рівня – глибоке вивчення цільового покупця, розробка стратегії утримання. Для цього досліджуються такі метрики:

- CLV – прибуток, отриманий за весь період співпраці з клієнтом;
- CPA – ціна цільової дії;
- CPL – вартість ліда;

- CPC – вартість кліка;
- ROI – окупність вкладень.

Моніторинг цих показників допомагає краще зрозуміти клієнта, вивчити механізм ухвалення рішень про купівлю, спрогнозувати прибуток та раціонально розподілити бюджет на рекламу.

Якісна веб-аналітика для інтернет-магазину – це комплекс методів та інструментів, за рахунок яких проводиться всебічний аналіз. Процедура тісно пов'язана з CRM, системами обліку, коллтрекінгом, рекламними мережами. Для повноцінного дослідження потрібно вивантажувати звіти та аналізувати інформацію з усіх зазначених систем.

Огляд сервісів для веб-аналітики:

Статистика для аналізу збирається за допомогою лог-аналізаторів та лічильників.

Лог-аналізatori – внутрішні програми, які накопичують дані із сервера.

Лічильники – невеликі зовнішні програми. Працюють за таким принципом:

на сайт підвантажується картинка для збирання інформації;

отримані дані вносяться до бази;

статистика відвідувань стає доступною для перегляду на майданчику.

Крім кількості та тривалості сеансів, лічильники збирають дані про продаж, кліки та іншу корисну інформацію.

Онлайн-системи веб-аналітики відрізняються спрямованістю та спеціалізацією. Вони бувають платними та безкоштовними. Багато хто пропонує тестовий період для оцінки функціоналу. Перевагою цих інструментів є автоматизація отримання даних, а й наочність подання інформації. Графіки, діаграми, таблиці з динамікою змін допомагають сформулювати обґрунтовані гіпотези щодо подальшого розвитку.

Висновки. Таким чином – аналіз ринкового стану та формування вимог показує актуальність впровадження сучасної технології онлайн аналітики на основі Google Analytics в умовах сучасних інтернет магазинах компанії, що дозволяє використовувати отримані дані та формувати подальші рекомендації для відповідних відділів, а саме:

- Змінити цінову політику та провести рекламні акції на основі даних електронної торгівлі яка збільшить дохідну частини на основі;

- З'ясування, які товари продаються найкраще. Підтримка маркетинговими заходами товарних груп, зменшення витрат на рекламу на товар з низьким рейтингом продажу;

- За умовами зменшення кількості товарів на одну транзакцію нижче за бажану, впровадити або запропонувати більш вигідні знижки при купівлі кількох товарів або забезпечити безкоштовну доставку, якщо сума замовлення перевищує певний мінімум рентабельності;

- Отримати інформацію – скільки часу у клієнтів займає ухвалення рішення про покупку і скільки разів клієнт відвідав сайт компанії. Якщо клієнти

багато разів відвідує сайт перед придбанням товару, треба спростити процедуру замовлення, зробити функцію порівняння пропозицій з конкуруючими товарами.

ПЕРЕЛІК ПОСИЛАНЬ

1. А.В. Малієнко. Аналіз та формування методологічних основ управління циклічною економікою./ Управління проєктами. Перспективи розвитку проєктного та нейроменеджменту, інформаційних технологій управління, технологій створення та використання об'єктів права інтелектуальної власності: збірник наукових праць за матеріалами IV Міжнародної науково-практичної інтернет конференції (24-25 березня 2022 р.). – УДУНТ, УКРNET, НДІВ НАПрН України, Дніпро: Юрсервіс, 2022. С 66- 71.

2. Інтернет-маркетинг для керівника [Електронний ресурс] // Школа бізнесу «Управляй майбутнім»: Статті з управління підприємством. – 2013. – Режим доступу: <http://www.u-b-s.ru/publikacii/internet-marketing.html>

3. Інтернет ресурс: <https://analytics.google.com/>.

РОЗДІЛ 3

ПРОГРАМНІ ЗАСОБИ УПРАВЛІННЯ, ЗБОРУ, ОБРОБКИ І ПЕРЕДАЧІ ІНФОРМАЦІЇ

УДК 004.932:528.854

С.Л. Нікулін¹, В.Ю. Каштан¹, К.Л. Сергєєва¹, О.В. Коробко¹, М.П. Дік¹
¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ВПЛИВ ПОПЕРЕДНЬОЇ ОБРОБКИ СУПУТНИКОВИХ ЗНІМКІВ РІЗНИМИ МЕТОДАМИ НА ВИДІЛЕННЯ ЛІНЕАМЕНТІВ ПРИРОДНОГО ПОХОДЖЕННЯ

Анотація. Досліджується низка високочастотних фільтрів, фільтри для підвищення різкості та інші. Оброблені ними космічні знімки є основою виділення границь яскравості шляхом використання детектора Кенні. Для оцінки якості виділення використовують ряд спеціально розроблених критеріїв якості. За підсумками аналізу отриманих границь яскравості формулюються висновки про доцільність використання тих чи інших методів попередньої обробки космознімків, а також рекомендації, які здатні скоротити час на обробку та аналіз даних, а також підвищити достовірність одержуваних результатів.

Ключові слова: обробка зображень, космічні знімки, границі яскравості, високочастотні фільтри, лінеаментний аналіз.

Вступ. Розвиток апаратної та методологічної бази супутникових зйомок призводить до зростання ролі космічних методів вивчення земної поверхні. На даний момент у практиці наук про Землю питома вага мульти- та гіперспектральних зйомок постійно збільшується, що робить актуальним питання розробки нових методів обробки та аналізу космічних даних. Нині під час вирішення низки геологічних задач дедалі ширше використовуються методи лінеаментного аналізу зображень. Вони засновані на виділенні та подальшому аналізі лінеаментів – лінійних елементів космічного знімка. Вважається, що лінеаменти відповідають розламним структурам земної кори та їхнє вивчення дає можливість уточнювати геологічну будову територій, а також для виконання прогнозно-пошукових робіт.

Існує багато підходів до автоматичного виділення лінеаментів на знімках. Найчастіше на практиці використовується той, що полягає у попередньому визначенні границь яскравості знімка з подальшим виділенням лінеаментів автоматичними та напівавтоматичними методами. Для виділення границь яскравості використовуються різноманітні маскові алгоритми, найбільш прийнятним у тому числі вважається оптимальний детектор Кенні [1].

Для ефективного виділення лінеаментів важливе значення має попередня обробка знімка. Традиційно на практиці застосовується низка нескладних методів обробки зображень, як-от контрастування, корекція яскравості,

еквалізація гістограми космознімка. Разом з тим, у теорії обробки цифрових зображень існує велика кількість методів, які до цього часу не застосовувалися при лінеаментному аналізі, та ефективність яких не вивчена. Оцінці цих методів обробки зображень присвячена дана робота.

Метою роботи є виявлення найбільш ефективних методів обробки космічних знімків (КЗ) під час вирішення геологічних завдань методами лінеаментного аналізу.

Постановка задачі. Для досягнення поставленої мети в роботі сформовані і вирішені такі завдання:

- дослідити існуючі методи та алгоритми обробки зображень;
- вивчити існуючі методи попередньої обробки космічних знімків, що виконується у процесі лінеаментного аналізу;
- програмно реалізувати та дослідити ряд алгоритмів обробки зображень з подальшою оцінкою їх ефективності з точки зору вирішення завдання виділення лінеаментів на космічних знімках;
- зробити висновки про доцільність застосування тих чи інших методів обробки зображень під час вирішення геологічних завдань методами лінеаментного аналізу.

Вихідні дані та територія досліджень. Дослідження проводилися на матеріалах сейсмонебезпечної території Туреччини, яка займає площу понад 150000 км² і містить низку великих міст, у т.ч. Анкару.

Використовувався знімок із супутника TERRA MODIS з просторовою роздільною здатністю 300 м, та дані про епіцентри землетрусів (джерело – сайт earth-quake.usgs.gov), (рис.1)

Дослідження проводилися в середовищі геоінформаційної системи РАПІД [2], яка має розвинений блок лінеаментного аналізу.

Методика досліджень.

Експерименти проводились за такою методикою:

1. Реалізовані декілька методів обробки зображень у середовищі Python.
2. У середовищі ГІС РАПІД методом Кенні виділялися границі яскравості. Для методу Кенні вказувався ряд параметрів: значення порогів для верхньої та нижньої меж, вид маски, та інше [3].
3. Проводилася обробка отриманих карт границь яскравості, в результаті були отримані карти просторової концентрації границь яскравості.
4. За допомогою спеціального модуля ГІС РАПІД для кожної отриманої карти концентрації границь яскравості розраховувався показник, що характеризує характер виділених границь яскравості, - дисперсія значень гістограми, побудованої за значеннями щільності.
5. Оцінка якості виділення границь яскравості передбачає використання умов, що відбивають інтуїтивно-зрозумілі евристичні критерії якості виділення границь [4]:

- а) границі повинні виділятися чітко, і не мати великої кількості розривів;

б) повинні виділятися тільки основні границі, що мають достатню протяжність і чіткість, щоб уникнути ситуації, коли отримувана бінарна карта перевантажена численними 1-2х-піксельними фрагментами.

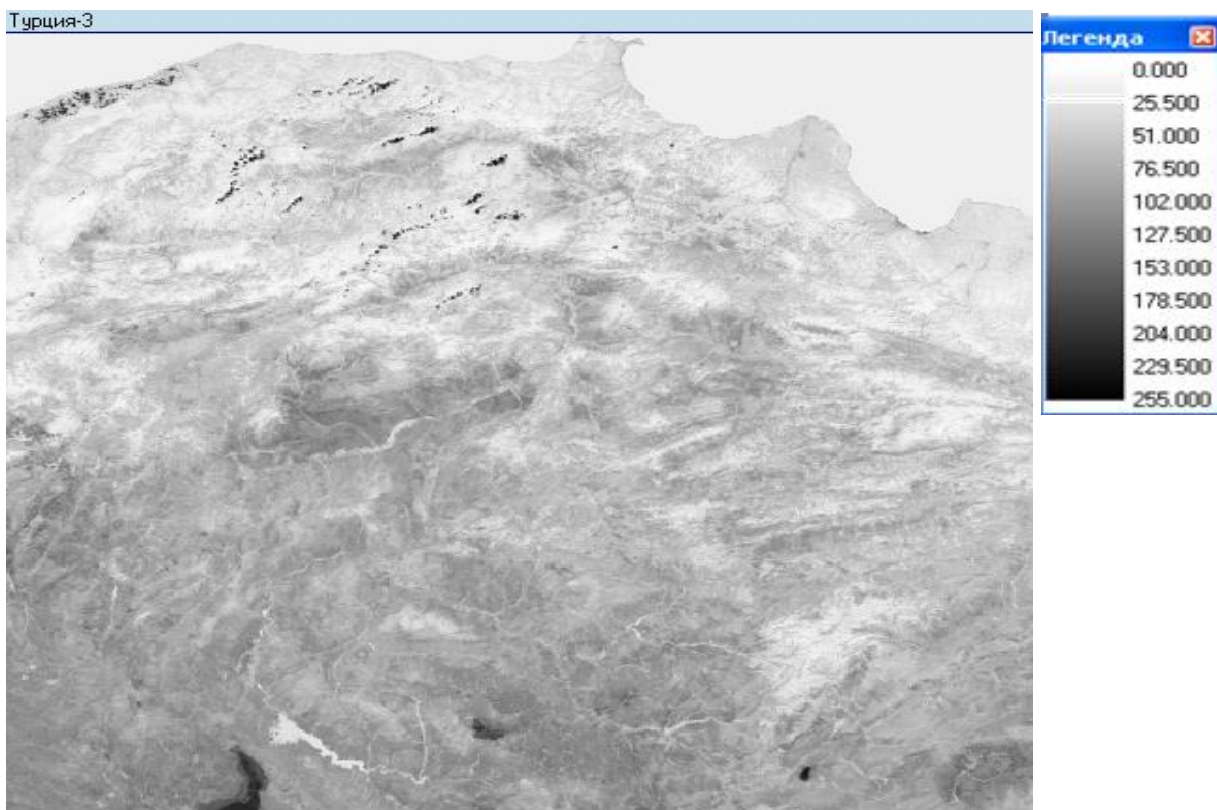


Рис. 1. Приклад вихідного геозображення сейсмонебезпечної території Туреччини (знімок із супутника TERRA MODIS з просторовою роздільною здатністю 300 м)

Зазначені вимоги є взаємовиключними – максимальне задоволення одного призводить до мінімального задоволення іншого. Тому на практиці найбільш продуктивний підхід з використанням декількох показників, що відображають ті чи інші характеристики границь, що виділяються. Їх значимість визначається фахівцем в залежності від конкретної задачі, що вирішується.

Використовуються такі показники: N – кількість одиничних (ненульових) пікселів, що відповідають границі яскравості на бінарному зображенні. Характеризує загальну кількість виділених границь. Чим вищий цей показник, тим більш детально виділені межі.

$$F = K/N,$$

де K – кількість безперервних фрагментів границь зображення. Цей показник характеризує ступінь фрагментації границь. Чим він нижчий, тим точніше виділені границі. Слід зазначити, підвищення показника N зазвичай веде до

небажаного зростання F . D – дисперсія гістограми значень щільності границь яскравості зображення. Значення показників N і F автоматично розраховувалися відповідним модулем ГІС РАПД.

6. Пункти 1-5 виконувалися для всіх зображень, отриманих за допомогою різних методів обробки вхідного зображення.

7. Показники N і F , упорядковувалися за зростанням і, залежно від отриманого порядкового номера, їм присвоювався ранг від 1 (найменше значення) до 10 (найбільше значення). За результатами ранжирування показників N і F був проведений аналіз досліджуваних методів. Вони стали основою для формулювання висновків та рекомендацій щодо застосування досліджених методів у процедурах лінеаментного аналізу.

Обговорення результатів. Обчислення показали, усі досліджені характеристики збільшують значення показника N в порівнянні з вхідним, необробленим, космічним знімком. Це означає, що застосування будь-якого з методів дозволяє збільшити кількість границь яскравості в порівнянні з вхідним. Про це свідчить і той факт, що застосування будь-якого методу призводить до збільшення дисперсії гістограми значень обробленого зображення в порівнянні з вхідним.

Підвищення кількості виділених границь, з одного боку, дозволяє виділити більшу кількість лінеаментів, а з іншого, веде до виділення значної кількості дрібних, несуттєвих деталей, коротких, уривчастих фрагментів границь, які суттєво ускладнюють вирішення багатьох практичних геологічних завдань. Тому велике значення має значення показника F , яке є своєрідною мірою цілісності одержуваних границь. Низькі значення F свідчать, що виділені границі яскравості є цілісними, протяжними і, як наслідок, мають геологічний сенс і значимість.

Тому особливу цінність мають методи обробки зображень, які з одного боку, підвищують значення параметра N при одночасному зниженні F . Доцільно виконувати їх пошук для кожної окремої ділянки, що має досліджуватись методами лінеаментного аналізу.

Висновки. В роботі було розроблено комп'ютерну методичку, що дозволяє на основі систем ГІС РАПД та Python обробляти та аналізувати зображення, оброблені різними методами; виконано реалізацію декількох методів обробки зображень у середовищі Python; виконано аналіз декількох методів обробки зображень з погляду ефективності їх застосування для вирішення задачі виділення границь яскравості; встановлено, що найкращі результати можуть бути отримані шляхом застосування до вихідних зображень фільтрів які забезпечують збільшення кількості виділених границь яскравості при одночасному збільшенні їх цілісності та протяжності; виявлено методи, застосування яких для попередньої обробки зображень недоцільно; отримані результати дають можливість скоротити витрати часу, необхідні для рішення геологічних завдань методами лінеаментного аналізу, і навіть підвищити достовірність одержуваних результатів.

ПЕРЕЛІК ПОСИЛАНЬ

1. Canny. J.F. A computational approach to edge detection / J.F. Canny // IEEE Trans. on Pattern Analysis and Machine Intelligence. – 1986. –N 8(6). – P. 679-698.
2. Бусигін Б.С. Спеціалізована геоінформаційна система РАПД: структура, технологія, завдання /Б.С.Бусигін, С.Л.Нікулін // «Геоінформатика», Київ. 2016. - № 1 (57). - С. 22-37.
3. Нікулін С.Л., Коробко О.В. Вплив параметрів алгоритму Кенні на результати виділення границь яскравості космічних знімків// Зб. наук. тр. Нац. гірн. ун-ту/редкол.: Г.Г. Півняк [та ін.] - Д.: Національний гірничий університет, 2013. - №1. - 51-52 с.
4. Півняк Г.Г., Бусигін Б.С., Нікулін С.Л. ГІС-технологія інтегрованого аналізу різномірних та різномірних геоданих. // Доповіді Національної академії наук України. - 2007. - №6 - С. 121-128.

УДК 004.932:528.854

К.Л. Сергєєва¹, В.Ю. Каштан¹, О.В. Коробко¹, Д.В. Іванов¹, І.С. Качан¹
¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

АНАЛІЗ СУПУТНИКОВИХ ДАНИХ ДЛЯ ОЦІНКИ ОСТРОВІВ ТЕПЛА УРБАНІЗОВАНИХ ТЕРИТОРІЙ

Анотація. Представлено складові інформаційної технології моніторингу ділянок підвищеного теплового фону урбанізованих територій за різночасовими даними супутникових зйомок, що дозволяє виявляти стабільні у часі ділянки теплового забруднення. Виконана апробація технології за часовим рядом зображень Landsat 8 TIRS 2013-2022 рр. території м. Дніпро.

Ключові слова: інформаційна технологія, острови тепла, яскравісна температура, Landsat 8.

Вступ. Процеси розвитку урбанізованих геотехнічних систем призводять до зміни міських ландшафтів, вирубки зелених насаджень та збільшення площі поверхонь зі штучних матеріалів, здатних накопичувати зовнішнє теплове випромінювання й підвищувати тепловий фон прилеглих територій, що становить потенційну небезпеку для навколишнього середовища та здоров'я населення.

Дані супутникового моніторингу у тепловому діапазоні електромагнітного спектру дозволяють здійснювати регулярний температурний моніторинг великих за площею територій з просторовим розрізненням, що перевищує розрізнення даних спостережень наземних метеостанцій, а також виявляти потенційно небезпечні для мешканців міські острови тепла (Urban Heat Islands, UHI) [1]. Явище UHI характеризує підвищення температури у центрі міста у порівнянні з приміськими територіями [2]. Інтенсивність локальних ділянок UHI залежить від багатьох чинників, таких як локальні нерівності рельєфу, альbedo поверхонь, наявність та площа ділянок рослинності, повітряні потоки та ін. [3].

На сьогоднішній день відкриті дані супутникового моніторингу температур земної поверхні доступні з сенсорів Thermal InfraRed Sensor (TIRS)

супутника Landsat 8 (просторове розрізнення – 100 метрів, часове розрізнення – 16 діб, період спостережень – з 2013 року); Advanced Spaceborne Thermal Emission and Reflection Radiometer (ASTER) супутника Terra (просторове розрізнення – 90 метрів, нерегулярна зйомка, період спостережень – з 2000 року); Moderate Resolution Imaging Spectroradiometer (MODIS) супутників Terra та Aqua (просторове розрізнення – 500 та 1000 метрів, часове розрізнення – 1 доба, період спостережень – з 2000 року) та ін. Наявність великого обсягу даних супутникових зйомок потребує створення нових методів та інформаційних технологій аналітичної обробки різночасових багатоспектральних космічних зображень для підтримки прийняття управлінських рішень щодо покращення міського середовища.

Метою роботи є створення складових інформаційної технології виявлення сталих у часі ділянок підвищеного теплового фону у межах міста Дніпро за даними супутникових зображень Landsat 8 у тепловому інфрачервоному діапазоні електромагнітного спектра.

Постановка задачі. Актуальною науково-прикладною задачею для досягнення мети роботи є використання методу виділення сталих островів тепла урбанізованих територій за часовими рядами даних супутникових зйомок, а також побудова картограм території м. Дніпро з виділеними стабільними у часі ділянками теплового забруднення.

Вихідні дані та територія досліджень. Тестова ділянка площею близько 360 км² включає територію м. Дніпро в межах її географічних границь та містить ділянки міської забудови й промислові території, зелені насадження (сквери, парки та ін.), водні об'єкти й невеликі за площею сільськогосподарські угіддя. Території сільськогосподарського призначення здебільшого не враховувалися у дослідженні з огляду на наявність теплових аномалій нагрітих сонцем відкритих ґрунтів та їх внесок у загальний розподіл температур.

Для досліджуваної ділянки сформований часовий ряд 46 зображень Landsat 8 з близьким до нуля відсотком хмарності за 2013-2022 роки. Дані надані сервісом ESA EO Browser (<https://apps.sentinel-hub.com/eo-browser>). Дати спостережень представлені у табл. 1.

Таблиця 1

Дати зйомки вихідних зображень Landsat 8										
	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
Травень		16, 25	12		1	4, 11, 27				6, 31
Червень	14, 30			22		5, 21	8, 15	10, 26		
Липень		28		17	20			3, 12	15	2
Серпень	26		7, 23		12, 21	8, 24, 31	27		16, 23	
Вересень		5, 30		10	22, 29		3	14, 21	8, 17	

Дослідження проводилося за даними, зібраними з травня по вересень за умови відсутності сніжного покриву для усунення його впливу на просторовий розподіл температур.

Методи досліджень. Температурний аналіз даних космічних зйомок базується на тому факті, що будь-який об'єкт земної поверхні випромінює тепло. "Сирі" дані теплових космічних зйомок (Digital Numbers, DN) перетворюються на спектральну енергетичну яскравість (Spectral Radiance), перераховану потім у значення яскравісної температури (Brightness Temperature, °C) і температури земної поверхні (Land Surface Temperature, LST). Перетворення DN даних у спектральну енергетичну яскравість виконується відповідно до виразу [4]:

$$F_{\lambda} = \frac{F_{\max \lambda} - F_{\min \lambda}}{Q_{\text{cal max}} - Q_{\text{cal min}}} (Q_{\text{cal}} - Q_{\text{cal min}}) + F_{\min \lambda} \quad (1)$$

де $F_{\min \lambda}, F_{\max \lambda}$ – кількість випромінювання, що надходить на сенсор і перетворюється, відповідно, у $Q_{\text{cal min}}, Q_{\text{cal max}}$ (Вт / (м² · ср · мкм));

$Q_{\text{cal min}}$ – мінімальне відкаліброване значення пікселів;

$Q_{\text{cal max}}$ – максимальне відкаліброване значення пікселів;

Q_{cal} – значення DN.

Значення параметрів $F_{\min \lambda}, F_{\max \lambda}$ каналів космознімка залежать від типу сенсора та наводяться у файлі метаданих.

Альтернативний вираз:

$$F_{\lambda} = \text{gain} \cdot Q_{\text{cal}} + \text{offset} \quad (2)$$

де $\text{gain}, \text{offset}$ – калібрувальні константи, наведені у файлі метаданих космічного зображення.

Отримані значення спектральної енергетичної яскравості перетворюються на яскравісну температуру (градуси Цельсія) відповідно до виразу [5]:

$$T_{\text{br}} = \frac{c_2}{\ln((c_1 \cdot \varepsilon / F_{\lambda}) + 1)} - 273,15 \quad (3)$$

де ε – коефіцієнт спектрального випромінювання земної поверхні;

c_1, c_2 – калібрувальні константи.

Будемо вважати, що ділянкою УНІ поверхні території міста є група пікселів супутникової карти температур за обрану дату, температура (T) якої перевищує просторове середнє температур (T_{mean}) плюс середньоквадратичне відхилення (T_{sd}). На підставі часового ряду даних про УНІ розрахований

інтегральний показник (UHI_prob) – статистична оцінка ймовірності появи УНІ, а також виділені границі зон сталих у часі УНІ зі значеннями $UHI_prob > 0.5$ (рис. 1).

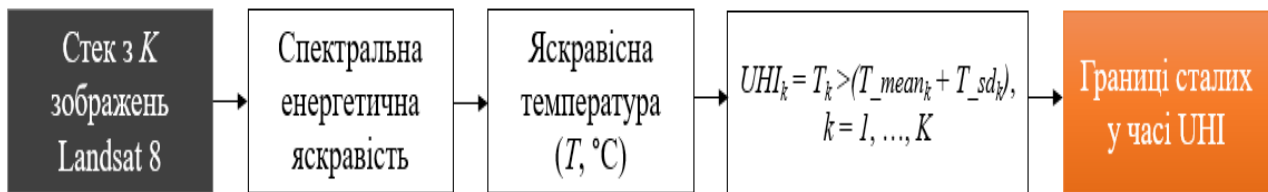


Рис. 1. Складові інформаційної технології виявлення сталих у часі УНІ

Обговорення результатів. Запропоновані складові інформаційної технології використані для картування ділянок УНІ, незмінних протягом останніх 10 років. Для досліджуваної території м. Дніпро побудовано різночасові картограми яскравісних температур (градуси Цельсія). Окремі картограми у псевдокольорах представлено на рис. 2.

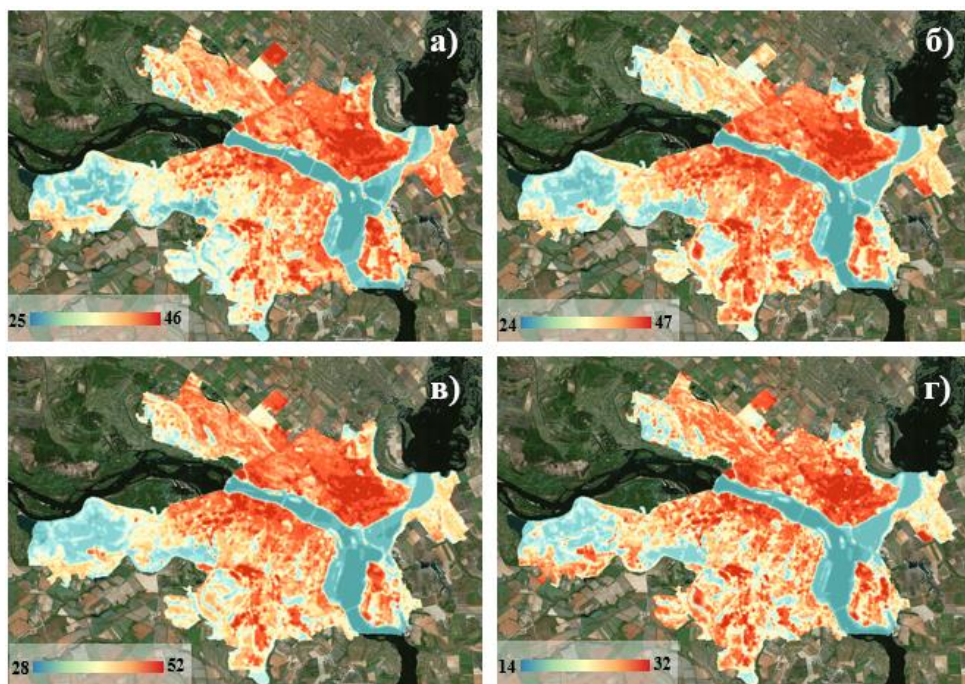


Рис. 2. Картограми яскравісних температур території м. Дніпро за даними Landsat 8: (а) 30.06.2013; (б) 22.06.2016; (в) 03.07.2020; (г) 06.05.2022. Базова карта: Bing Satellite

За результатами розрахунку інтегрального показника оцінки ймовірності появи УНІ (рис. 3, а) виділено границі зон УНІ, присутніх на більш ніж 50% зображеннях часового ряду супутникових спостережень (рис. 3, б).

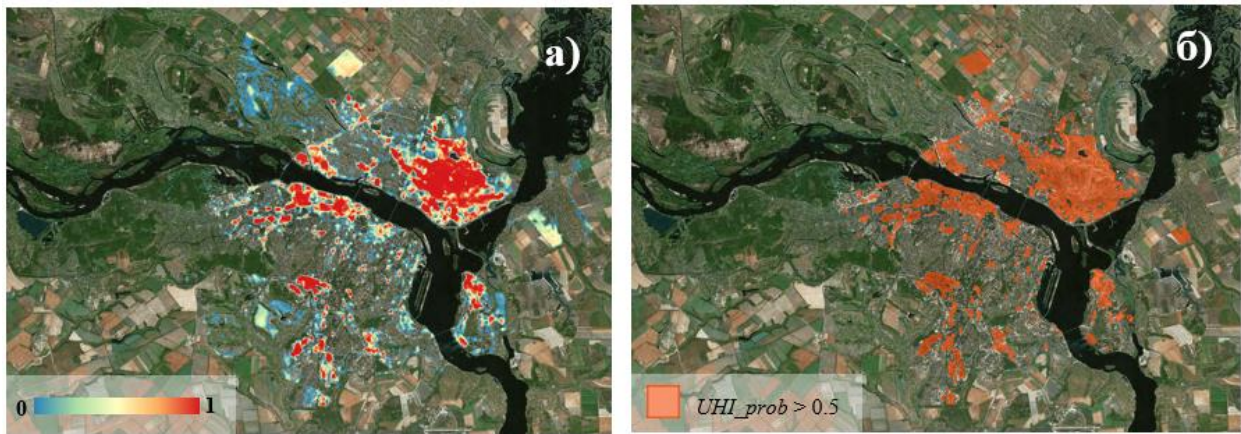


Рис. 3. Ділянки сталих у часі УНІ, 2013-2022 рр.: (а) оцінки ймовірності появи УНІ; (б) границі зон $UHI_prob > 0.5$. Базова карта: Bing Satellite

Виявлені ділянки сталих у часі УНІ здебільшого відповідають промисловим територіям м. Дніпро: Інтерпайп НТЗ, Південний машинобудівний завод ім. А.М. Макарова, ПрАТ "Дніпропетровський тепловозремонтний завод" та ін. Причиною появи температурних аномалій є нагріті штучні поверхні (асфальт, бетон), зони проведення технологічних процесів, що супроводжуються тепловим забрудненням прилеглих територій, та ділянки відкритого ґрунту.

Наукова новизна полягає у розвитку методу виділення сталих у часі островів тепла урбанізованих територій, що базується на результатах аналізу різночасових даних супутникових зйомок та дозволяє будувати картограми просторового розподілу зон підвищених температур.

Висновки. В роботі запропоновано складові інформаційної технології виявлення сталих у часі ділянок підвищеного теплового фону за різночасовими даними супутникових зйомок у тепловому діапазоні електромагнітного спектру. Застосування технології для аналізу теплового стану території м. Дніпро з використанням зображень Landsat 8 дозволило виявити ділянки підвищеного теплового фону – УНІ, що незмінно присутні на більшості наявних даних спостережень.

ПЕРЕЛІК ПОСИЛАНЬ

1. Lee Y.Y., Din M.F.M., Ponraj M., Noor Z.Z., Iwao K., Chelliapan S. Overview of urban heat island (UHI) phenomenon towards human thermal comfort // Environmental Engineering & Management Journal (EEMJ). – 2017. – Т. 16. – №. 9.
2. Kanga S. et al. Understanding the Linkage between Urban Growth and Land Surface Temperature – A Case Study of Bangalore City, India // Remote Sensing. – 2022. – Т. 14. – №. 17. – С. 4241.
3. Zhang Y., Wang Y., Ding N. Spatial Effects of Landscape Patterns of Urban Patches with Different Vegetation Fractions on Urban Thermal Environment // Remote Sensing. – 2022. – Т. 14. – №. 22. – С. 5684.

4. Gangopadhyay P.K. Application of remote sensing to identify coalfires in the Raniganj Coalbelt, India / P.K. Gangopadhyay, K. Lahiri-Dutt, K. Saha // International Journal of Applied Earth Observation and Geoinformation. – 2006. – Vol. 8. – P. 188-195.

5. Reddy S. N., Manikiam B. Land surface temperature retrieval from LANDSAT data using emissivity estimation // International Journal of Applied Engineering Research. – 2017. – Т. 12. – №. 20. – С. 9679-9687.

УДК 004.415.3:681.6

В.В. Спірінцев¹, А.Л. Ширін¹, І.Г. Гуліна¹, А.Ю. Прудченко¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ СТВОРЕННЯ ДИЗАЙНУ ІНТЕРФЕЙСІВ КОРИСТУВАЧІВ ВЕБ-САЙТІВ У СЕРЕДОВИЩІ FIGMA

Анотація. В роботі реалізовано демонстрування можливостей онлайн-сервісу для розробки веб-інтерфейсів Figma з використанням прикладів.

Ключові слова: *Figma, веб-дизайн, інтерфейс користувача, веб-сервіс, компоненти, інструмент.*

Вступ. У сучасному світі існує багато сервісів для розробки веб-дизайну, але кожен з них є індивідуальним. Веб-дизайн є сферою індустрії технологій, яка швидко розвивається завдяки цифровій трансформації та збільшенню кількості веб-додатків. Веб-дизайнери шукають інструменти дизайну, які пропонують інтригуючий досвід користувача. Figma – це унікальний інструмент веб-дизайну, який додає йому цінність з точки зору доступності, співпраці та унікальних функцій. Він має багато можливостей, які суттєво прискорюють та полегшують роботу веб-дизайнера.

Основний зміст роботи. Серед багатьох інструментів для розробки веб-дизайну, за останні декілька років, Figma вважається лідером у цій сфері. За результатами опитування дослідницької, консалтингової та маркетингової компанії Captterra Figma вважається найкращим інструментом для прототипування 2021 року. Зріст попиту на цей сервіс у 2017-2020 роках можна побачити на графіку, що наведено на рис.1.

У 2017 році InVision був найкращим інструментом для створення прототипів: 60% респондентів опитування UX Tool вказали, що вони використовували його. Лише за три роки їх частка користувачів впала до 23%, причому більше половини цих людей використовували його лише як допоміжний інструмент. Навпаки, Figma пережила вибухове зростання та популярність, яка зросла з приблизно 8% респондентів, які використовували інструмент у 2017 році, до приблизно 57% у 2020 році. [1]

З моменту запуску у 2015 році Figma нараховує сьогодні понад 4 мільйони користувачів по всьому світу. Понад 2,700 компаній довіряють

конструктивним особливостям Figma. Більше того, такі провідні компанії як Twitter, Rakuten, Volvo та Dropbox також приєдналися до родини Figma. [2]

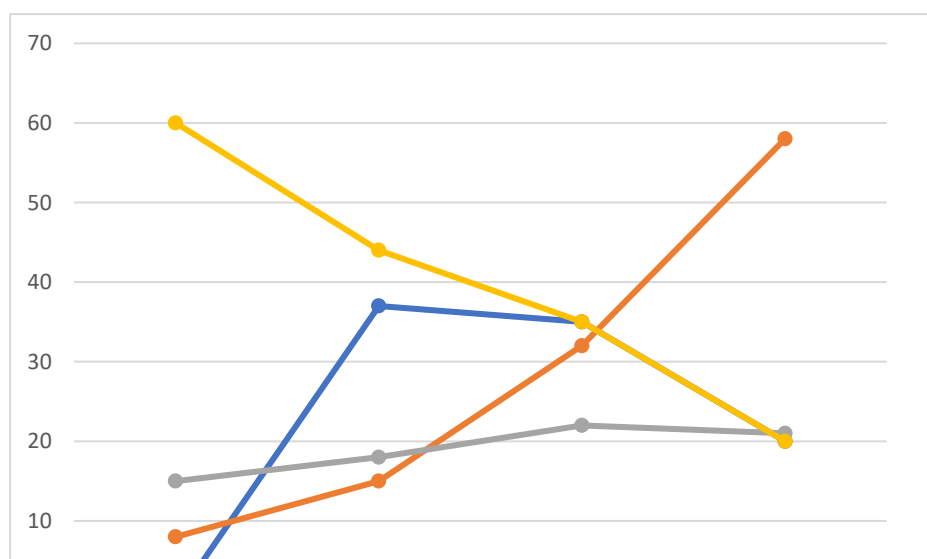


Рис. 1. Графік відсотків респондентів, які використовують платформу за даними ресурсу UX Tool

З метою отримання особистих даних щодо популярності Figma серед студентів Національного технічного університету «Дніпровська політехніка» було проведено опитування на тему: «Який сервіс для розробки веб-дизайну Ви вважаєте кращим та який Ви використовуєте?» (Опитування проводилось у жовтні 2022 року тільки серед студентів галузі знань «Інформаційні технології»). Інфографіку цього опитування наведено на діаграмі (рис. 2).

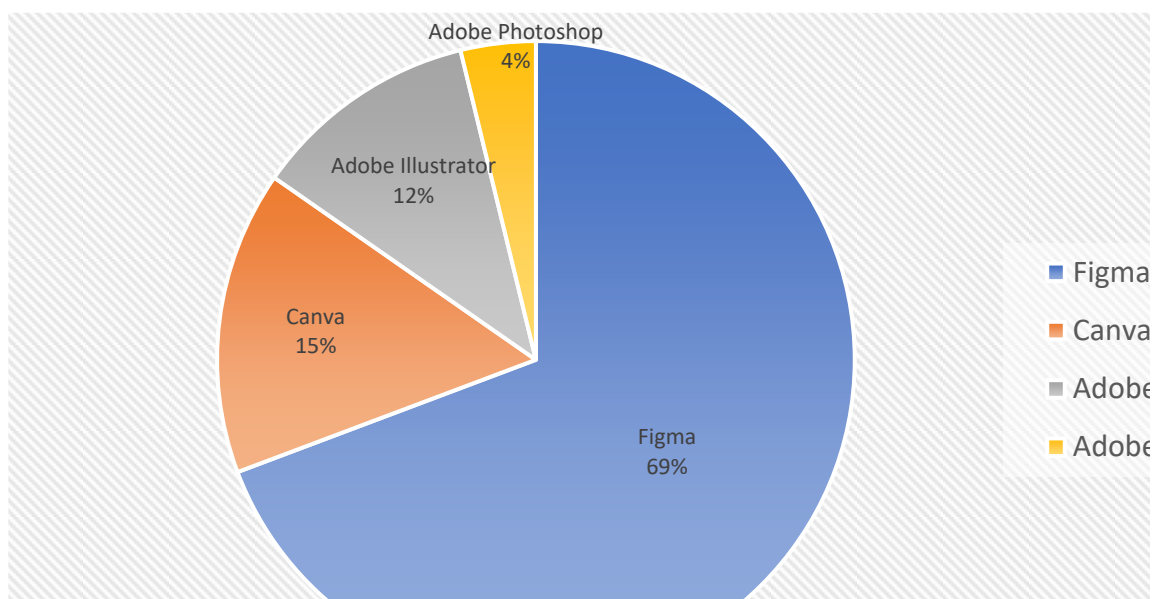


Рис. 2. Діаграма використання сервісів для розробки веб-дизайну (результати опитування студентів НТУ «Дніпровська політехніка»).

За результатами опитування було зроблено висновок, що найбільше студентів користуються платформою Figma. Нижче наведено переваги цього онлайн-сервісу:

- Кросплатформність. У Figma можна працювати на будь-яких носіях, браузерах та операційних системах через те, що вона знаходиться у хмарі.

- Вміння керувати процесами. Ви можете переглядати дизайн, вносити миттєві оновлення і отримувати зворотній зв'язок щодо ваших змін. Час між ітераціями може скоротитися з кількох днів до кількох хвилин, оскільки не потрібно витрачати час на завантаження або синхронізацію екранів, створення посилань для спільного доступу, надсилання повідомлень людям, щоб вони переглянули ці посилання, тощо. [3]

- Є можливість працювати дистанційно. Дизайнерам легше працювати паралельно, досліджуючи варіанти та ітерації з меншими кроками. Розробники можуть виявити і озвучити технічні проблеми раніше, ніж пізніше. Зацікавлені сторони, менеджери проектів або будь-хто, хто має зв'язок, можуть бачити, як дизайн розгортається від ідеї до відшліфованого візуального ефекту, а не чекати на велике відкриття.

- Проектування інтерфейсів. Figma дає можливість створювати інтерактивні прототипи. Можемо відкрити та подивитися на будь-якому носії, як буде виглядати додаток. [4]

- Компоненти. За допомогою Figma ви можете отримати доступ і змінити властивості будь-якого шару в компоненті, не від'єднуючи його від головного. Те саме стосується і вкладених компонентів. Тепер, коли ви змінюєте властивість шару у головному компоненті, ці зміни поширюватимуться лише на ті екземпляри, для яких ця властивість ще не була перевизначена.

- Фрейми. За допомогою Figma легше структурувати наші проекти таким чином, щоб відобразити, як ці проекти будуть кодуватися. Figma використовує фрейми замість артбордів. Фрейми відрізняються тим, що ви можете вкладати один фрейм всередину іншого. Коли ви розміщуєте менший фрейм над більшим, вони автоматично групуються, і менший фрейм стає дочірнім для більшого, батьківського фрейму. Дочірні фрейми розташовуються і обмежуються відносно батьківського фрейму. Використання таких фреймів корисно для розробників, оскільки фрейм у Figma подібний до контейнера в HTML. Коли розробники перевіряють дизайн, вони зможуть бачити елементи інтерфейсу, вкладені у відповідні контейнери, а це означає, що у них буде більш точний план, на який вони зможуть посилатися при написанні коду.

Висновки. Для веб-дизайнера важливі швидкість роботи над проектом та комунікація у процесі. Figma допомогла відмовитися від різних графічних програм та об'єднати весь процес створення інтерфейсу в одному інструменті. На відміну від інших платформ, Figma можна користатися без попереднього скачування. Немає необхідності постійно завантажувати, синхронізувати та впорядковувати файли – проекти автоматично зберігається у спільному

просторі в хмарі. Враховуючи усе вищевикладене можна впевнено вважати Figma лідером серед інструментів для роботи з інтерфейсами та веб-дизайном.

ПЕРЕЛІК ПОСИЛАНЬ:

1. [Wenzel M.](https://uxdesign.cc/the-precipitous-rise-of-figma-and-fall-of-invision-435f07e8d1b6), The precipitous rise of Figma and fall of InVision, 2021 [Електронний ресурс]. URL: <https://uxdesign.cc/the-precipitous-rise-of-figma-and-fall-of-invision-435f07e8d1b6> (дата звернення: 01.11.2022).
2. Potterman R., Canva vs Figma Review 2022 [Електронний ресурс]. URL: <https://www.everythingfreelance.com/vs/canva-vs-figma/> (дата звернення: 01.11.2022).
3. Pacifico M., Why We Should Be Using Figma, 2018 [Електронний ресурс]. URL: <https://blog.prototypr.io/why-we-should-be-using-figma-1510d0923be> (дата звернення: 01.11.2022).
4. Ким О., Відмовилися від 10 програм. Як графічний редактор Figma вдвічі пришвидшив роботу команди дизайнерів, 2021 [Електронний ресурс]. URL: <https://vctr.media/specials/vidmovylysy-a-vid-10-program-figma/> (дата звернення: 01.11.2022).

УДК 004.415.3:681.6

М.В. Петрига¹, Н.О. Соколова¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

РОЗРОБКА ВЕБ-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ ТАЙМ МЕНЕДЖМЕНТУ З ВИКОРИСТАННЯМ СТЕКУ ТЕХНОЛОГІЙ MERN

Анотація. Описано дослідження та створення інформаційної системи тайм-менеджменту для користування як звичайними користувачами, так і працівниками, з використанням стеку технологій MERN для розробки клієнтської та серверної сторони.

Ключові слова: веб-програмування, ефективність, тайм менеджмент, адаптивність, frontend, backend.

Вступ. Проблема швидкоплинності часу та його ефективного використання має глибоке коріння в історії людства. Ще за часів Стародавнього Риму, відомий мислитель та державний діяч Луцій Анней Сенека, розмірковував у своєму першому листі з «Моральні листи до Луцілія» як не витратити час: «Час – це найцінніший ресурс і єдиний ресурс, яким ви володієте. Будьте суворі зі своїм часом, знайте, на що витрачаєте свій час, і цінують, як інші приділяють вам свій час.» [1]. Сенека вів постійний облік часу в письмовому виді так само, як вів облік грошей.

Будуть доречними і слова бізнесмена та інвестора Роберта Кійосакі, що у кожної людини не більше часу, аніж у вас, але деякі встигають більше. Головний ключ – правильне управління часом, так як це інвестиція не лише в успішний розвиток робочих проєктів, а і особистий, що є невід’ємними речами [2].

Наразі пандемія надала більше вільного часу і більше простору та відповідальності для вміння його врегулювання. За даними навчальної платформи Prometheus, в період пандемії кількість активних слухачів онлайн курсів збільшилося в 4 рази [3]. І знову настала задача самодисципліни та зручних інструментів для самоорганізації.

Можемо побачити, що міркування та розуміння поняття часу має невід’ємну частину реалізації з його управління, тобто – тайм-менеджмент.

Тайм-менеджмент – дія або процес свідомого контролю за кількістю часу, витраченого на конкретну діяльність, що спеціально підвищує ефективність і продуктивність. Спочатку тайм-менеджмент означав лише ділову чи робочу діяльність, але з часом термін розширився, включивши також особисту діяльність. Система управління часом — це розроблена комбінація процесів, інструментів, прийомів і методів.

Системи управління часом часто включають веб-додаток, який використовується для відстеження робочого часу. Системи управління часом дають користувачам уявлення про їхню робочу силу, дозволяючи їм бачити, планувати й керувати часом. Це дозволяє контролювати і підвищувати продуктивність. Система тайм-менеджменту автоматизує процеси, що позбавляє від паперової тяганини та виснажливих завдань [4].

Наразі маємо дуже багато різних веб-додатків, націлених окремо на управління робочими проектами, планування особистого життя чи поєднуючи в собі ці дві сфери.

Постановка задачі. Мета роботи полягає у розробці системи зручного планування навчання, праці, особистих планів за допомогою MERN. Можливість поділу на групи та пріоритезація задач.

Для досягнення поставленої мети в роботі сформовані і вирішені такі завдання:

- обґрунтовано вибір стеку для розробки
- розроблено діаграму прецедентів
- розроблена серверна сторона додатку
- розроблено дизайн сайту
- розроблено клієнтська сторона додатку
- проведено тестування

Основний зміст роботи. Під час роботи були використані наступні методи та інструменти.

Для клієнтської сторони:

- інтерфейсний фреймворк Materializecss;
- бібліотека ReactJS для створення інтерфейсів користувача;
- синтаксичне розширення JSX для роботи з React;

Для серверної сторони:

- інструмент моделювання об’єктів MongoDB - Mongoose JS, призначений для роботи в асинхронному середовищі;
- HTTP-клієнт Axios для роботи з браузером та node.js;

Допоміжні системи для розробки та проектування:

- документо-орієнтована система керування базами даних MongoDB, яка не потребує опису схеми таблиць;
- розширення ThunderClient для VSCode для тестування API;
- швидкий веб-фреймворк Express.js для node.js;
- асинхронне середовище виконання Node.js;
- менеджер пакетів npm;
- система контролю версій Git;
- редактор для коду VSCode;
- патерн проектування MVC;
- графічний редактор Figma.

Обраний стек технологій для створення системи, означає розробляти і об'єднувати різні компоненти, що дозволяє пізніше додавати новий функціонал уникаючи конфліктів з вже створеною системою.

Була створена діаграма прецедентів (рис.1), в якій є 1 учасник – кінечний користувач системи.



Рис. 1. Діаграма прецедентів

Готовий проект має наступний функціонал:

- додавати нового користувача;
- додавати завдання;
- редагувати завдання;
- створювати групи для завдань;
- продивлятися інформацію про завдання;
- створювати групу користувачів;
- обмежувати функції користувачів, які були додані;
- можливість поєднати з Google Calendar;

- додати додаткову поштовий адрес;
- система нагадування;
- система аналізу прогресу;
- видалити завдання.

Система надає можливості створювати групи для працівників. В той самий момент, користувач не виходить зі своєї ролі, але набуває додаткової можливості обмеження функцій по редагування завдань інших користувачів. Це дозволить поєднувати робочі та особисті планування, так як не потрібно буде створювати нового користувача для робочого акаунту.

Також система містить систему аналізу прогресу, що уявляє собою збір інформації по затраченим годинам на певні завдання з особистого чи робочого планування та процент виконання планів. Це дозволить підвищити працездатність і аналіз ефективності працівника чи особистий.

Наукова новизна розробки полягає в реалізації базових методів тайм менеджменту, зокрема таймеру Pomodoro, у вигляді веб-додатку з можливістю розбиття завдання на групи та підгрупи за пріоритетами задач з сучасним актуальним мінімалістичним та зручним інтерфейсом з використанням стеку технологій MERN.

Висновки. В результаті роботи було створено інформаційна система тайм менеджменту для всебічного користування та багатьма функціями. Дану систему можна використовувати та вдосконалювати для запровадження функції тайм менеджменту працівника чи групи працівників у фірмах. Обраний стек технологій дозволяє вносити зміни не порушуючи цілу систему.

ПЕРЕЛІК ПОСИЛАНЬ

1. Thoughts on Moral Letters to Lucilius: Letter 1 - On Saving Time. <https://Oxedward.io/thoughts-on-moral-letters-to-lucilius-letter-1-on-saving-time/>
2. Маліновська О.Я. Конспект лекцій з навчальної дисципліни «Особистий, корпоративний та публічний тайм-менеджмент» / Маліновська О.Я. – Львів: ЛНУ, 2018. – 112 с.
3. Іван Примаченко про онлайн-навчання на карантині. <https://www.radiosvoboda.org/a/30582636.html>
4. Stella Cottrell. The Study Skills Handbook / Stella Cottrell. – Palgrave Macmillan, 2013. – 123 с.

Л.І. Мещеряков¹, С.І. Випанасенко¹, Н.С. Дрешпак¹, П.Р. Кунденко¹
¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ФРЕЙМВОРКУ REACT NATIVE ПРИ РОЗРОБЦІ МОБІЛЬНИХ ДОДАТКІВ

Анотація. Описано дослідження ефективності фреймворку при розробці мобільного додатку під операційні системи iOS та Android. Дослідження включає докладний аналіз технології, а також створення програмного пакету для тестування швидкодії додатку.

Ключові слова: мобільний додаток, кросплатформенний додаток, фреймворк, React Native, Javascript, Typescript, iOS, Android.

Вступ. На сьогодні є очевидною тенденція на зростання попиту на мобільні додатки та розробку мобільного програмного забезпечення. Кількість мобільних пристроїв, смартфонів, планшетів, смарт-годинників стрімко зростає та вже значно перевищує кількість класичних комп'ютерів.

Серед мобільних операційних систем перші позиції безперечно займають Android та iOS. Android є open-source операційною системою та використовується на більшості мобільних пристроїв у світі, його доля становить 71%, тоді як доля iOS становить 28%.

Існує багато програмних засобів створення мобільних додатків, але складність сучасної мобільної розробки полягає у тому, що дві домінуючі мобільні операційні системи потребують різних підходів та навіть мов програмування для розробки додатків. Однак створення програмного забезпечення це досі дорогий процес, і утримувати одразу дві команди розробки для того, щоб створити один додаток одразу під дві різні платформи, є досить важкою задачею для бізнесу. Саме відповідями на ці виклики стали такі не так давно створені фреймворки як React Native або Flutter, які дозволяють створювати кросплатформенні додатки.

Постановка задачі. Для досягнення поставленої мети в роботі сформовані і вирішені такі завдання:

- викласти існуючі методи та інструменти для розробки сучасних мобільних додатків;
- здійснити аналіз особливостей та архітектури фреймворку React Native;
- спроектувати та розробити окремий універсальний програмний пакет для оцінки швидкодії кросплатформенних мобільних додатків;
- спроектувати та розробити тестовий мобільний додаток, що реалізує шаблони сучасних додатків і зробити висновки щодо доцільності створення подібної системи.

Основний зміст роботи. Під час роботи над даною роботою були використані наступні методи та інструменти.

- мови програмування Typescript та Javascript, фреймворк React Native;
- інтегрована середовище розробки WebStorm;
- математичний метод інтерполяції.

На сьогодні провідними мобільними операційними системами є Android, що займає 70,96% ринку мобільних пристроїв, та iOS, що займає 28,43%. Також слід зазначити що на ринку мобільних операційних систем присутні KaiOS, Samsung OS, Windows Mobile тощо. Але їх загальна доля в цілому складає лише близько 1% ринку.

Тому слід зазначити, що під кросплатформним мобільним додатком у широкому розумінні вважають додаток, що працює на двох основних платформах Android та iOS.

Існує три основні підходи до створення мобільних додатків:

- створення окремих нативних додатків під кожен платформу;
- створення гібридного мобільного додатку;
- створення кросплатформного мобільного додатку.

Під нативним мобільним додатком, вважається додаток, що був розроблений під спеціальну платформу, у випадку мобільних додатків – Android або iOS, відповідно такий додаток може працювати лише на одній платформі.

Оскільки нативні додатки створено спеціально для конкретної операційної системи, вони забезпечують кращий користувацький досвід, ніж гібридні додатки. Нативні мобільні додатки зазвичай працюють і виглядають краще, ніж їхні веб-аналоги, які мають обслуговувати численні платформи. Крім того, нативні мобільні додатки мають доступ апаратного забезпечення пристрою, таких як датчики та камери, які недоступні через інтерфейс мобільного браузера.

Нативні мобільні програми, на відміну від веб-сайтів і веб-додатків, не працюють у браузері. Їх потрібно завантажувати з магазинів додатків для певної платформи, таких як Apple App Store і Google Play.

Нативні додатки створювати складніше, ніж веб-сайти для мобільних пристроїв. Однак не потрібно турбуватися про сумісність або поведінку браузера.

До переваг нативних додатків можна віднести:

- оскільки додаток створено та оптимізовано під конкретну платформу, то, як наслідок, додаток має високу швидкодію.
- оскільки додаток є завантаженим та встановленим на пристрій користувача, то всі необхідні ресурси та матеріали вже зберігаються на пристрої, що забезпечує швидкий старт додатку.
- оскільки середовище виконання нативного мобільного додатку є завчасно визначеною, на відміну від веб-додатку, то це робить нативний мобільний додаток безпечнішим.
- нативні програми написані спеціально для платформи та використовують всі можливості пристрою. Ці програми можуть отримати

миттєвий доступ до апаратного забезпечення пристрою, наприклад GPS, камери та мікрофона, що робить їх більш швидкими у виконанні. Push-сповіщення є ще однією значною перевагою вибору розробки нативної програми. Розробник матиме доступ до всіх API та інструментів, які пропонує платформа.

Втім у нативних мобільних додатків є і мінуси, найсуттєвішим з яких є вартість розробки. Оскільки додаток створюється під конкретну платформу, то, відповідно, його неможливо використати на іншій платформі. Це вимагає створення та утримання одночасно двох команд розробки, які будуть працювати над окремими додатками для Android та iOS. Також для уникнення програмних збоїв та помилок, оновлення та зміни у додатках повинні бути синхронізовані між собою. Отже, розробники потрапляють у ситуацію, коли в них є дві абсолютно різні кодові бази, проте оновлення яких повинні бути синхронними та налагодженими. Як наслідок, розробки нативних додатків може бути дуже дорогою у фінансових та часових ресурсах.

Наступним підходом до створення мобільних додатків є створення гібридного додатку. Це додаток, що поєднує в собі елементи нативного додатку і веб-додатку. Гібридні додатки – це, по суті, веб-додатки, які розміщено у нативній оболонці. Після того, як їх завантажено з магазину додатків і встановлено локально, оболонка може підключитися до будь-яких апаратних можливостей, які надає мобільна платформа, через браузер, вбудований у програму. Браузер і його плагіни працюють на сервері та невидимі для кінцевого користувача.

Гібридні додатки запускають код усередині контейнера. Механізм браузера пристрою використовується для відтворення HTML і JavaScript, а також власних API для доступу до апаратного забезпечення пристрою.

Незважаючи на те, що гібридна програма зазвичай має веб-подібні елементи навігації, чи зможе програма працювати в автономному режимі, залежить від її функцій. Якщо програмі не потрібна підтримка з сторонніх API, її можна змусити працювати в автономному режимі.

Гібридні додатки популярні, оскільки вони дозволяють розробникам написати код для мобільного додатка один раз і все одно підтримують кілька платформ. Оскільки гібридні програми додають додатковий рівень між вихідним кодом і цільовою платформою, вони можуть працювати дещо повільніше, ніж нативні або веб-версії тієї самої програми. Також оскільки гібридний додаток є фактично є веб-додатком, то елементи інтерфейсу будуть відповідати веб-елементам, а не конкретним елементам цільової платформи, що може негативно позначитись на користувацькому досвіді.

Отже, гібридний додаток вирішую найголовнішу проблему нативних додатків – вартість розробки, втім якість самого додатку буде дещо нижчою.

Останнім підходом до створення мобільних додатків є створення кросплатформенного мобільного додатку. Цей метод, як і метод гібридних додатків, вирішує головну проблему методу нативних додатків – дві окремі кодові бази під окремі операційні системи.

Методи роботи кросплатформенного мобільного додатку залежать від конкретного фреймворку, адже кожен кросплатформенний фреймворк реалізує роботу у свій спосіб. Найпоширенішими фреймворками для створення кросплатформенних додатків є React Native, Flutter, Xamarin.

Xamarin – це потужний і багатофункціональний фреймворк для розробки мобільних додатків. Його технологічний набір складається з C#, який дозволяє розробникам кодувати як для Android, так і для iOS однією мовою.

Програми Xamarin.Android компілюються з мови C# в проміжну мову (IL), який при запуску програми зазнає Just-in-Time-компіляції (JIT) в машинну збірку. Програми Xamarin.Android працюють у середовищі виконання Mono паралельно з віртуальною машиною середовища виконання Android (ART). Xamarin надає прив'язки .NET до просторів імен Android.* та Java.*.

Програми Xamarin.iOS проходять повну Ahead-of-Time-компіляцію (AOT) з мови C# у свій код збірки ARM. Xamarin використовує селектори для надання коду Objective-C керованого коду C# і Registrars для надання керованого коду C# коду Objective-C. Селектори і Registrars разом називаються «прив'язками» і забезпечують взаємодію між Objective-C і C#.

Flutter – це кросплатформенний фреймворк для розробки мобільних додатків, створена Google. Flutter набуває все більшу популярність через те, що він забезпечує узгоджену взаємодію з користувачем на всіх пристроях, і його легше підтримувати та налагоджувати.

Flutter розроблено як розширювану багатошарову систему. Він створений як ряд бібліотек, кожна з яких залежить від базового рівня. Жоден рівень не має привілейованого доступу до нижнього рівня, і кожна частина рівня структури створена як необов'язкова та замінна.

Для цільової операційної системи програми Flutter упаковані так само, як і будь-яка інша нативна програма. Спеціальна платформа для вбудовування забезпечує точку входу, координує роботу з базовою операційною системою для доступу до таких служб, як рендеринг інтерфейсу, доступність і ввід даних, і керує циклом подій повідомлення.

Flutter не потребує жодних специфічних для платформи компонентів інтерфейсу користувача для відтворення свого інтерфейсу. Це означає, що немає потреби в трансформуванні відображення властивостей для відображення кожної анімації у віджет певної платформи. Втім Flutter використовує доволі непопулярну мову програмування Dart, що робить старт роботи з ним дещо складнішим, порівнюючи з іншими фреймворками.

React Native – це фреймворк JavaScript для написання мобільних додатків для iOS і Android, що відображаються нативно. Він заснований на React, наслідує всі його підходи і можливості, але замість того, щоб націлюватися на браузер, він націлений на мобільні платформи. React Native дозволяє легко одночасно розробляти як для Android, так і для iOS.

Оскільки React Native базується на бібліотеці React, то необхідно виділити її основні особливості роботи.

Virtual DOM – це полегшена копія реального DOM, що призначена для підвищення ефективності маніпуляцій з реальним DOM. Щоразу, коли змінюється стан будь-якого компонента, React оновлює своє віртуальне дерево DOM. Оскільки віртуальний DOM є полегшеною версією реального, то оновлення віртуального DOM не займає багато часу. React підтримує два віртуальних DOM кожного разу, один містить оновлену віртуальну DOM, а інший — попередню версію до оновлення. Тепер він порівнює версію до оновлення з оновленою віртуальною DOM і визначає, що саме змінилося в DOM, наприклад, які компоненти було змінено. Цей процес порівняння поточного дерева Virtual DOM з попереднім називається «відмінюванням». Коли React дізнається, що саме змінилося, він оновить лише ці об'єкти на реальному DOM.

У React дані передаються в одному напрямку, від батьківського до дочірнього. Це допомагає компонентам бути простими та передбачуваними. Компоненти React – це прості функції, які отримують властивості, стан і повертають HTML-подібну розмітку. Коли дочірні компоненти отримують атрибути від своїх батьків, вони або застосовують модифікації (рендеринг), або передають їх іншому дочірньому компоненту, яка може їх використовувати.

React надає розробникам набір попередньо визначених функцій, які за наявності викликаються під час певних життєвих циклів компонента.

Подібно до React для веб-розробки, програми React Native написані з використанням суміші розмітки JavaScript і XML, відомої як JSX. Потім під капотом «міст» React Native викликає рідні API рендерингу в Objective-C (для iOS) або Java (для Android). Таким чином, програма відтворюватиметься за допомогою компонентів реального мобільного інтерфейсу користувача, а не веб-компонентів, і виглядатиме та працюватиме як будь-яка інша нативна мобільна програма.

Платформа RN складається з трьох частин (рис. 1):

1. Нативний код/модулі: більшість нативного коду в iOS написано на Objective C або Swift, тоді як на Android – на Java або Kotlin. Але для написання додатка React Native не знадобиться писати нативний код для iOS або Android.

2. Javascript VM: віртуальна машина JS, яка запускає весь код JavaScript. На симуляторах і пристроях iOS/Android React Native використовує JavaScriptCore, який є механізмом JavaScript, який працює в Safari. У випадку iOS React Native використовує JavaScriptCore, наданий платформою iOS. У випадку Android React Native об'єднує JavaScriptCore разом із програмою.

3. React Native bridge – це міст C++/Java, який відповідає за зв'язок між нативним потоком і потоком Javascript. Для передачі повідомлень використовується спеціальний протокол.

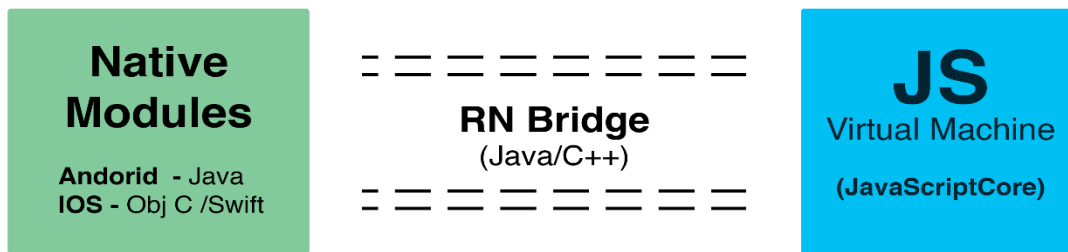


Рис. 1. Архітектура React Native

Для дослідження швидкодії мобільного додатку розробленого за допомогою фреймворку React Native, було вирішено спроектувати та розробити програмний пакет. Головним параметром оцінки мобільного клієнтського додатку є показник частоти кадрів. Враховуючи це, можна виділити наступні необхідні для програмного пакету показники:

- поточна частота кадрів;
- середня частота кадрів;
- поточний показник використаної оперативної пам'яті у мегабайтах;
- поточний показник використаної оперативної пам'яті у процентах від загальної кількості оперативної пам'яті пристрою;
- час роботи програми.

Програмний пакет повинен бути реалізований у вигляді окремого ізольованого модулю, що може бути легко використаний з іншою програмою для виміру її ефективності, або може бути опублікований як окремий npm-пакет для React Native та бути використаним у будь-якому проекті.

Також пакет повинен мати можливість конфігурації, а саме виводу поточних показників у консоль розробника, виводу поточних показників у графічному виді, і числовий параметр для налаштування періоду оновлення показників.

Головний функціонал пакету було реалізовано у вигляді спеціальної функції-хуку React. Оскільки сам пакет має бути оптимізований та мати мінімальний вплив на швидкодію додатку, вимагати мінімум ресурсів для роботи, то дані про поточну кількість кадрів, та інші зберігаються використовуючи механізм Ref в React. Цей механізм дозволяє зберігати проміжні дані в спеціальних об'єктах, які здатні зберігати свій поточний стан між рендерами компоненту, але жодним чином не впливати на життєвий цикл компоненту, тим самим не провокуючи зайвих рендерів інтерфейсу. Сам стан програмного пакету оновлюється лише відповідно вказаного розробником періоду.

Показник поточної кількості кадрів вимірюється за допомогою вбудованої функції requestAnimationFrame, що викликається під час зміни кадру на пристрої. Також для оптимізації показника поточної кількості кадрів, та уникнення ситуації відсутності даних у разі тестування додатку при великому навантаженні, було використано математичний метод інтерполяції. Враховуючи майбутній розвиток програмного пакету та додавання графіків для моніторингу показників, цей метод допоможе отримати нормалізовані дані для відображення.

Головним недоліком вбудованих інструментів розробника є те, що вони вимагають змінити оточення для запуску. При запуску програми в режимі дебагінгу,

стандартний двигун JavascriptCore замінюється на браузерний V8. В деяких випадках, це може стати джерелом неочікуваної поведінки або програмних збоїв.

Саме для відстежування стану роботи програми у реальному часі при її нативних умовах роботи і створено програмний пакет, який можна використовувати як і під час розробки, так і в продакшн збірці для збору статистики.

Для використання програмного пакету на реальному проєкті, відповідно до поставленої задачі, було спроектовано та розроблено тестовий мобільний додаток. Додаток реалізує найпоширеніші шаблони сучасних мобільних додатків:

- перегляд динамічно-завантаженого контенту;
- взаємодію зі стороннім API;
- взаємодію з користувачем через поля вводу;
- взаємодію з локальною базою даних пристрою.

Відповідно до цих вимог, був спроектований та розроблений додаток для пошуку перегляду інформації про фільми, який включає наступні елементи (рис.2):

- головна сторінка з механізмом Infinite Scroll, що забезпечує динамічне завантаження даних при прокрутці екрану пристрою;
- шар API для завантаження даних з серверу;
- сторінка пошуку фільмів з полем вводу;
- сторінка улюблених фільмів, що завантажує дані з локальної бази даних пристрою.

В результаті тестування додатку за допомогою розробленого пакету, середня частота кадрів становила 60, що є високим стандартом для сучасних мобільних пристроїв. В рідких випадках пікового навантаження спостерігались різкі зменшення показнику поточної кількості кадрів в секунду, однак вплив на показник середньої частоти кадрів мінімальний. Завантаження оперативної пам'яті також було помірним - від 260 до 320 мегабайтів.

Наукова новизна дослідження полягає у вдосконаленні методів визначення ефективності мобільного програмного забезпечення створеного з використанням програмного засобу React Native.

Висновки. В результаті було викладено методи та підходи щодо створення сучасних мобільних додатків, а також проаналізовано методи та підходи фреймворку React Native. Було розроблено програмний пакет для оцінки ефективності роботи мобільного додатку на React Native, а також тестовий додаток. Результати тестування додатку за допомогою розробленого програмного пакету є позитивними.

ПЕРЕЛІК ПОСИЛАНЬ

1. Shaun Lewis & Mike Dunn: Native Mobile Development. – 2019 – P.35-38
2. What is a Native Mobile App Development? – [Електронний ресурс] – Режим доступу: <https://mdevelopers.com/blog/what-is-a-native-mobile-app-development>
3. Carmine Zaccagnino: Programming Flutter: Native, Cross-Platform Apps the Easy Way. 1st Ed. – 2020 – P.11-12
4. Dan Bernes: Xamarin Mobile Application Development. – 2015 – P.15-16
5. Nader Dabit: React Native in Action. – 2019 – P. 4-5

К.Л. Сергєєва¹, С.Л. Нікулін¹, В.Ю. Каштан¹, О.В. Коробко¹, В.О. Шевченко¹
¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ДОСЛІДЖЕННЯ ОСОБЛИВОСТЕЙ ВИДІЛЕННЯ ГРАНИЦЬ ЯСКРАВІСТІ ПРИРОДНИХ ОБ'ЄКТІВ НА РІЗНОМАСШТАБНИХ КОСМОЗНІМКАХ

Анотація. Розглянуто проблему обрання оптимального масштабу космічних знімків для дослідження сейсмонебезпечних територій за допомогою аналізу границь яскравості (фототону) супутникових зображень. Показано можливість обрання масштабу, для якого зв'язок між просторовим розташуванням аномалій щільності границь яскравості та локалізацією епіцентрів землетрусів є найбільш суттєвим. Додатково досліджено вплив маскових операторів, що використовуються в алгоритмі виділення границь яскравості.

Ключові слова: *обробка зображень, масштаб геозображення, генералізація, космічні знімки, епіцентри землетрусів, границі яскравості, карти щільності границь яскравості.*

Вступ. Використання космічних знімків для вивчення сейсмоактивних територій, і, зокрема, при прогнозуванні сейсмічних подій, вже давно визнано перспективним напрямом геологічних досліджень завдяки своїй оперативності та відносній простоті.

На космічних знімках зміни сейсмічної активності проявляються у різних діапазонах спектру: у далекому інфрачервоному діапазоні змінюється інтенсивність теплових аномалій, у оптичному та радіодіапазонах зміни проявляються опосередковано, через гідрографічну та яружно-балкову мережі, рельєф, стан вегетації та ін. Другий випадок є більш складним, і тому вимагає додаткових досліджень. Саме їм присвячено дану роботу.

Одним із ефективних способів експрес-вивчення ландшафтних ознак є аналіз границь яскравості (фототону) космічних знімків. До границь відносять вузькі протяжні зони різкої зміни яскравості. Вони виділяються на знімках автоматично на основі низки масочних алгоритмів [1]. Границям зазвичай відповідають лінійні елементи гідрографічної та балкової мережі, лінійні форми рельєфу, межі різкої зміни типів рослинності. Зазначені елементи ландшафту, у свою чергу, перебувають у взаємозв'язку з низкою геологічних особливостей будови території, насамперед, мережами розломів і тріщин. Таким чином, аналіз границь яскравості особливо корисний щодо тектонічних деформацій земної кори. Останні тісно пов'язані з сейсмічними явищами.

Тож, методи аналізу знімків, що базуються на виділенні та вивченні границь яскравості, є досить перспективними. Оскільки конфігурація і кількість границь яскравості, що виділяються, чутлива до масштабу вихідного космознімку, встає задача пошуку масштабу, оптимального з точки зору виявлення стійкого зв'язку між границями яскравості та сейсмічними об'єктами.

Метою роботи є вивчення впливу масштабів космічних знімків на тісноту просторового зв'язку між концентрацією границь яскравості космічних знімків і

розташуванням епіцентрів землетрусів, та визначення принципової необхідності виконання досліджень з обрання оптимального масштабу знімків.

Постановка задачі. Дослідження базувались на припущенні, що підвищена сейсмічна активність веде до зростання диференційованості ландшафтів, та підвищенню кількості границь яскравості космознімків. Ставилось завдання експериментального дослідити вплив масштабів знімків декількох сейсмоактивних ділянок на концентрації границь яскравості біля епіцентрів землетрусів, що сталися в останні роки. Додатково вивчався вплив оператора (маски), за допомогою якого виділялися границі яскравості.

Вихідні дані та територія досліджень. Експериментальні дослідження проводилися на матеріалах трьох ділянок. Ділянка 1 розміром 156 x 67 км включає фрагмент Судетських гір; ділянка 2 (334 x 132 км), охоплює частину Карпат; ділянка 3, – сейсмонебезпечну територію центральної Туреччини, що займає площу понад 150 тис. км².

Використовувалися цифрові моделі рельєфу (Digital Elevation Model, DEM), побудовані за матеріалами зйомок з супутника ASTER (продукт ASTER GDEM 2, отриманий у жовтні 2011 року; джерело – сайт earthexplorer.usgs.gov), знімок з супутника TERRA MODIS з просторовим розрізненням 300 м, та дані про епіцентри землетрусів (джерело – сайт earth-quake.usgs.gov) (рис. 1).

Дослідження проводилися в середовищі геоінформаційної системи РАПІД, яка має розвинені блоки виділення й аналізу лінійних елементів та границь яскравості геозображень [2].

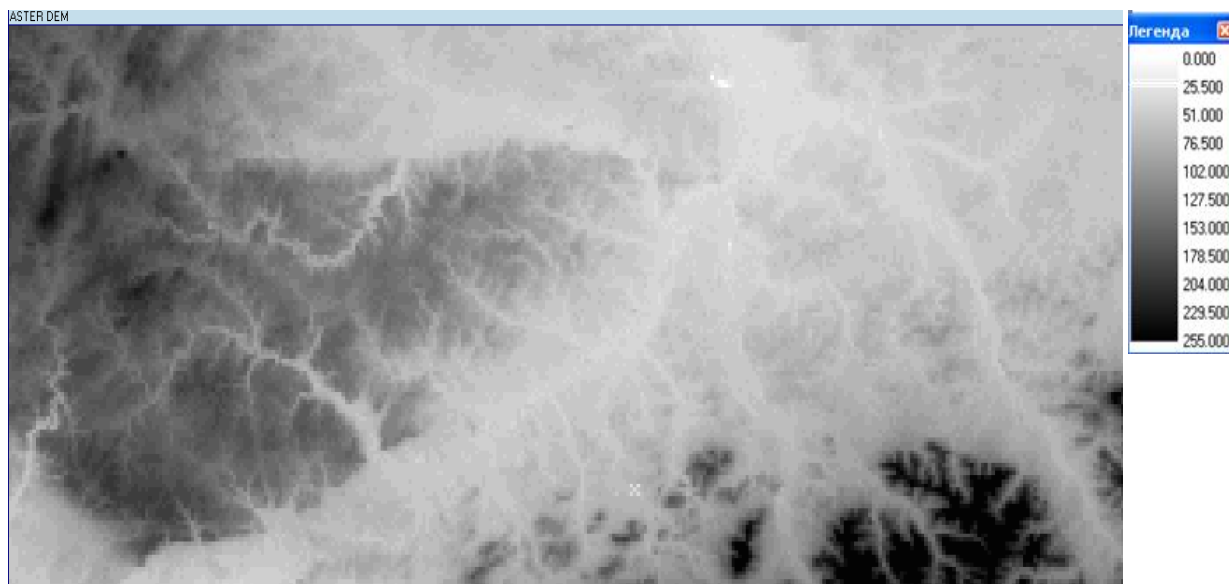


Рис. 1. Приклад вихідного геозображення сейсмонебезпечної території Судетських гір (знімок з супутника ASTER (продукт ASTER GDEM 2))

Методика досліджень.

Експерименти для кожної з трьох ділянок проводились за такою методикою:

1. Підготовка ансамбля різномасштабних знімків. Оскільки знайти різномасштабні знімки ділянки, які можна було б коректно порівнювати між собою, практично дуже складно, був обраний шлях розрідження вхідного знімку в 2, 3, 4 і т.д. разів.

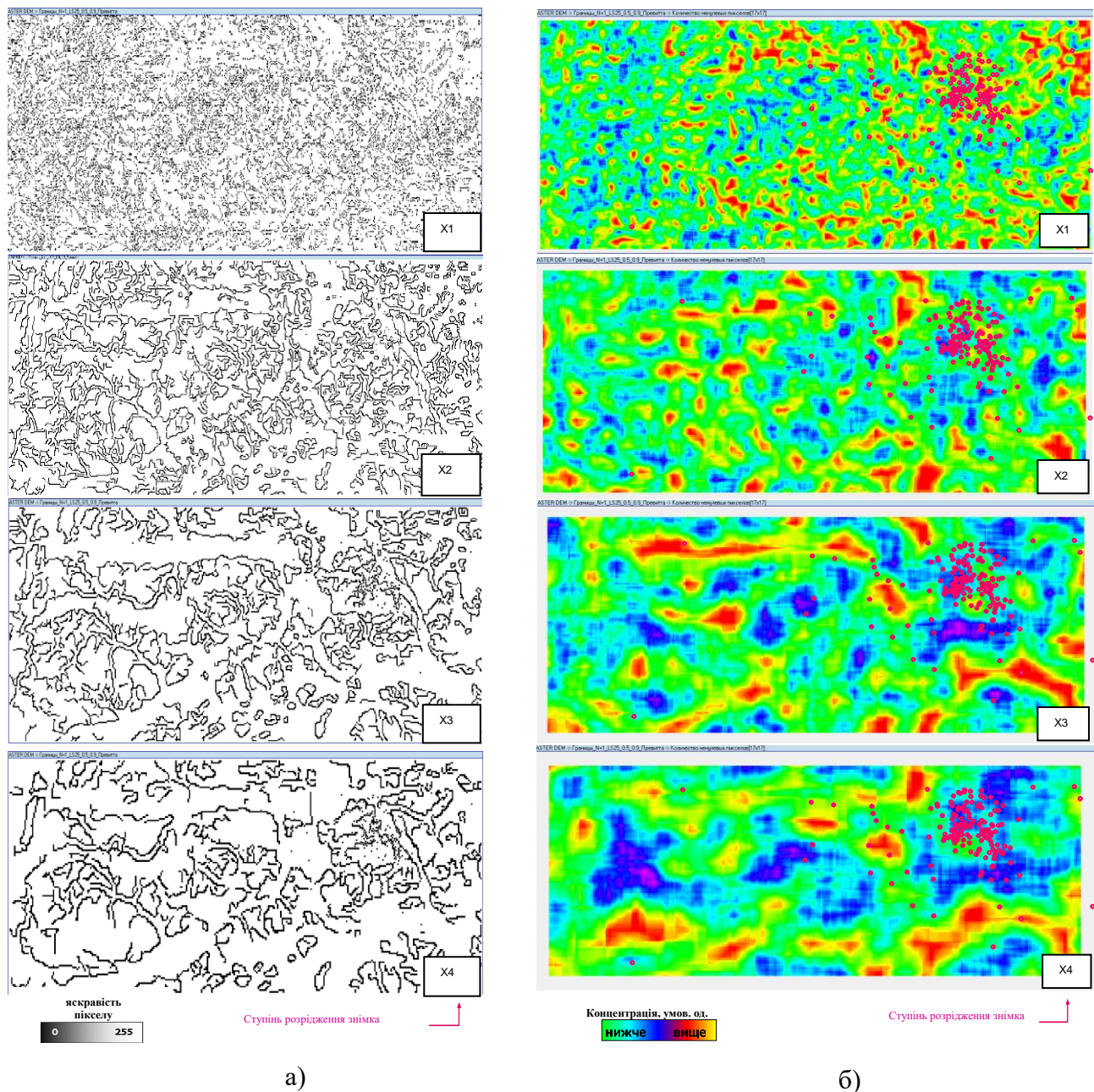


Рис. 2. Результати обробки геообразення: а) відображення карт границь яскравості території Судетських гір; вихідний знімок – продукт ASTER GDEM 2, маска Прюїтта (пороги 0,5-0,9); б) відображення карт концентрації границь яскравості (розмір ковзного вікна – 17x17 пікселів)

2. Методом Кенні виділялися границі яскравості. Для методу Кенні вказувався ряд параметрів: значення порогів для верхньої та нижньої меж, вид маскового оператора та інше (рис 2, а) [3].

3. Проводилася обробка карт границь яскравості. В результаті були отримані карти концентрації границь яскравості (рис. 2, б).

4. Для кожної отриманої карти концентрації границь яскравості проводився відбір 35% та 50% пікселів з найбільшими значеннями концентрації границь яскравості (рис. 3, а, 3, б). Пікселі, що залишились, формували зони підвищених значень (ЗПЗ).

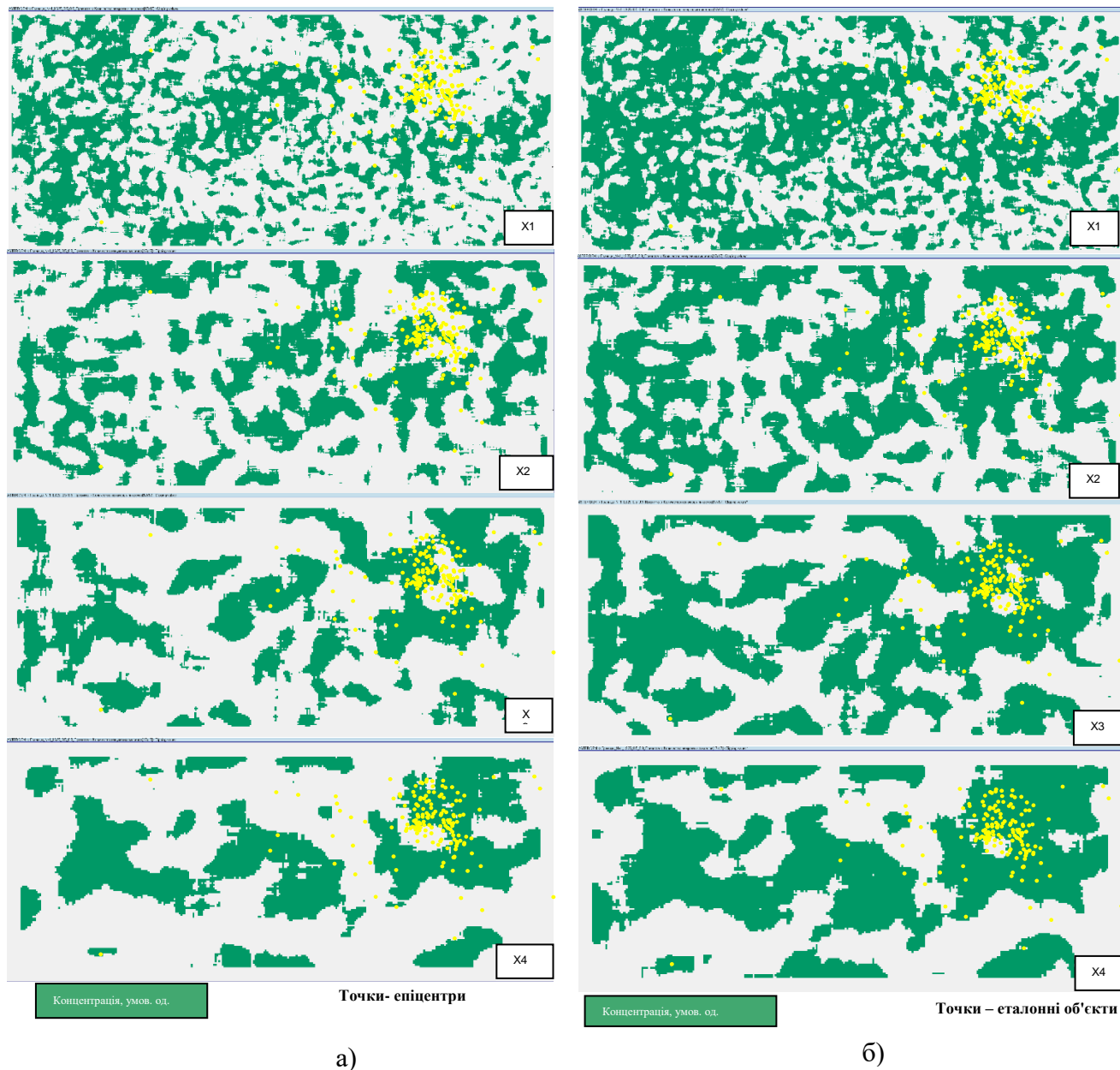


Рис. 3. Ділянка Судетських гір: а) відображено 35% пікселів з найбільшими значеннями концентрації границя яскравості; б) відображено 50% пікселів з найбільшими значеннями концентрації границь яскравості

5. Виконувався підрахунок епіцентрів землетрусів, що потрапляють у ЗПЗ досліджуваних геозображень. За результатами будувалися таблиці (наприклад, табл. 1).

Для аналізу карт концентрації границь яскравості створена зведена таблиця, в якій вказано, при якому масштабі в межі ЗПЗ потрапляє найбільша кількість епіцентрів землетрусів (табл. 2).

Таблиця 1

Кількість епіцентрів землетрусів, що потрапили в ЗПЗ з різним ступенем розрідження вхідного зображення (на території Судетських гір)

Ступінь розрідження вихідного знімка	Кількість точок епіцентрів землетрусів у зонах підвищеної концентрації границь яскравості			
	Всього на ділянці: 161 епіцентрів землетрусів			
	Маска: Прюїтта (пороги 0,5-0,9)		Маска: Кірша (пороги 0,5-0,9)	
	% площі з тах значеннями	% площі з тах значеннями	% площі з тах значеннями	% площі з тах значеннями
	35%	50%	35%	50%
-	33	66	38	69
2	72	114	103	134
3	122	138	113	139
4	133	141	135	141

Таблиця 2

Масштаби знімків, при яких найбільша кількість епіцентрів землетрусів потрапляє у межі ЗПЗ

Ділянка	Кращі результати (землетруси)				
	Ступінь розрідження вихідного знімка	Пікселі найбільшими значеннями КГЯ, %	Обраний масочний оператор	Кількість епіцентрів	%
Туреччина	2	50	Прюїтта	16	65%
Частина Карпат	1	50	Прюїтта	16	63%
Судети	4	50	Прюїтта, Кірша	141	88%

Обговорення результатів. Як витікає з отриманих результатів, конфігурація ЗПЗ границь яскравості значно змінюється при зміні масштабу вихідного знімка. Практично завжди існує деякий масштаб, при якому тіснота зв'язку між концентрацією границь яскравості та розташуванням епіцентрів

землетрусів максимальна. Такий масштаб може бути знайдений експериментальним шляхом при послідовному розрідженні вихідного знімка в 2, 3, 4 і більше разів або при використанні знімків однієї і тієї ж території, зроблених з різним просторовим розрізненням.

Висновки. В роботі наведена розроблена методика, що дозволяє проводити додаткові дослідження зв'язку між масштабом космознімку та концентрацією його границь яскравості. Вона має на меті суттєво підвищити якість отримуваних результатів, зокрема, при дослідженні сейсмоактивних ділянок і прогнозуванні наближення сейсмічних явищ.

Методика застосована на низці реальних сейсмічних зон, розташованих у різних частинах Земної кулі.

Додатково оцінено вплив маскового оператора, що використовується при виділенні границь яскравості. Навіть за отриманими попередніми даними можна стверджувати, що цей вплив досить суттєвий, і тому масковий оператор також має обиратися за наведеною методикою.

ПЕРЕЛІК ПОСИЛАНЬ

1. Canny. J.F. A computational approach to edge detection / J.F. Canny // IEEE Trans. on Pattern Analysis and Machine Intelligence. – 1986. –N 8(6). – P. 679-698.
2. Бусигін Б.С. Спеціалізована геоінформаційна система РАПД: структура, технологія, завдання /Б.С.Бусигін, С.Л.Нікулін // «Геоінформатика», Київ. 2016. – № 1 (57). – С. 22-37.
3. Нікулін С.Л., Коробко О.В. Вплив параметрів алгоритму Кенні на результати виділення границь яскравості космічних знімків// Зб. наук. тр. Нац. гірн. ун-ту/редкол.: Г.Г. Півняк [та ін.] – Д.: Національний гірничий університет, 2013. – №1. – 51-52 с.

УДК 004.415.3:681.6

Б.І. Мороз¹, Г.Г. Швачич¹, Д.М. Мороз¹, І.В. Смигунов¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

РОЗРОБКА ТА ДОСЛІДЖЕННЯ ЕФЕКТИВНОСТІ ВПРОВАДЖЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ УПРАВЛІННЯ ПЕРСОНАЛОМ З ВИКОРИСТАННЯМ HRM СИСТЕМИ НА БАЗІ АВТОМАТИЗОВАНИХ КЛЮЧОВИХ HR-ПРОЦЕСІВ

Анотація. Досліджено ситуацію на сучасному ринку систем керування персоналом та методи реалізації даної системи за допомогою актуальних на сьогоднішній день рішень.

Ключові слова: HRMS, Odoо, персонал, XML, автоматизована система, бізнес, база даних, фреймворк, ERP.

Вступ. З кожним днем технології стають все доступнішими та невід'ємною частиною нашого повсякденного життя, тому не дивно, що компанії, великі й малі, тепер намагаються автоматизувати якомога більше внутрішніх процесів, оскільки це забезпечує ефективність. сухого залишку

дозволяють виконувати більше завдань з меншими витратами, як матеріальними, так і моральними. Одним із таких процесів є робота менеджера з персоналу, який щодня взаємодіє з даними про співробітників, а також відповідає за набір нових співробітників на робоче місце, що, природно, робить цю посаду дуже важливою та трудомісткою.

Успіх і зростання будь-якої організації залежить від управління та ставлення до її працівників. Ось чому деякі організації експериментують із різними підходами, щоб покращити свої кадрові стратегії та досвід. Тут справді вступає в дію HR-програмне забезпечення. З програмним забезпеченням для управління персоналом управління співробітниками більше не є рутиною, яку можна автоматизувати, а чудовим досвідом, який підтримує мотивацію та задоволення співробітників. Кожен аспект вашої роботи з кадрами можна виконувати з одного централізованого місця за допомогою програмного забезпечення для кадрів. Воно виконує всю адміністративну роботу, щоб ви могли подбати про свій персонал.

Постановка задачі. Завдання – розробити автоматизовану систему управління персоналом та підбору нових працівників. Метою запиту щодо інформаційного наповнення програми було створення швидкодіючої та зручної системи з гнучкими функціями, зрозумілим і зручним інтерфейсом, а також можливістю подальшого розширення спектру доступних функцій шляхом встановлення додаткових модулів.

Необхідно розробити систему управління людськими ресурсами на основі структури бази даних з такими функціями:

- гнучкий відбір кандидатів і можливість створення нових співробітників на основі кандидатів, які успішно пройшли відбір;
- створення нових співробітників і зручне управління ними, наприклад, зміна відділу, надання певних дозволів і т.д.;
- дозвіл на створення бази контактів, за допомогою якої можна легко зв'язатися з людиною;
- зручний блок з календарем, який можна використовувати для планування співбесід та інших подій;
- можливість брати відпустки наданих типів або створювати нові типи.

Основний зміст роботи. Під час роботи над даною роботою були використані наступні методи та інструменти.

- методика оцінки комплексної ефективності управління персоналом підприємства;
- Odoo, Python, PostgreSQL, XML.

Обсяг світового ринку систем управління персоналом (Human Resource Management, HRM) зростатиме в середньому на 12,8% щорічно і сягне \$56,15 млрд до 2030 року.

За словами експертів, основними драйверами зростання ринку стануть аналітичні інструменти, хмарні та мобільні технології, а також рішення у сфері штучного інтелекту та інтернету речей. Зазначається, що HR-аналітика та її

інтеграція з HRM-системами з метою ухвалення обґрунтованих рішень є вкрай важливими для розвитку галузі.

У період пандемії коронавірусу особливою популярністю у багатьох галузях стали користуватися інструменти автоматизованої оптимізації роботи персоналу. Таке ПЗ, як правило, використовує алгоритми штучного інтелекту, дозволяє складати графіки роботи співробітників, оцінювати зміни попиту та зіставляти роботу персоналу з їхніми робочими процесами.

Найбільшими виробниками систем управління персоналом дослідники назвали такі компанії (перераховані в алфавітному порядку): Accenture, ADP, Cezanne HR, Ceridian HCM, IBM, Kronos, Mercer, Odoo, Oracle, PwC, SAP, Cegid, UKG, Workday [1].

На поточний час найбільшу долю ринку розробки систем для бізнесу займає 1С. Програмні продукти російської компанії «1С» з початку 2000-х історично є одними з найпопулярніших рішень для автоматизації управління та обліку на підприємствах на пострадянському просторі, зокрема в Україні. Навіть попри те, що на продукти цього розробника з 2017 року накладені санкції, впродовж кількох останніх років аж до лютого 2022-го в українських компаніях був стабільний попит на спеціалістів «1С».

Після початку повномасштабного вторгнення розробники програмного забезпечення для підприємств та представники бізнесів будь-якого рівня почали шукати альтернативи 1С. Однією з таких альтернатив є продукт бельгійської компанії Odoo S. A.

Odoo зарекомендувала себе щодо ефективного управління та реалізації як підприємницької та державної діяльності на усіх етапах розвитку. Використання та впровадження Odoo сприятиме розвитку культури введення бізнесу в Україні та забезпечить підвищення якості знань підприємців у сфері ведення бізнесу.

Для більш детального розуміння поставленої задачі нам необхідно ознайомитися з тим, як вимірюється ефективність керування персоналом.

Розрахунок комплексної ефективності керування персоналом підприємства визначається як відношення фактично досягнутих відповідних показників до базисних значень кінцевих результатів діяльності, що зважені ваговими коефіцієнтами значимості функцій керівництва щодо нормативного значення ефективності, що дорівнює 100 балам.

Методика оцінки комплексної ефективності управління персоналом складається з методів економічного аналізу, експертних оцінок, бальному методі і теорії класифікації [2].

Метод експертних оцінок та кореляційний аналіз визначаються послідовність економічних, соціальних і організаційних показників, які характеризують кінцеві результати діяльності підприємства.

Список даних показників обумовлений вивченням нормативнозаконодавчих актів, матеріалів підприємства, форм і інструкцій для складання статистичної та оперативної звітності.

Задаються критерії досягнення встановлених кінцевих результатів з щонайменшими витратами ресурсів та високою якістю продукції. Чисельні значення даних критеріїв визначаються з фінансових документів, форм статистичної і оперативної звітності і розраховуються у виді відсоткового відношення фактичного значення кінцевого результату до базисного (2.1):

$$X_i = \frac{P_{fi}}{P_{bi}} * 100\%, \quad (1)$$

де X_i – процентне відношення i -го часткового показника ефективності, %;

P_{fi} – фактичне значення i -го показника кінцевого результату за звітний період;

P_{bi} – базисне значення i -го показника кінцевого результату (план, норматив, факт попереднього періоду) за звітний період.

Показник X_i відображає ступінь досягнення фактичного кінцевого результату.

Для того, щоб критерії більш об'єктивно відображали результати керування персоналом та не перекривали один одного, вони попередньо піддаються корегуванню по формулі (2.2).

$$P_i = \int f(X_i), \quad (2)$$

де P_i – чисельне значення скорегованого i -го показника кінцевого результату, %; $f(X_i)$ – математична функція корегування i -го показника.

При цьому використовують 4 залежності:

- лінійна висхідна ($P = X$), коли заохочується кожен відсоток досягнення кінцевого результату, а при недовиконанні приймається його фактичне значення (має відношення до соціальних показників);

- лінійна низхідна ($P = 200 - X$), коли заохочується досягнення результату з найменшими витратами ресурсів, а за перевитрату ресурсів нараховується менша кількість балів. Ця функція застосовується для таких ресурсних показників, як витрати на 1 грн. товарообігу, фонд заробітної плати, плинність робочих кадрів, втрати робочого часу;

- піраміда, коли заохочується тільки 100-відсоткове досягнення кінцевого результату і не заохочується недовиконання або перевиконання. При цьому чисельне значення скорегованого показника до 100% визначається за формулою $P = X$, а при $X > 100\%$ – за формулою $P = 200 - X$;

- лінійна зворотна (штрафні санкції) передбачає нарахування негативних відсотків за формулою $P = -X$, коли чисельне значення такого показника приводить до негативних явищ у діяльності підприємства (наприклад, розкрадання матеріальних цінностей, виробничий травматизм,

порушення трудової дисципліни). Ці показники не плануються, а враховуються у виді штрафних санкцій.

За допомогою зазначених показників розраховується комплексний показник ефективності, у якому порівнюються різні економічні, соціальні і організаційні показники з урахуванням їх важливості.

Комплексний показник ефективності розраховується як сума часткових показників ефективності, що відображають кінцеві результати діяльності підприємства, використання ресурсів, соціальну діяльність і організаційну результативність праці співробітників підприємства.

Часткові показники визначаються за результатами виконання економічних, соціальних і організаційних показників шляхом множення відсотків їхнього виконання на вагові коефіцієнти значимості функцій керівництва (2.3):

$$\text{ЧПі} = \text{Пі} * \text{Ві}, \quad (3)$$

де ЧПі – значення і-го часткового показника ефективності управління персоналом, бали; Пі – виконання економічних, соціальних і організаційних показників, %; Ві – ваговий коефіцієнт і-го часткового показника, частка.

Ваговий коефіцієнт показує відносну важливість відповідного показника у загальній сукупності показників комплексної ефективності.

Він також вводиться для усунення розбіжності інтересів підприємства, трудового колективу і кожного працівника. Вагові коефіцієнти прямо пропорційно впливають на величину часткових показників ефективності роботи.

Вони визначаються методом експертних оцінок, шляхом ранжирування показників із присвоєнням їм питомих ваг у частках одиниці. При цьому для окремих показників кінцевих результатів діяльності підприємства доцільно зафіксувати певне значення частки в розмірі не менш 0,5, залишивши на показники результативності праці, організаційної і соціальної ефективності питому вагу 0,5. У цьому випадку буде дотримуватися пріоритет результатів економічного розвитку підприємства.

Комплексний показник ефективності управління персоналом підприємства (е у балах) визначається за формулою (2.4):

$$E_{уп} = \sum_{i=1}^n \frac{(\text{Пі} - \text{Ві})}{\sum_{i=1}^n \text{Ві} * 100\%}, \quad (4)$$

де n – кількість часткових показників ефективності.

Оцінка підсумкового значення комплексного показника ефективності управління

персоналом підприємства залежить від його чисельного значення:

- якщо воно менш 95 балів, то управління персоналом підприємства є незадовільним;
- якщо воно знаходиться у діапазоні 95-100 балів, то управління персоналом підприємства є задовільним (але не використані усі можливості);
- якщо воно знаходиться у діапазоні 100-105 балів і виконані усі часткові показники, то управління персоналом підприємства є добрим;
- якщо воно більш 105 балів, то загальна оцінка ефективності управління персоналом підприємства є відмінною.

Призначення та ціль створеної автоматизованої системи керування ключовими HR-процесами – спрощення взаємодії менеджерів та звичайних співробітників з повсякденними кадровими задачами, а також реалізація вбудованого відбору кандидатів, що дозволяє рекрутерам досягати максимальної ефективності.

У зв'язку з цим наша система повинна задовольнити наступним умовам:

- максимальна швидкість доступу до інформації;
- представлення інформації повинно бути максимально простим і чітким;
- швидкий доступ до всіх функціональних розділів;
- розділення функціоналу системи на ролі для того, щоб користувачі мали доступ лише до тієї інформації, яка відповідає їх посаді, таким чином мінімізується вірогідність порушення конфіденційності та непередбачуваних проблем в системі.

Наукова новизна полягає в тому, що було досліджено та удосконалено існуючі на ринку рішення по керуванню персоналом за допомогою більш актуальних технологій на поточний час.

Висновки. В результаті було спроектовано та розроблено автоматизовану систему керування персоналом, що задовільняє більшість базових потреб менеджера, підвищує ефективність та спрощує повсякденні процеси, а також дозволяє проводити гнучкий відбір нових кадрів.

ПЕРЕЛІК ПОСИЛАНЬ

1. Thomson Carswell. Hr Manager's Guide to Human Resources Management Systems, 2018, 42 с.
2. Романенков Ю., Зейнієв Т. Завдання контуру стратегічного управління ефективністю бізнес-процесів в організації. Системні дослідження та інформаційні технології №3, м. Київ, Україна, 2015, С.43-47

Л.І. Мещеряков¹, О.В. Зберовський¹, А.Л. Ширін¹, С.М. Денисюк¹
¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ПОТЕНЦІАЛ ВИКОРИСТАННЯ ОБ'ЄКТІВ ТИПУ LOFT У СЕРЕДОВИЩІ AUTODESK 3DS MAX

Анотація. Описано потенціал використання складних об'єктів типу Loft у середовищі Autodesk 3ds Max, наведено приклади створення та налаштування властивостей loft-об'єктів, роз'яснено визначення лофтіngu та освітлено варіанти його використання у створенні складних об'єктів.

Ключові слова: *об'єкти, сцени, loft-об'єкти, моделювання, Autodesk 3ds Max, модифікатори геометрії.*

Вступ: Loft-об'єкти (об'єкти, отримані в результаті лофтіngu) є найбільш складним та цікавим типом об'єктів, які можна створювати у середовищі Autodesk 3ds Max. Вони можуть бути створені шляхом застосування модифікатору Loft, використання якого відкриває величезні можливості для моделювання. Ці можливості ґрунтуються на широкому спектрі змін, що можуть бути застосовані до об'єктів типу Loft, а також на різноманітності вибору форм, що можуть бути надані елементу із застосуванням лише одного модифікатора. Потенціал таких об'єктів може бути реалізовано у різних сферах професійної діяльності, наприклад, такій як лікарська, дизайнерська, проектувальна та інженерна, в тому числі вузької спеціалізації, такої як 3D-візуалізатор або 3D-моделлер.

Мета роботи: проаналізувати можливі сфери використання Loft-об'єктів, дати зрозуміле визначення та призначення таких об'єктів, продемонструвати переваги лофтіngu, що надають цьому типу складних об'єктів потенціал для потужного використання у різних сферах.

Основний зміст роботи: Створення loft-об'єктів полягає у поєднанні будь-яких форм поперечного перерізу за допомогою одиночної форми шляху. Тобто створюється основна лінія (шлях) підтримки будь-якої кількості поперечних перерізів (форм). Професіонали художники та скульптори створюють головну лінію та додають лініями поперечні перерізи, потім утворюється поверхня за рахунок заповнення проміжків між поперечними перерізами [1].

У роботі з Loft-об'єктами користувач стикається з рядом таких визначень, як:

- Вершини – вони визначають форми поперечного перерізу та шляхи. Вони мають властивості Corner, Smooth та обидва типи Bezier.
- Сегменти – частина сплайну між двома вершинами.
- Сплайни – сукупність з'єднаних сегментів.
- Кроки – кількість поділів сегмента, що використовується для подання кривої. Кількість кроків визначає гладкість та щільність каркасу поверхні для лофтіngu.

- Шляхи – описують одну форму, що визначає основну лінію loft-об'єкта.

Є кілька вимог як до форм поперечного перерізу, так і для форми шляху. Форми шляху можуть містити тільки один сплайн. Autodesk 3ds Max відмовляється приймати будь-яку форму, що містить як шлях більше одного сплайну (наприклад, пончик). До форм поперечного перерізу застосовуються два обмеження. Всі форми на шляху повинні містити однакову кількість сплайнів. Усі форми на шляху повинні мати однаковий порядок вкладення.

Багато loft-об'єктів створюються за допомогою однієї форми на шляху, але можна створити більш цікаві та складні об'єкти, помістивши на шлях безліч форм.

Для побудови Loft-об'єкту треба спочатку створити базові форми та виділити сплайн-шлях. Після цього обрати пункт меню Create/Compound Objects/Loft. Активувавши інструмент, потрібно увімкнути режим вибору форми Get Shape, після чого можна обрати сплайн-форму об'єкту.

Розглянемо можливості лофтингу. До створених об'єктів можна додавати форми, шляхом встановлення рівня поточного шляху у світку Path Parameters (1, рис. 1) для позначення того, де буде додаватися форма і натискання кнопки Get Shape для вибору форми. Також перед вибором можна обрати метод клонування Move, Copy або Instance (2, рис. 1). Move – форма-перетин, буде поміщена в loft-об'єкт, що створюється, і перестане існувати як незалежний об'єкт; Copy - у складі створюваного loft-об'єкта буде використана незалежна копія вихідної форми-перетину; Instance – буде застосовано зразок форми-перетину, який встановлений за умовчанням і використовується найчастіше, оскільки будь-яка наступна допустима зміна форми відобразиться на loft-об'єкті [2]. Переходи між формами можна контролювати у відсотковому відношенні або за відстанню, а навігація по рівням форм здійснюється за допомогою кнопок (3, рис. 1).

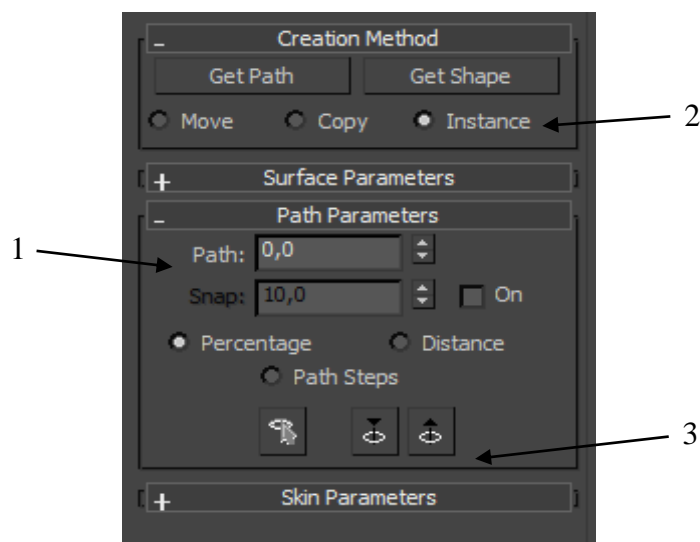


Рис. 1. Параметри побудови об'єкту типу Loft

Розглянемо на прикладі процес створення об'єкту типу Loft «Новорічна прикраса» (рис. 2). На сцену було додано три сплайни: коло, квадрат та лінія, що була взята за шлях. До лінії було застосовано модифікатор Loft та взято за форму коло. Шлях було поділено на дві частини за допомогою додавання квадратної форми на відмітці 75% Path. Після цього об'єкт було деформовано за допомогою деформатора Scale.

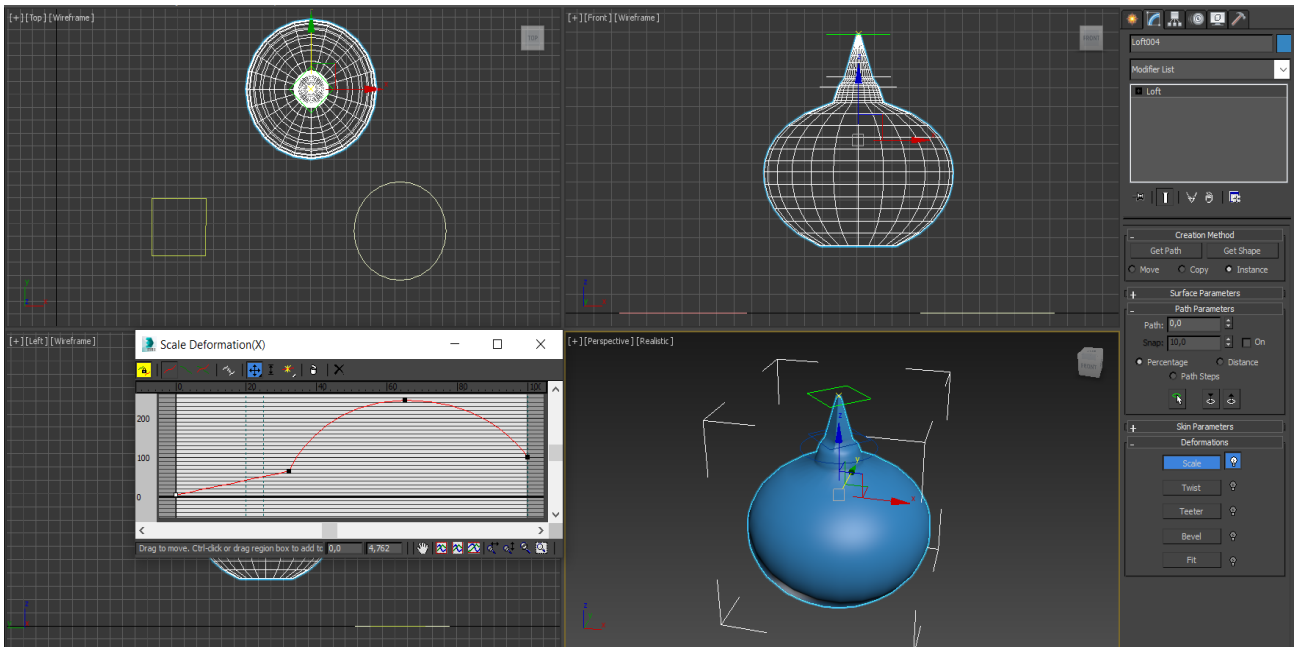


Рис. 2. Готовий об'єкт «Новорічна прикраса»

Для деформації loft-об'єкту треба виділити його та обрати вкладку Modify – тоді у списку параметрів з'явиться свиток Deformations, де будуть доступні деформатори Scale, Twist, Teeter, Bevel та Fit. При їх використанні з'являється вікно налаштувань, у якому можна додати точки та застосовувати до них Bezier-Smooth (рис. 3).

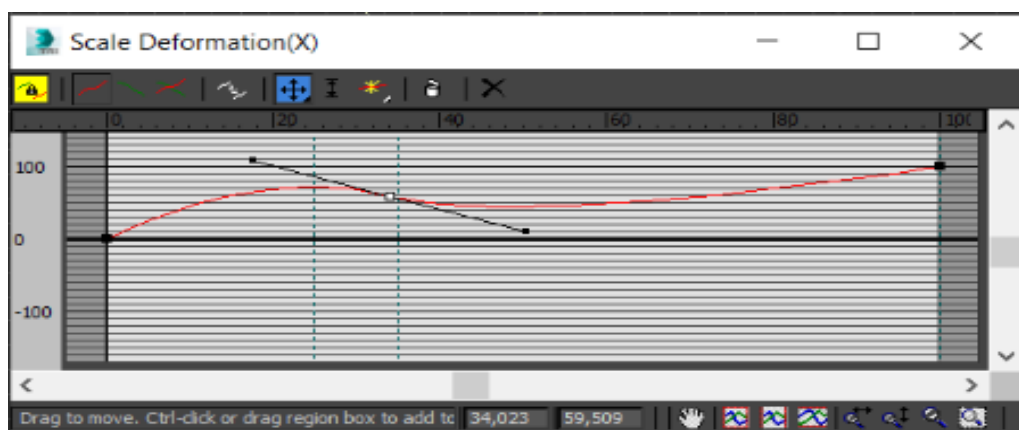


Рис. 3. Вікно налаштувань деформатора Scale

До об'єкту можна застосовувати одночасно декілька деформацій. Так, при побудові наступного об'єкту (рис. 4), було використано деформацію Scale для надання необхідної форми, та Twist для закручування форми.

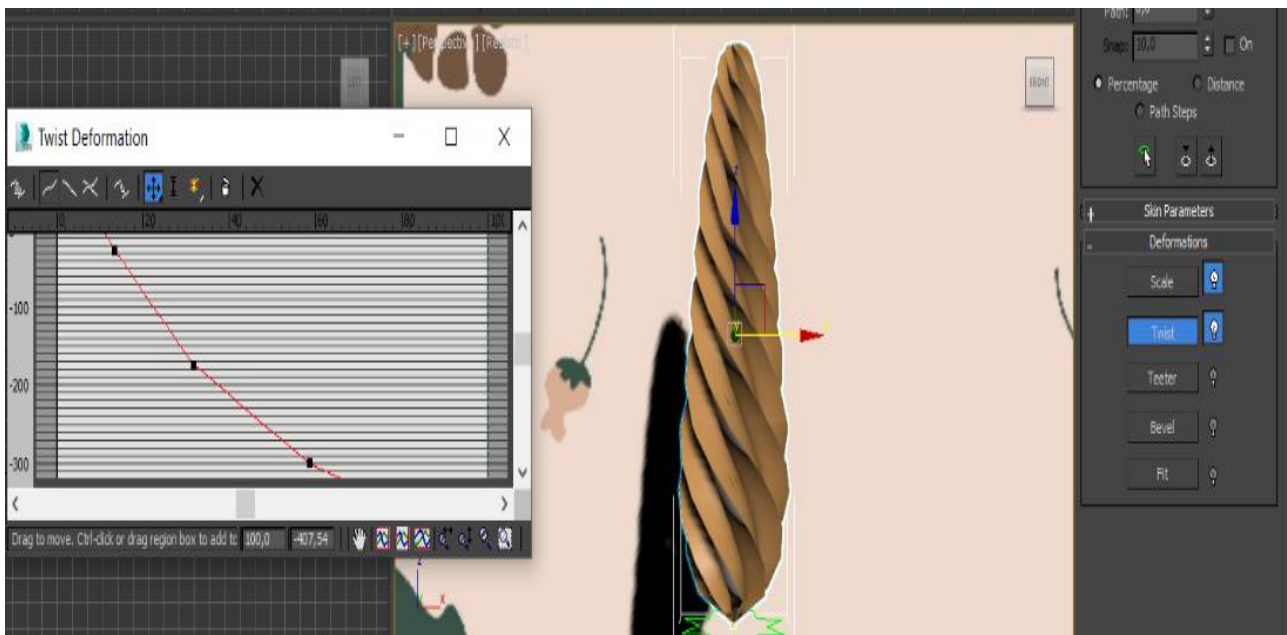


Рис. 4. Готовий об'єкт «Новорічна прикраса шишка»

Можна зробити висновок, що шляхом простих модифікацій користувач може створювати loft-об'єкти із складною структурою. Використовуючи примітивні спотворення було створено тримірні моделі простих ялинкових прикрас (рис. 5). Моделювання сплайнами цікаве тим, що воно дуже автоматизоване. При мінімальній кількості дій можна отримати повноцінний об'єкт із коректною сіткою. При цьому сам інструмент досить гнучкий і добре працює з будь-якими формами [3].

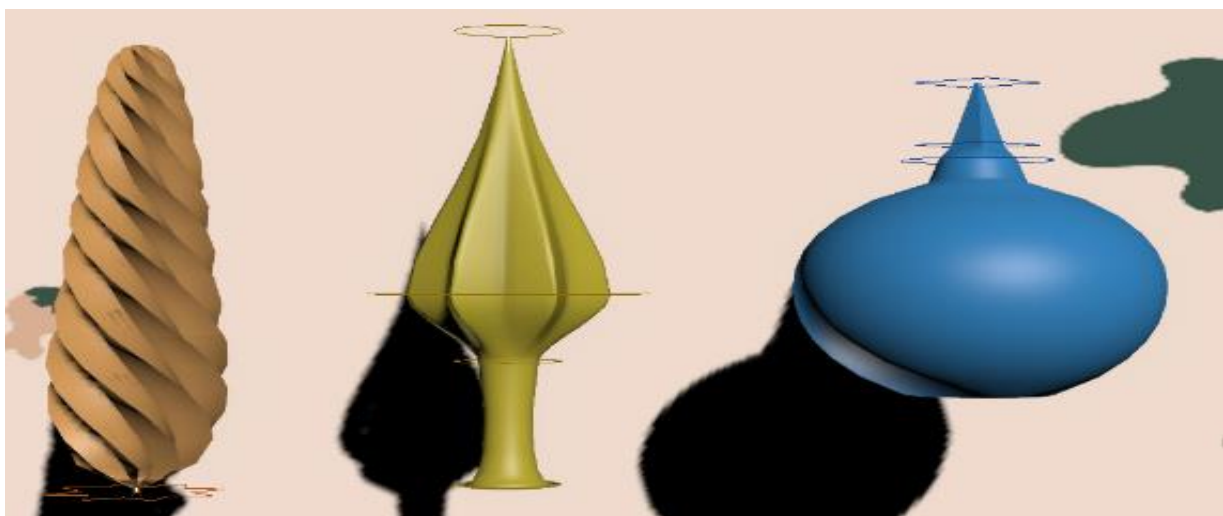


Рис. 5. Моделі ялинкових прикрас

Висновки. В ході роботи було описано на прикладах потенціал використання об'єктів типу Loft у середовищі Autodesk 3ds Max, сформульовано основні методи створення таких об'єктів, роз'яснено специфічні визначення для loft-об'єктів та освітлено варіанти використання лофтіну у створенні складних об'єктів.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Глібко О. А., Максимова М. О., Гречка І. П. Комп'ютерна графіка. Створення моделей та сцен у тривимірному середовищі : навчальний посібник / Харків: НТУ «ХП», 2018 – 130 с.
2. Бойко А. П., Дворник О. В. Комп'ютерне проектування в середовищі 3Ds Max : навчальний посібник / Миколаїв: Видавництво ЧНУ ім. Петра Могили, 2020 – 140 с.
3. Лотошинська Н., Ізонін І. Технології 3D-моделювання в програмному середовищі 3ds Max з дисципліни "3D-Графіка" / Львів : Львів. політехніка, 2020 – 216 с.

УДК 004.415.3:681.6

С.Д. Приходченко¹, М.О. Нугуманов¹, О.В. Іванченко¹, О.В. Голінько¹
¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ОСОБЛИВОСТІ ЗАСТОСУВАННЯ АЛГОРИТМІВ ТИПУ «РОЗДІЛЯЙ І ВОЛОДІЙ» В ГАЛУЗІ DATA SCIENCE

Анотація. В роботі розглянуті основи підвалини розрахунків часу роботи алгоритмів взагалі. Особливий акцент в роботі зроблено на алгоритми типу «Розділяй та володарюй»; їх підтипи, загальну побудову, та час, що витрачається цими типами алгоритмів. Приведені структурні схеми реалізації та приклади коду для цих типів алгоритмів. Обчислені в загальному вигляді затрати часу на функціонування цих алгоритмів.

Ключові слова: алгоритми, розрахунки часу, об'єкти, структурні схеми реалізації, матаналіз, асимптотичний час, концепцією асимптотичного позначення.

Вступ. На даний момент технологічного розвитку людина винайшла велику кількість пристроїв та чисельних методів для автоматизації рутинних процесів та вирішення завдань із різних сфер людської діяльності. Закон Мура говорить: "Кількість транзисторів, що розміщуються на кристалі інтегральної схеми, подвоюється кожні 24 місяці". Дійсно сучасні комп'ютери мають потужність на порядок вище, ніж комп'ютери 20-річної давності, але насправді швидкість роботи комп'ютера залежить не тільки від щільності і кількості транзисторів, але і від якості алгоритмів, що використовуються для виконання завдань, що особливо актуально при роботі з BIGDATA. На даний момент досить добре вивчено методи оцінки часу роботи алгоритмів. У computer science використовується поняття - асимптотичний час [1]. Різні алгоритми мають різну складність (залежність часу роботи від розміру вхідних даних може задаватися математичною формулою): лінійну, квадратичну... Не складно

здогадатися у якому разі швидкість буде лінійною (у разі, якщо алгоритм містить цикл), квадратична складність з'являється у разі, якщо у програмі є вкладений цикл (у разі програма виконуватиме число операцій кратне $n \cdot n$, де n – розмір вхідних даних). Приклади вище досить очевидні, але, крім наведених вище варіантів, існують алгоритми з логарифмічною складністю – $O(\log(n))$. Про них ми й поговоримо в основній частині статті.

Основний зміст роботи. Перед тим як заглибитися в алгоритми типу розділяй і владарюй слід познайомитися з позначеннями і розібратися з їх математичним змістом. Складність алгоритмів називають асимптотичним часом і позначають як O (наприклад: складність алгоритму з лінійним часом роботи позначається, як $O(n)$). Тепер слід розібратися зі словом асимптотична. Для людини, яка не вивчала матаналіз, це може здатися не так просто, асимптота – це пряма до якої при наближенні до нескінченності устремляється функція. У нашому випадку використовувати асимптотичний час зручно для оцінки складності алгоритмів так як даний підхід дає можливість вийти на певний рівень абстракції, позбавитися членів нижчого порядку і класифікувати алгоритми їх асимптотичної складності. Наприклад: повний час роботи алгоритму може дорівнювати $6n^2 + 6n + 6$ – у такому разі ми говоритимемо, що асимптотичний час даного алгоритму дорівнюватиме $O(n^2)$ або ж $T(n) = O(f(n)) = O(n^2)$, де $T(n)$ - повний час роботи алгоритму (у нашому прикладі - $6n^2 + 6n + 6$). Використовуючи таке позначення, можна говорити, що є такі позитивні константи c і n_0 при яких $T(n)$ буде обмежено згори $c \cdot f(n)$ при $n > n_0$ (рис. 1.), тобто $T(n) \leq c \cdot f(n)$, при всіх $n > n_0$.

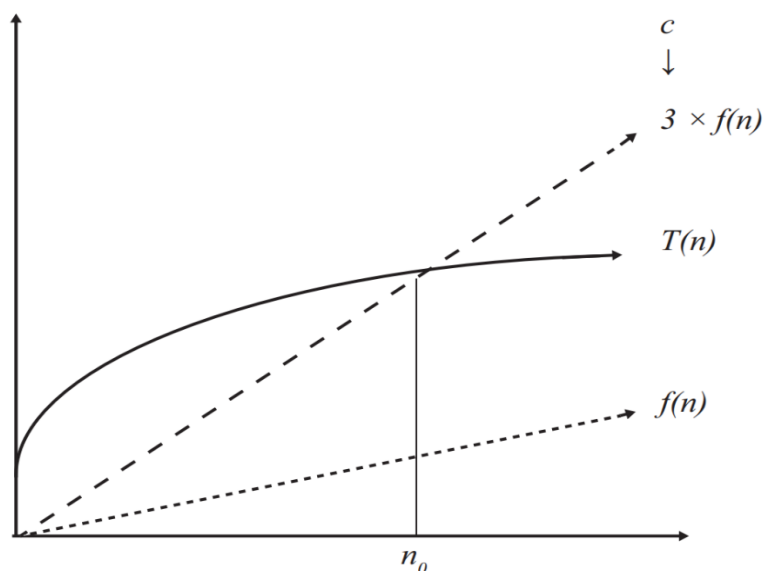


Рис. 1. Відображення $T(n)$ відносно $f(n)$ та $cf(n)$

Тепер, коли розібралися трохи з концепцією асимптотичного позначення часу роботи алгоритмів, давайте згадаємо про ті загадкові алгоритми, чий час роботи дорівнює $O(\log(n))$ [2]. $O(\log(n))$ швидше ніж $O(n)$, а $O(n \cdot \log(n))$

швидше ніж $O(n^2)$, оскільки функція $f(n) = \log_a(n)$ зростає повільніше аніж функція $g(n) = kn + b$ (рис. 2.), де $a \in N$, $k \in N$, $b \in R$. Це можна зрозуміти, досліджуючи похідні від цих функцій: $f'(n) = \frac{1}{n \cdot \ln(a)}$, в то час як $g'(n) = k$. В граничному випадку можемо побачити, що при будь якому значенні a и $n \rightarrow \infty$ функція $f'(n) = \frac{1}{n \cdot \ln(a)}$ наближається до нуля $\lim_{n \rightarrow \infty} \frac{1}{n \cdot \ln(a)} = 0$. Отже, враховуючи природу похідної (похідна є швидкістю зростання функції) можливо сказати, що зі збільшенням n (вхідних даних) функція часу роботи алгоритму складності $O(n)$ зростає з постійною швидкістю, у той час як для $O(\log(n))$ – швидкість зростання зменшується і прагне нуля при $n \rightarrow \infty$. Примітно, що в записі $O(\log(n))$ навіть не вказується основа логарифму, оскільки особливої ролі воно не відіграє.

Такий час роботи можна досягти при використанні алгоритмів типу «розділяй і володарюй» (хоча не всі алгоритми даного типу дадуть такий результат, але про це пізніше). Справа в тому, що дані алгоритми використовують зовсім інший підхід до вирішення задачі. Основна концепція полягає в рекурсивному розподілі даних на частини (відбиває сенс слова «розділяй» у назві методу) і розв'язанні задачі для меншого набору даних (відображає сенс слова «володарюй» у назві методу). «Поділяй і володарюй» – концепція, що народилася досить давно і відображає вона принцип державного управління, в якому різнорідна держава функціонує шляхом поділу сил на невеликі частини та зштовхування їх між собою. Алгоритми, що вимагають розділити дані для роботи з невеликими їх наборами, отримали ту ж назву.

Розглянемо принцип роботи алгоритмів «розділяй та володарюй» на прикладі алгоритму сортування злиттям MergeSort, але для початку оцінимо складність алгоритму сортування вибором SelectionSort [3].

Псевдокод SelectionSort:

Вхід: довільний n -розмірний масив A

Вихід: упорядкований n -розмірний масив

$minIndex := 0$

for $i := 1$ **to** $n - 1$ **do**

for $j := i + 1$ **to** n **do**

if $A[j] < A[minIndex]$ **then**

$minIndex := j$

if $minIndex \neq i$ **then**

$A[i], A[minIndex] := A[minIndex], A[i]$

Наведений вище алгоритм має складність $O(n^2)$. Чи можна досягти кращого?

Розглянемо алгоритм MergeSort. В даному випадку завдання – побудувати алгоритм типу «розділяй та володарюй» для сортування масиву. Спочатку рекурсивне розділимо масив (рис. 2.).

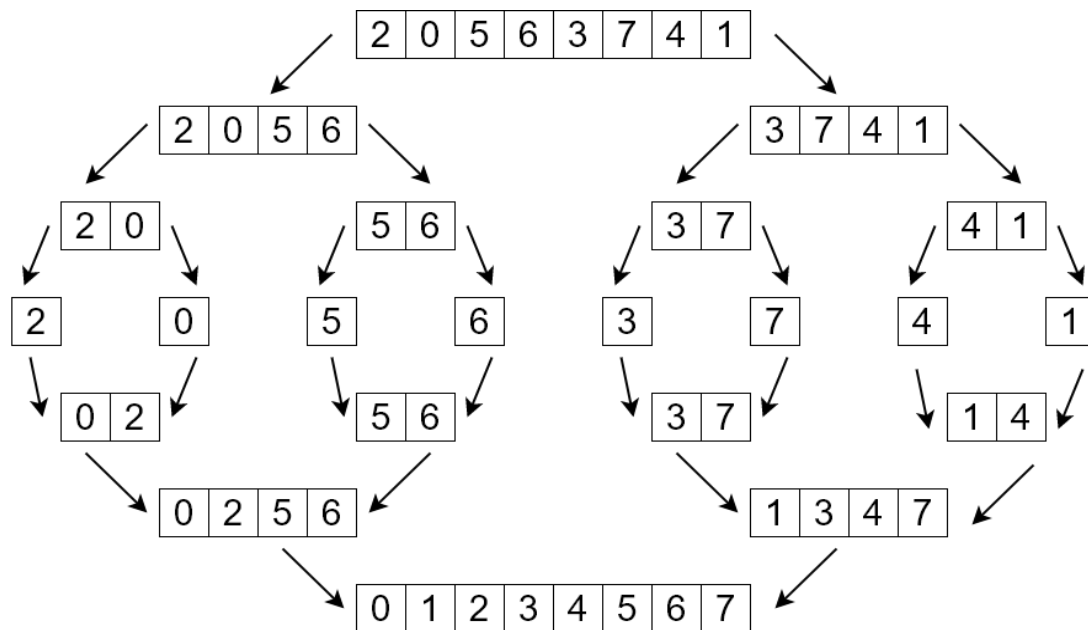


Рис. 2. Рекурсивне розділення масиву

Далі необхідно для кожних 2-х рекурсивних дзвінків виконати злиття Merge. Давайте подивимося на псевдокод функції Merge [4].

Псевдокод Merge:

Вхід: відсортовані масиви C і D (довжиною n/2 кожен)

Вихід: відсортований масив B

$i := 1$

$j := 1$

for $k := 1$ **to** n **do**

if $C[i] < D[j]$ **then**

$B[k] := C[i]$

$i := i + 1$

else

$B[k] := D[j]$

$j := j + 1$

Як видно алгоритм Merge лінійний – отже для кожного рекурсивного виклику (рис. 3) буде виконано лінійну кількість операцій, крім базового (вважатимемо, що кількість операцій на базовому рівні не залежить від розміру даних).

Псевдокод MergeSort:

Вхід: довільний n -розмірний масив A

Вихід: відсортований масив A

// базові випадки ігноровані

$C :=$ рекурсивно відсортована права частина

$D :=$ рекурсивно відсортована ліва частина

Повернути $Merge(C, D)$

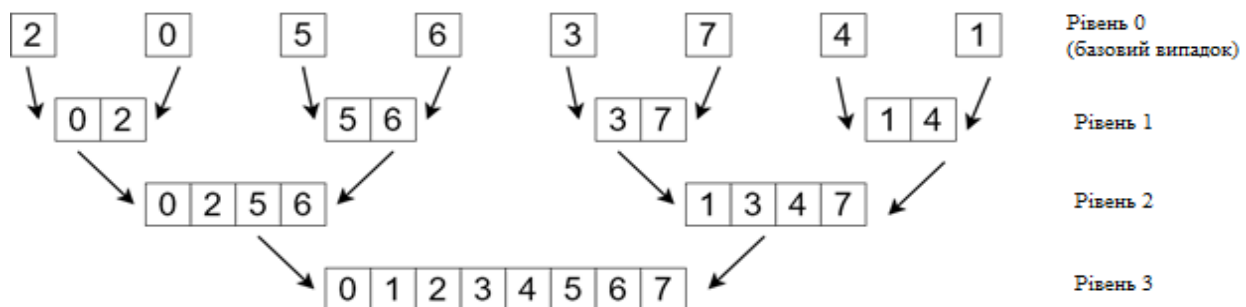


Рис. 3. Реалізація алгоритму MergeSort на Python представлена на рис. 4

Для 1 рівня: $\frac{n}{4}$ операцій для 4 рекурсивних викликів, отже разом $4 \cdot \frac{n}{4} = n$ операцій на 1 рівень.

Для 2 рівня: $\frac{n}{2}$ операцій для 2 рекурсивних викликів, отже разом $2 \cdot \frac{n}{2} = n$ операцій на 2 рівень.

Для 3 рівня також n операцій.

Виходить, що у кожному рівні виконується n операцій, отже всього операцій n помножене на кількість рівнів. Питання, як підрахувати кількість рівнів? Зауважимо, що кожного разу дані діляться на 2 і кількість рівнів дорівнює можливій кількості поділів до того моменту, поки не отримано одиницю. Вищеописана операція по суті є операцією зворотного зведення в ступінь або логарифмом на підставі 2. Виходить, що кількість рівнів дорівнює $\log_2 n$. Отримуємо, що складність алгоритму MergeSort – $O(n \cdot \log n)$, що набагато швидше ніж $O(n^2)$ для алгоритму SelectionSort.

```
def mergeSort(a, key = lambda x: x):  
    length = len(a)  
    if length > 1:  
        middle_index = length // 2  
        first_half = a[:middle_index]  
        second_half = a[middle_index:]  
        first_half = mergeSort(first_half, key)  
        second_half = mergeSort(second_half, key)  
        i = 0  
        j = 0
```

```

output_list = []
for k in range(length):
    if i >= len(first_half):
        output_list.append(second_half[j])
        j+=1
    elif j >= len(second_half):
        output_list.append(first_half[i])
        i+=1
    elif key(first_half[i]) > key(second_half[j]):
        output_list.append(second_half[j])
        j+=1
    else:
        output_list.append(first_half[i])
        i+=1
return output_list
else:
return a

```

Рис. 4. Реалізація алгоритму MergeSort на Python

Алгоритмів типу поділяй і володар існує багато і не кожен з них має логарифмічну складність також можна отримати субступеневу складність або ступеневу в гіршому випадку. Для того щоб зрозуміти це складемо формулу за допомогою якої можна розв'язати задачу про знаходження часу роботи алгоритму типу «розділяй і владарюй». Скажімо, що на кожному рекурсивному виклику виконується $f(n)$ операцій, тоді робота на рівні буде визначатися як сума операцій для кожного з рекурсивних викликів. Кількість рекурсивних викликів на рівні можна обчислити, поділивши кількість викликів на 0-му рівні a^i , де i – номер рівня, a – кількість рекурсивних викликів на одну гілку дерева (рис. 3.) (У прикладі MergeSort рекурсивних викликів – 2). Кількість викликів на 0-му рівні можна розрахувати, як $a^{\log_b n}$, де b – коефіцієнт поділу вхідних даних; a – кількість рекурсивних викликів; n – розмір вхідних даних. Отже, кількість операцій на рівні дорівнюватиме сумі операцій у всіх рекурсивних викликах на рівні (1).

$$\sum_{j=1}^{n^{\log_b a}} / a^i f(b^i) \quad (1)$$

де, i – номер рівня, j – номер рекурсивного виклика на рівні, b – коефіцієнт ділення вхідних даних, a – кількість рекурсивних викликів, $f(n_i)$ – функція, що відображає кількість операцій, для одного рекурсивного виклику ($n_i = b^i$).

Кількість операцій на всіх рівнях алгоритму визначається як сума операцій на кожному з рівнів з урахуванням того, що рівнів $\log_b n$ можемо скласти необхідну формулу (2).

$$T(n) = \sum_{i=0}^{\log_b n} \sum_{j=1}^{n^{\log_b a} / a^i} f(b^i) = \sum_{i=0}^{\log_b n} n^{\log_b a} \cdot \frac{f(b^i)}{a^i} = n^{\log_b a} \sum_{i=0}^{\log_b n} \frac{f(b^i)}{a^i} \quad (2)$$

де, i – номер рівня, j – номер рекурсивного виклику на рівні, b – коефіцієнт ділення вхідних даних, a – кількість рекурсивних викликів, $f(n_i)$ – функція, що відображає кількість операцій, для одного рекурсивного виклику ($n_i = b^i$).

За наведеною вище формулою (формула 2) можна розрахувати складність будь-якого алгоритму типу «розділяй і володарюй». Виходячи з отриманої формули можна зробити дуже цікаве узагальнення (3).

$$T(n) = \begin{cases} O(n^d * \log(n)), & \text{если } a = b^d \\ O(n^d), & \text{если } a < b^d \\ O(n^{\log_b a}), & \text{если } a > b^d \end{cases} \quad (3)$$

де, b – коефіцієнт поділу вхідних даних, a – кількість рекурсивних викликів, d – ступінь n в $f(n) = n^d$, де $f(n)$ – функція, що відображає кількість операцій для одного рекурсивного виклику.

Розглянемо ефективність формул з прикладу алгоритму Карацуби. Для алгоритму множення Карацуби: $a = 3$, $b = 2$, $f(n) = C$ ($C \in N$), $d = 0$. Пошук часу роботи для алгоритму Карацуби за допомогою формули 2 (реалізацію алгоритму Карацуби на python див. рис. 5):

$$T(n) = n^{\log_2 3} \sum_{i=0}^{\log_2 n} \frac{C}{3^i} = C n^{\log_2 3} \left(\frac{3 - \frac{1}{n^{\log_2 3}}}{2} \right) = \\ = C(1.5n^{\log_2 3} - 0.5) = O(n^{\log_2 3}). \quad (4)$$

```
def N(x):
    x //= 10
    n = 1
    while x > 0:
        x //= 10
        n += 1
    return n
def karatsuba(x, y):
    l = 0
    if N(x) == 1:
        return x * y
```

```

if N(x)%2 !=0:
    x *=10
    l+=1
if N(y)%2 !=0:
    y *=10
    l+=1
while N(y) < N(x) and x!=0 and y!=0:
    y *= 10
    l+=1
while N(x) < N(y) and x!=0 and y!=0:
    x *=10
    l+=1
n = N(x)
a , b= x // (10**(n//2)), x % (10**(n//2))
c , d= y // (10**(n//2)), y % (10**(n//2))
ac = karatsuba(a,c)
bd = karatsuba(b,d)
pq = karatsuba(a+b,c+d)
adbc = pq - ac -bd
return (ac * (10**n) + adbc * (10 ** (n//2)) + bd)// 10 **l

```

Рис. 5. Реалізація алгоритму Карацуби на Python

Те саме можливо отримати, скориставшись формулою 3:

$$a = 3, b = 2, d = 0 \Rightarrow a > b^d \Rightarrow T(n) = O(n^{\log_2 3}).$$

Висновок: алгоритми типу «розділяй і володарюй» цікаві до вивчення, знання алгоритмів даного типу та вміння визначити їхню складність – необхідна навичка для кожного програміста, особливо для фахівця в галузі Data Science.

ПЕРЕЛІК ПОСИЛАНЬ

1. Noam Nisan, Tim Roughgarden, Eva Tardos, and Vijay V. Vazirani. Algorithmic Game Theory. Cambridge University Press, New York, NY, USA, 2007. ISBN 0521872820.
2. Беляев О.Д., Ливаковский В.К. «Эмпирические оценки сложности некоторых алгоритмов в Python» 2022.
3. Introduction to algorithms / Thomas H. Cormen . . . [et al.].—3rd ed. Includes bibliographical references and index. ISBN 978-0-262-03384-8 (hardcover : alk. paper)—ISBN 978-0-262-53305-8 (pbk. : alk. paper)
4. Bhargava A. Grokaem algoritmy. Illjustrirovannoe posobie dlja programmistov i ljubopytstvujushhikh. [Grokking Algorithms: An Illustrated Guide for Programmers and Other Curious People] / A.Bhargava - SPb.: Piter, 2017. - p. 288

ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ ДЛЯ МОДЕЛЮВАННЯ ОПТИМАЛЬНОГО РАЦІОНУ ХАРЧУВАННЯ

Анотація. Результатом дослідження є створений програмний додаток для оптимізації раціону харчування хворих на цукровий діабет другого типу шляхом аналізу їх персональних даних та на основі обраного продуктового кошику.

Ключові слова: раціон харчування, хлібна одиниця, оптимізація, програмний додаток, програмні засоби C#.

Вступ. Пошук та аналіз наявних програмних засобів з оптимізації раціону харчування показав, що такі або не існують на сучасному ринку, або не покривають весь необхідний користувачеві функціонал. Це робить питання розробки програмного забезпечення моделювання оптимального раціону харчування досить актуальним. Розробка такого програмного забезпечення зумовлюється необхідністю пересічної людини, особливо, хворої на цукровий діабет другого типу, постійно слідкувати за своїм раціоном харчування задля недопущення перевищення норм споживання вуглеводів. Метою дослідження є створення програмного додатку мовою C# для оптимізації харчування на основі споживання хлібних одиниць.

Основний зміст роботи. Цукровий діабет – це захворювання, яке зумовлюються підвищенням рівня глюкози в крові через частковий або повний брак гормону інсуліну. Цукровий діабет може спричиняти низку інших захворювань, ураження внутрішніх органів, судин, нервової системи тощо.

Зважаючи на причини виникнення захворювання для хворих на цукровий діабет вкрай важливо мати збалансований раціон харчування, зокрема, не перевищувати допустимі норми щоденного вживання вуглеводів, адже вони найсильніше впливають на підвищення рівня цукру в крові. Розрахувати оптимальний склад продуктового кошику самостійно для багатьох людей, яким необхідно власноруч контролювати рівень цукру в крові, майже неможливо. Це робить актуальною необхідність розробки програмного засобу, який максимально спростить та пришвидшить процес таких обчислень.

Хлібна одиниця (ХО) – це штучний параметр для оцінки вмісту вуглеводів в конкретному продукті харчування. Одна хлібна одиниця рівнозначна 10 г простих вуглеводів в продукті або 12 г вуглеводів включаючи клітковину. Щоденне обмеження щодо споживання вуглеводів при цукровому діабеті будь-якого типу становить 18-24 ХО. У випадку, коли індекс маси тіла (ІМТ) людини не перевищує показник у 18,49 одиниць – тобто спостерігається недостатність ваги – щоденне обмеження споживання вуглеводів становить 18-

30 ХО. Якщо ІМТ перевищує 35 одиниць – кількість вуглеводів на день становить 10 ХО. При ІМТ вище 40 одиниць – 6-8 ХО на день.

Окрім цих рекомендацій існують обмеження споживання вуглеводів в залежності від віку та статі людини, а також від статі людини та рівня її фізичного навантаження.

Розглянемо поведінку споживача, який має намір за добу закупити продукти в кількості: x_i , $i = 0, 1 \dots n - 1$, за ціною p_i . На закупку продуктів споживач виділяє z гривень. Вказані величини повинні бути зв'язані співвідношенням:

$$\sum_{i=0}^{n-1} p_i x_i = z, z > 0$$

На суму z товари можуть бути закуплені неоднозначно. Тому з усіх варіантів закупок споживач повинен обрати найкращий.

Для цього покупець повинен обрати стратегію закупок:

$$z \rightarrow \min$$

$$P_1 < \sum_{i=0}^{n-1} x_i * XO_i < P_2 \quad 1.2)$$

де P_1 – мінімальна кількість хлібних одиниць,

P_2 – максимальна кількість хлібних одиниць,

XO_i – кількість хлібних одиниць в i -му продукті.

У якості методу оптимізації математичної моделі було обрано послідовну версію методу найменших квадратів з методології квадратичного програмування.

Послідовне квадратичне програмування – один з найбільш ефективних алгоритмів загального призначення, головною ідеєю якого є послідовне вирішення задач квадратичного програмування, апроксимуючих певну задачу оптимізації.

Програмний додаток реалізовано на основі архітектурного шаблону проектування MVVM (Model – View – ViewModel), який передбачає розмежування візуальної частини додатку від внутрішньої логіки його роботи:

1. Model – описує дані, які використовуються в додатку. Моделі можуть містити логіку валідації даних, проте не повинні визначати логіку відображення цих даних або взаємодії з елементами графічного інтерфейсу.

2. View – представлення графічного інтерфейсу, з яким взаємодіє користувач. Представлення визначає перелік та розміщення графічних елементів керування та логіку взаємодії користувача з ними за допомогою команд.

3. ViewModel – модель представлення – своєрідний посередник між моделлю та її представленням. Модель представлення призначена для переправлення даних, введених через графічний інтерфейс до моделі, та навпаки – при зміні даних в моделі – відобразити ці зміни у графічному інтерфейсі. Також модель представлення реалізує команди взаємодії з графічним інтерфейсом, наприклад, команду зміни представлення (навігацію).

Висновки. Створений програмний додаток призначений для використання в побуті людьми, які мають бажання чи яким необхідно контролювати рівень цукру в крові і надасть можливість виконувати наступні дії:

- проводити оптимізацію свого раціону харчування;
- створювати обліковий запис у системі для збереження результатів оптимізації;
- авторизуватися в системі для отримання можливості додавати користувальницькі продукти харчування та обмеження щодо їхнього споживання;
- додавати та редагувати користувальницькі продукти харчування та обмеження щодо їхнього споживання;
- переглядати список стандартних та користувальницьких продуктів та обмежень;
- додавати нові продукти до списку стандартних продуктів харчування;
- додавати нові стандартні обмеження щодо споживання продуктів харчування.

УДК 621.391.14:519

М.О. Алексєєв¹, О.М. Алексєєв¹, А.Є. Ляшенко¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

КОНТРОЛЬ ПАРАМЕТРІВ РОТОРНИХ ОБ'ЄКТІВ З ВИКОРИСТАННЯМ ЧАСТОТНО-ХВИЛЕВОГО СПЕКТРАЛЬНОГО АНАЛІЗУ

Анотація. В роботі пропонується метод контролю функціонального стану роторного об'єкту по частотно-хвильовому спектру звукометричного сигналу. Метод дозволяє визначити координати джерела звукометричного сигналу.

Ключові слова: звукометричний сигнал, частотно-хвильовий спектр, координати джерел обурення.

Вступ. При функціонуванні роторних об'єктів управління, які є складними динамічними системами, що складаються з безлічі підсистем, що взаємодіють, виникають обурювальні сили і моменти, прикладені в різних місцях об'єкта. Реакцією динамічної системи на сили, що обурюють, є

механічні і звукові коливальні процеси, тобто вібрація і звуковий сигнал, які залежать від величини, характеру і місця дії збурювальних сил. Тому завдання керування роторними об'єктами управління може бути вирішена більш ефективно, коли, поряд з оцінкою параметрів сигналів як реакції сили, що збурює, можна співвіднести зміни в спектрі сигналів і місцезнаходження джерел сигналів в об'єкті.

Постановка задачі. Для об'єктів, що у стані статистичного рівноваги, можуть застосовуватися методи аналізу стаціонарних однорідних полів. Для отримання ознак, інваріантних до часового та просторового зсуву, використовуємо як ознаки частотно-хвильовий спектр Фур'є. Просторово-часову функцію $S(t, \mathbf{v})$ можна у вигляді розкладання Фур'є в такий спосіб (для випадку плоских хвиль):

$$S(t, \mathbf{v}) = \int_{-\infty}^{\infty} \int G(f, \mathbf{k}) \exp[2\pi j(ft - \mathbf{k}\mathbf{v})] dt d\mathbf{k} ,$$

де $\exp[2\pi j(ft - \mathbf{k}\mathbf{v})]$ - монохроматична хвиля; \mathbf{v} - вектор, що визначає координати точки, в якій оцінюється $S(t, \mathbf{v})$; \mathbf{k} - векторне хвильове число (вектор "просторової частоти"); $G(f, \mathbf{k})$ - взаємна за простором спектральна щільність.

Тоді частотно-хвильовий спектр, пов'язаний перетворенням Фур'є з взаємною простором спектральної щільністю, можна представити у вигляді:

$$E(\mathbf{k}, f) = \int_{-\infty}^{\infty} G(f, \mathbf{k}) \exp(-j\mathbf{k}\mathbf{v}) d\mathbf{v}$$

або з функцією частотно-хвильової кореляції у вигляді:

$$E(\mathbf{k}, f) = \int_{-\infty}^{\infty} \int R(\tau, \mathbf{v}) \exp[-j2\pi(ft - \mathbf{k}\mathbf{v})] d\tau d\mathbf{v}$$

де багатовимірна автокореляційна функція дорівнює математичному очікуванню

$$R(\tau, \mathbf{v}) = M[S(t, \mathbf{v}_0)S(t + \tau, \mathbf{v}_0 + \mathbf{v})]$$

Розглянемо монохроматичну хвилю виду:

$$S(t, \mathbf{v}) = A \cos[2\pi f_0(t - \alpha\mathbf{v}) + \Theta] = a \cos[2\pi(f_0 t + \mathbf{k}\mathbf{v}) + \Theta]$$

де α - вектор, що вказує напрямок поширення хвилі;

Θ – випадкова змінна, рівномірно розподілена в $[0, 2\pi]$.

Частотно-хвильовий спектр для цього випадку можна уявити:

$$\begin{aligned} E(f, \mathbf{k}) &= \int_{-\infty}^{\infty} \int (A^2/2) \cos[2\pi(f_0\tau - \mathbf{k}\mathbf{v})] \\ &\exp[-j2\pi(ft - \mathbf{k}\mathbf{v})] d\tau d\mathbf{v} = \\ &= (A^2/4) [\delta(f - f_0, \mathbf{k} - \mathbf{k}_0) + \delta(f + f_0, \mathbf{k} + \mathbf{k}_0)] . \end{aligned}$$

При такому визначенні частотно-хвильовий спектр є дельта-функцією, зосередженою на частоті f_0 та хвильовому числі \mathbf{k}_0 . Це визначає можливість використання частотно-хвильового спектру для визначення напрямку поширення хвилі, що біжить.

У системи цифрової обробки випадкових полів зазвичай надходять масиви чисел, отримані шляхом дискретизації поля просторовими змінними. Для випадкових полів інтервал дискретизації повинен бути обраний так, щоб прямокутна область, обмежена верхніми граничними частотами хвильового спектру (k_{xgr} , k_{ygr}), лежить усередині прямокутної області, яка визначається половинами частот дискретизації ($k_{xd}/2$, $k_{yd}/2$). Отже, повинні виконуватися відомі нерівності $k_{xgr} \leq k_{xd}/2$, $k_{ygr} \leq k_{yd}/2$. Ця умова еквівалентна теоремі Котельникова про дискретизацію одновимірних сигналів, у якій сформульовано вимогу, що частота дискретизації має хоча б удвічі перевищувати найвищу частоту спектра сигналу..

Наукова новизна складається в розробці методу контролю параметрів роторних об'єктів керування на основі використання частотно-хвильового спектрального аналізу звукометричного сигналу при керуванні роторними об'єктами.

Висновок. Використання частотно-хвильових спектрів, визначених шляхом обробки звукометричних сигналів роторних об'єктів, що надходять не менше ніж від двох решіток датчиків сигналів, порівняно з методами одновимірних Фур'є сигналів дозволяє визначити не тільки спектральні параметри сигналів, але і координати джерел обурення, що підвищує точність контролю функціонування роторних об'єктів за рахунок точного визначення області максимального випромінювання звукової енергії.

ПЕРЕЛІК ПОСИЛАНЬ

1. Charles H. Sherman, Sung H. Ko, Barry G. Buehler. Measurement of the turbulent boundary layer wave-vector spectrum // J. Acoust. Soc. Am. 1990. V. 88.
2. Long D.F. Effect of nozzle geometry on turbofan shock cell noise at cruise // 46rd AIAA Aerospace Sciences Meeting. 2005.

РОЗРОБКИ ІНФОРМАЦІЙНОЇ СИСТЕМИ КОНТРОЛЮ ЗА ПОВОДЖЕННЯМ З ВІДХОДАМИ

Анотація. Виконано бізнес-аналіз для розробки інформаційної системи контролю за поведінням з відходами, економічний аналіз ресурсів, прибутків окупності та рентабельності проекту інформаційної системи. Обґрунтовано застосування інтернет-аналітики для аналізу поведінки користувачів та підвищення ефективності взаємодії користувача з інформаційною системою.

Ключові слова: екологія, бізнес-аналіз, проект, вимоги, розробка, інформаційна система, відходи, технічне завдання, додаток.

Вступ. Сучасні екологічні проблеми роблять нас уразливими для лих та трагедій - як зараз, так і в майбутньому. По всьому світу люди стикаються з безліччю нових та складних екологічних «збоїв». Деякі з них малі і зачіпають лише кілька екосистем, інші різко змінюють умови життя, які ми вважаємо нормальними.

Науково-технічна революція та недбале використання корисних копалин землі призвело до того, що екологічна ситуація на нашій планеті погіршується буквально на очах. Рівень забруднення надр, гідросфери та повітряного шару землі наближається до критичного рівня. Людство стоїть на порозі світової катастрофи техногенного характеру. На щастя, дедалі більше державних та громадських організацій розуміє глибину та небезпеку проблеми. З огляду на все зазначене компанія «Contexht», яка обрана як базове підприємство, вирішила створити інформаційну систему по вирішенню проблем з екологією.

Опитаними в ході дослідження експертами було представлено головні проблеми суббасейну Дніпра:

- забруднення органічними речовинами, як результат недостатньої очистки стічних вод або ж її відсутності;
- забруднення біогенними елементами (зокрема азотом і фосфором);
- забруднення небезпечними речовинами;
- гідроморфологічні зміни (спрямлення та зарегулювання русел річок).

Крім цих головних проблем, до переліку слід включити забруднення побутовими відходами (зокрема пластику) та зміни клімату (з паводками та посухами).

Всі перелічені фактори обумовлюють об'єкт дослідження даної роботи, а саме процес розробки інформаційної системи для розв'язання екологічних проблем міста Дніпро.

Відповідно метою роботи є підвищення ефективності поводження з відходами шляхом залучення громадян до їх сортування і контролю з використанням розробленого додатку.

Предметом дослідження є бізнес-аналіз розробки інформаційної системи контролю поводження з відходами, який збільшує мотивацію користувачів до сортування відходів та їх подальшої переробки.

Постановка задачі. Для досягнення поставленої мети в роботі вирішуються наступні задачі:

- Аналіз екологічних проблем міста Дніпро, факторів що їх визначають та можливих шляхів вирішення.
- Аналіз сучасних методів бізнес-аналізу та вибір інструментарію для добування, структурування і формалізації вимог до програмного забезпечення поводження з відходами.
- Виконання бізнес-аналізу та підготовка документації для розробки інформаційної системи контролю за поводження з відходами.
- Економічний аналіз ресурсів, прибутків окупності та рентабельності проекту інформаційної системи.
- Застосування інтернет-аналітики для аналізу поведінки користувачів та підвищення ефективності взаємодії користувача з інформаційною системою.

Основний зміст роботи. В роботі використані наступні методи дослідження:

- Метод аналізу літературних джерел і збору експертної інформації для аналізу екологічних проблем міста Дніпро, факторів що їх визначають;
- Метод мозкового штурму для формування набору ідей, потенційно корисних для розв'язання екологічних проблем міста Дніпра;
- Методи проектного аналізу для розробки проекту створення інформаційної системи;
- Метод RACI матриць для розподілу завдань на всіх етапах розробки проекту між виконавцями з їх обов'язками.

Вивчення екологічної ситуації в місті починалося з опрацювання його екологічної історії та розробки правильних програм для вирішення проблем. Далі проводилася дослідження сучасних екологічних умов та дослідження надання послуг суспільства, що пов'язані з екологічними темами. Це дозволило нам оцінити стан екології в місті Дніпрі та виявити основні проблеми.

Команда вирішила використати «Brainstorming» activity для того щоб зібрати ідеї завдяки яким можна буде вирішити екологічні проблеми. «Brainstorming» activity – це мозковий штурм, метод, що поєднує в собі невимушений, неформальний підхід до вирішення проблем із стороннім мисленням.

В ході мозкового штурму команда отримала декілька ідей, як побороти проблеми екології, а саме:

1. Створити систему сортування сміття в офісах компанії.

2. Створити сайт, що дозволить аналізувати витрати води в офісах компанії.

3. Створити додаток для заохочування переробки пластикових пляшок.

4. Заохотити користувачів до свідомого споживання та переробки

Перш за все компанія вирішила почати з себе, а саме вирішенню проблем які можуть створюватись в самих офісах компанії. Так як офіси компанії знаходяться в декількох країнах світу, головний офіс знаходиться в Лондоні, Великобританія та у компанії є офіси в Україні, Білорусі, Польщі та Португалії. Статистика продукування сміття різних видів після впровадження сортування наведена в таблиці на рисунку 1.

Офіс	Вироблено відходів (кг)	Пластику (кг)	Паперу (кг)	Скла (кг)	Електроніки (кг)	Металу (кг)	Органіки (кг)
Україна	2790	558	1395	139.5	27.9	279	393.6
Польща	4050	810	2025	202.5	40.5	405	573.4
Велика Британія	13310	2662	6653	665.3	133.1	1331	1883.4
Португалія	1350	270	675	67.5	13.5	135	189
Білорусі	810	162	405	40.5	8.1	81	114.4

Рис. 1. Статистика сортування сміття за його походженням за 3 міс

На думку більшості співробітників компанії «Contexht» ідея сортування сміття виявилася дуже привабливою з точки зору простоти дотримання, вартості та відносної легкості втілення, одже вони негайно розпочали сортування сміття. Так як на законодавчому рівні в усіх країнах де існують офіси компанії «Contexht», окрім України сортування вже відбувалося. Керівництво та співробітники Українських офісів високо оцінили таку ініціативу. Компанія почала вираховувати скільки тон сміття кожен офіс виробляє кожного місяця. Завдяки впровадженій системі сортування сміття та аналітиці, вже через три місяці можна було побачити та оцінити прогрес.

Друга ідея яка з'явилася під час мозкового штурму – це враховувати скільки води кожний офіс витрачає в місяць. Компанія «Contexht» вирішила створити сайт на якому можна було б дивитися статистику з кожного офісу та дивитися скільки в середньому літрів води на кожну людину витрачається. Цю ідею підтримало керівництво, та команда розпочала працювати в цьому напрямку. В ході дослідження були отримані показники, що ілюструються рис.2.

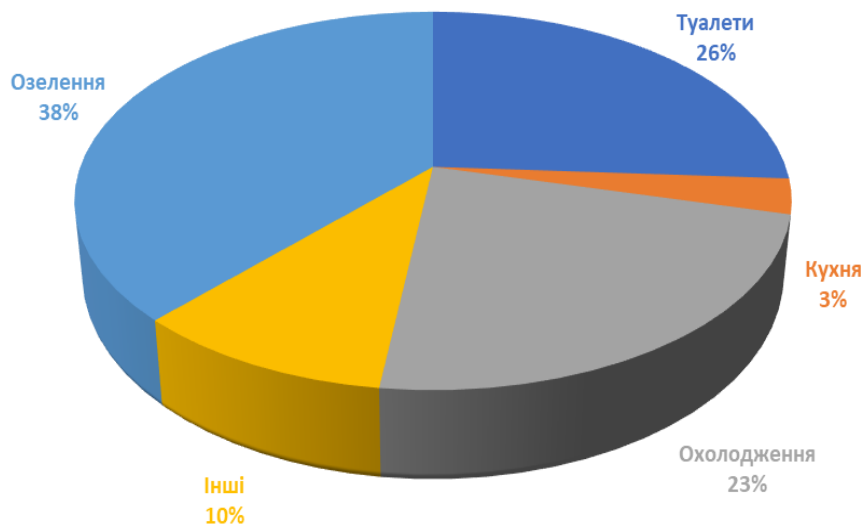


Рис. 2. Використання води в середньому для офісу.

Також компанія вирішила створити систему звітів, де вони відстежують куди та скільки було використано води кожного кварталу щороку. Це було створено з метою заощадження води, всупереч очікуванню щодо заощадження коштів. У національному масштабі використання води поза домівкою становить 30 відсотків від домашнього використання, але може бути набагато вищим у посушливіших частинах країни та у ландшафтах з більшою витратою води. Наприклад, посушливі регіони України мають одне з найвищих показників споживання води на душу населення завдяки фермерським угіддям.

Наступним екологічним проектом команди «Contexht» стала розробка пілотного проекту із переробки пластикових пляшок, завдяки якому легко переробляти пластикові пляшки (кредит надходить на картку «helpful»). За попередніми підрахунками, це допоможе скоротити викиди CO₂ в країні на 160 000 тон, що еквівалентно виведенню з доріг 83 000 автомобілів.

Як працює «Recycle»? Якщо у людини є кілька пластикових або скляних пляшок, треба завантажити програму Recycle, відсканувати штрих-коди, викинути відскановані пляшки до відповідного сміттевого баку, потім отримати кешбек у розмірі 1 або 2 гривні за пляшку на свою кредитну картку.

Компанія створила сайт з калькуляції ресурсів, де над цим продуктом працювала уся команда системних аналітиків та бізнес аналітиків. Вони розрахували та створили продукт який допомагає в повсякденному житті для таких країн як Україна та Велика Британія. Деякі інтерфейси додатку ілюструє рис. 3.

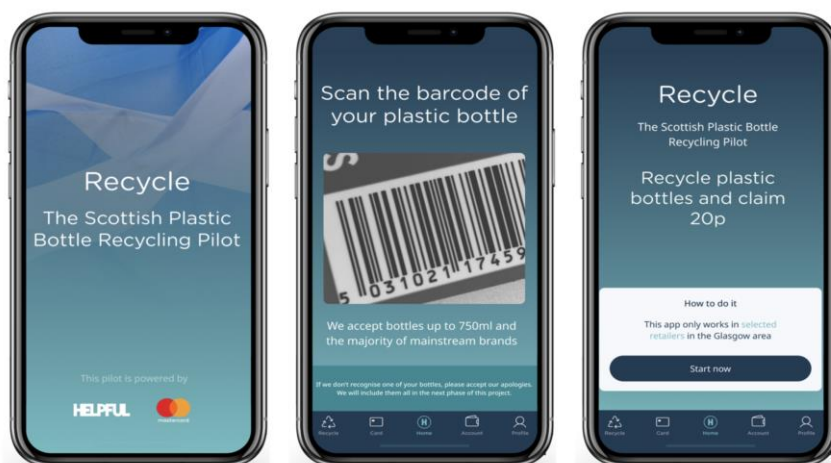


Рис. 3. Приклади інтерфейсів програмного продукту «Recycle».

Також в ході виконання роботи був виконаний економічний аналіз ресурсів, прибутків окупності та рентабельності проекту інформаційної системи. Показана ефективність запропонованої інформаційної системи та її окупність.

Зокрема, чиста теперішня вартість (Net Present Value) виявилась позитивною (131,5 тис. грн. при початкових інвестиціях 500 тис. грн.). це відносно невисокий показник для IT-проекту, але не слід забувати, що мова йде про програмне забезпечення, спрямоване на захист екології та навколишнього середовища. Відтак, комерційний прибуток не є головною самоціллю проекту.

Внутрішня норма прибутку (IRR) – це показник, який використовується у фінансовому аналізі для оцінки прибутковості потенційних інвестицій – за підрахунками виявилась рівною 16,61%. Це значно вище, ніж прибутковість від інвестицій наприклад у банківську діяльність, хоча й значно менше ніж в середньому по IT-галузі.

Загалом результат роботи полягає насамперед у формуванні екологічної свідомості, і лише в другу чергу – в фінансових результатах діяльності.

Наукова новизна розробки полягає у використанні методів проектного аналізу для розробки проекту створення інформаційної системи підвищення екологічної свідомості. Проведені ряд процедур з бізнес-аналізу та підготовлено документацію для розробки інформаційної системи контролю за поводженням з відходами.

Висновки. В роботі за допомогою методу мозкового штурму виконано формування набору ідей, потенційно корисних для розв’язання екологічних проблем міста Дніпра. Побудовані RACI матриці для розподілу завдань на всіх етапах розробки проекту між виконавцями з їх обов’язками. Виконаний економічний аналіз ресурсів, прибутків окупності та рентабельності проекту інформаційної системи. Показана ефективність запропонованої інформаційної системи та її окупність. Запропоновано застосування інтернет-аналітики для

аналізу поведінки користувачів та підвищення ефективності взаємодії користувача з інформаційною системою.

ПЕРЕЛІК ПОСИЛАНЬ

1. Чабанюк О.М. Екологічні проблеми України та шляхи їх вирішення / О.М. Чабанюк, І.В. Боляк // «Сталий розвиток країни в рамках Європейської інтеграції». – Житомир: «Житомирська політехніка», 2020. – с. 31.

2. Laplante, Ph. Requirements Engineering for Software and Systems (вид. 1st). Redmond, WA: CRC Press. - 2009. – 264 p.

УДК 004.415.3:681.6

В.В. Спирінцев¹, А.Л. Ширін¹, А.А. Мартиненко¹, С.А. Шолойко¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ДОСЛІДЖЕННЯ ФРЕЙМВОРКУ VUE.JS

Анотація. У роботі розглянуто основні переваги сучасного фреймворку Vue.js. Розкрито основні концепції побудови та зроблено порівняльний функціональний аналіз фреймворку Vue.js з його основними конкурентами – React, AngularJS та Ember.js.

Ключові слова: *Vue.js, фреймворк, веб-додаток, застосунок, інтерфейс користувача, веб-сервіс, компоненти, директиви.*

Вступ. З плином часу дуже стрімко розвивається веб-індустрія – з’являється все більше нових технологій та інструментів для створення веб-ресурсів. Деякі з них скеровані на полегшення роботи розробників, адже дуже велику кількість механізмів можна автоматизувати, скорочуючи час розробки. Гарним прикладом є фреймворки, які не тільки спрощують розробку, а й дозволяють стандартизувати вихідні продукти. Одним з таких є Vue.js – прогресивний фреймворк для створення інтерфейсів користувача (UI). На відміну від фреймворків-монолітів, Vue створений придатним для поступового впровадження, що спрощує інтеграцію з іншими бібліотеками та існуючими проектами.

Основний зміст роботи. Фреймворк (англ. Framework – каркас, структура) – шаблон для програмної платформи, що визначає уніфіковану структуру для побудованих на його базі додатків, на його основі можливо дописати власний код. Нажаль, більшість розробників вважають, що фреймворк – це набір бібліотек, але це не зовсім так. Бібліотеки та фреймворки між собою різняться потоком управління. Бібліотека містить функції для виклику батьківським кодом, у той час, як фреймворк опирається на структуру додатків у цілому – розробник не викликає його код, це фреймворк викликає код розробника і користується ним. [1]

Переваги використання фреймворків:

- Технологія менш затратна в економічному сенсі;
- Результат більш якісний, ніж написання проекту без використання жодних платформ;
- Проекти простіше супроводжувати та допрацьовувати, оскільки їх стандартизована структура залишається однаковою в усіх проектах;
- Полегшують роботу розробників, оскільки містять велику кількість підключених бібліотек;
- Вища швидкість роботи що забезпечено самою технологією.

Недоліки використання фреймворків:

- Більшість мають відкритий код, отже додатково потрібно буде вирішувати питання безпеки проекту;
- Обмежують розробників своєю архітектурою, якщо потрібно буде розробити нестандартний проект, від них прийдеться відмовитися.

Vue.js – один з найпопулярніших JavaScript-фреймворків з відкритим кодом. Використовується для створення UI, легко інтегрується в проект з іншими JavaScript-бібліотеками. [2] Vue.js призначений для спрощення та оптимізації процесу розробки веб-застосунків, а також досягнення балансу між його складністю та функціональністю. Він побачив світ у 2014 році і, завдяки його динамічності й доступності, із рядового середовища розробки він став однією з найулюбленіших платформ серед професіоналів. Сьогодні Vue.js користуються такі компанії, як Alibaba, Xiaomi, WizzAir, EuroNews, Sina Weibo, Baidu, а також система управління репозиторіями GitLab. [3]

У Vue.js є основні концепції – конструктор, директиви, компоненти, переходи.

Конструктор об'єктів.

Він є ядром Vue.js – ця функція дозволяє створювати екземпляри Vue. Коли ви такий створюєте, вам необхідно передати опції об'єкту – це звичайний набір іменованих параметрів на мові JavaScript, які передаються у функцію. Вони можуть містити інформацію про DOM-елемент, об'єкт даних, змішані методи, зворотні виклики життєвого циклу тощо. Сам конструктор також надає глобальний API, який дозволяє розширювати клас, налаштовувати глобальні параметри та реєструвати глобальні спеціальні ресурси, такі як компоненти, директиви, фільтри тощо.

Директиви.

Це спеціальні команди для додавання додаткового функціоналу, що записані як атрибути HTML-тегів. У Vue.js всі директиви мають префікс «v-», а деякі з них можна передавати додаткові аргументи через двокрапку. Найпоширенішими з вбудованих директив є: v-model; v-on; v-bind; v-once; v-html; v-for; v-if та v-else.

Компоненти.

Допомагають розширити основні елементи та впровадити повторно використаний код, тобто частини UI. На етапі проектування, зазвичай, розбивають застосунок на декілька незалежних частин й отримують

деревоподібну структуру компонентів (рис. 1). В Vue.js немає особливих вимог щодо імен компонентів, але більшість розробників користуються правилами W3C (The World Wide Web Consortium - міжнародна спільнота, де організації-члени, штатний персонал і громадськість разом працюють над розробкою веб-стандартів), тобто у назвах використовуються літери нижнього регістру, розділені дефісом.

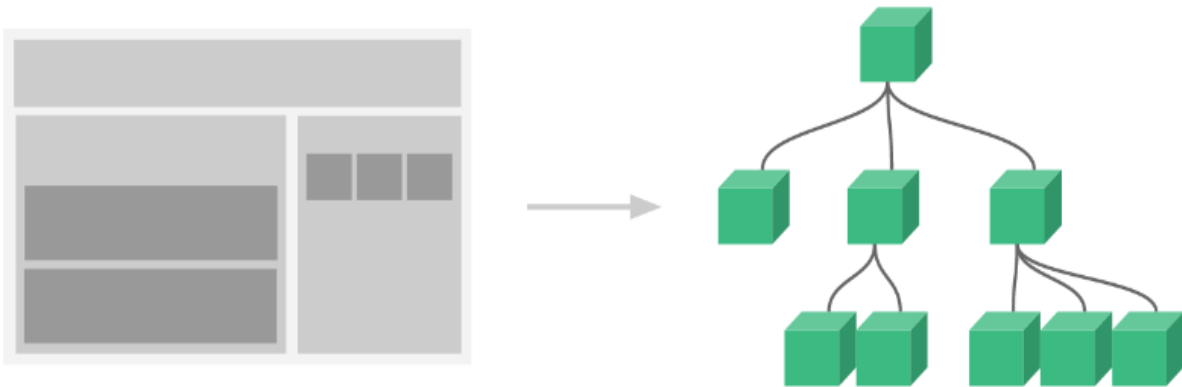


Рис. 1. Деревоподібна структура веб-застосунку.

Переходи.

Vue надає різні способи застосування анімаційних ефектів, коли елементи намальовані, поновлені або видалені з DOM. Вони включають у себе інструментарій для автоматичного застосування класів для CSS-переходів та анімацій, інтеграцій сторонніх CSS- та JavaScript-бібліотек, наприклад, Animate.css або Velocity.js, використання JavaScript для маніпуляцій з DOM. [4]

Нижче наведено порівняльний функціональний аналіз фреймворку Vue.js з основними його конкурентами – React, AngularJS та Ember.js.

Порівняння з React.

Vue.js відрізняється від React тим, що всі елементи інтерфейсу у React ми малюємо через так звані render-функції. Цей підхід має свої переваги, але також змушує йти на компроміси, які можуть бути неефективними для кожного окремого розробника. Vue, в свою чергу, охоплює класичні веб-технології та ґрунтується на них. Розробники Vue вважають, що шаблони – простіша альтернатива, хоча у Vue теж є render-функції і навіть підтримка JSX (синтаксис React).

Із схожих моментів можна виділити:

- використання Virtual DOM;
- надання реактивності та компонентів структур;
- фокусування на кореневій бібліотеці, виносячи інші питання, такі як роутинг або управління глобальним станом програми, в додаткові бібліотеки.

Порівняння з AngularJS.

Деякі частини синтаксису Vue (директиви v-if, v-for) виглядають дуже схожим на синтаксис AngularJS (директиви ngIf, ngFor). Але Vue значно

простіший, як у сенсі API, так і в сенсі архітектури. AngularJS має жорстку структуру веб-додатку, тоді як Vue виявляє гнучкість і є більш модульним рішенням. У Vue є система для швидкої розробки та створення прототипів (CLI), яка має стати стандартним базовим інструментом для екосистеми Vue. Це гарантує, що різні інструменти збирання будуть працювати разом з оптимальними налаштуваннями за промовчанням, що дозволить зосередитися на створенні програми, а не витратити час на конфігурування. У той же час Vue, як і раніше, надає гнучке налаштування конфігурації кожного інструменту відповідно до конкретних потреб.

Порівняння з Ember.js.

Vue забезпечує ненав'язливу реактивність на прості об'єкти JavaScript і повністю автоматичні обчислювальні властивості, в той час у Ember.js потрібно обернути все в Ember Objects і вручну оголосити залежності для обчислених властивостей. Синтаксис шаблону Vue використовує всю потужність виразів JavaScript, тоді як вираз і синтаксис помічника Handlebars навмисно досить обмежені в порівнянні. З точки зору продуктивності, Vue перевершує завдяки автоматичному пакетуванню оновлень, тоді як в Ember потрібно вручну керувати циклами запуску в критичних для продуктивності ситуаціях. [5]

Висновки. Однозначно кращого фреймворку на сьогодні не існує, але в останні роки Vue.js стає найбільш затребуваним для розроблення невеликих проектів і динамічних інтерфейсів, що адаптовані під користувача. Vue.js має гарну масштабованість і високу продуктивність, тому справляється з об'ємними односторінковими застосунками, а завдяки своїй функціональності й динамічності стрімко стає однією з найулюбленіших платформ серед веб-розробників.

ПЕРЕЛІК ПОСИЛАНЬ

1. Framework vs Library: Full Comparison, InterviewBit, 2021 [Електронний ресурс]. URL: <https://www.interviewbit.com/blog/framework-vs-library/> (дата звернення: 04.11.2022)
2. Vue.js guide: електронний збірник документації. [Електронний ресурс]. URL: <https://vuejs.org/guide/introduction.html> (дата звернення: 04.11.2022)
3. Робота з фреймворком Vue.js [Електронний ресурс]. URL: <https://indigo.co.ua/ua/technologies/vue-js> (дата звернення: 04.11.2022)
4. Comparison with Other Frameworks: електронна документація [Електронний ресурс]. URL: <https://v2.vuejs.org/v2/guide/comparison.html?redirect=true> (дата звернення: 04.11.2022)
5. The Guide to Tech: Vue.js, 2017 [Електронний ресурс]. URL: <https://stfalcon.com/en/blog/post/vue-js-guide-to-tech> (дата звернення: 04.11.2022)

БІЗНЕС-АНАЛІЗ ВИМОГ ДО ІНФОРМАЦІЙНОЇ АРХІТЕКТУРИ ФІТНЕС ДОДАТКУ ТА ЙОГО ПРОЕКТУВАННЯ

Анотація. Розглянуто тему гейміфікації цифрових продуктів. Проведено дослідження користувачів, вивчено вплив гейміфікації на їх мотивацію до занять спортом, спроектована інформаційна архітектура, та на її основі спроектовано дизайн інтерфейсу додатку. Основна перевага додатку полягає в підтримці мотивації користувачів до занять спортом та тим самим в підвищенні залученості до використання продукту.

Ключові слова: мотивація, гейміфікація, користувач, UX дослідження, user interview, user survey, user stories, usability, інформаційна архітектура.

Вступ. За спостереженнями фітнес тренерів понад 50% людей, які починають займатися спортом, після перших трьох місяців, завершують тренування та відвідування спортивних залів. Причинами часто є відсутність конкретної мети та консистентності тренувань, що несуть за собою втрату мотивації.

Гейміфікація – прийом, який пішов з комп'ютерних ігор, все частіше використовується поза ігрової сфери. Додатки онлайн банкінгу, шопінгу чи менеджменту використовують методи гейміфікації для підвищення залученості користувачів.

Об'єктом дослідження є процес проектування інформаційної архітектури (концепту) фітнес-додатку.

Метою даної роботи є підвищення мотивації та залученості потенційних користувачів фітнес-додатку до занять спортом та підтримки здорового образу життя шляхом впровадження методів гейміфікації.

Предметом дослідження є застосування методів гейміфікації цифрових продуктів для підвищення мотивації та залученості користувача та оцінка ефективності такого підходу.

Постановка задачі. Для досягнення поставленої мети необхідно вирішити наступні задачі дослідження:

- Підготувати та провести UX дослідження, отримати відгуки та думки потенційних користувачів щодо технік гейміфікації. Провести функціональний аналіз додатків конкурентів та відгуки їх користувачів.
- Обробити отримані під час дослідження дані та сформулювати User Stories, які відображають головні бажання та потреби користувачів.
- Розробити на базі отриманих інсайтів інформаційну архітектуру проекту.
- Розробити дизайн додатку, спираючись на інформаційну архітектуру та інсайти дослідження.
- Провести юзабіліті тестування для оцінки зручності користування додатком користувачами.

- Побудувати первинну базу даних додатку для подальшої роботи над додатком.

Основний зміст роботи. Гейміфікація – це техніка, яка в основному використовується розробниками ігор, але стає все більш популярною для використання в неігровому контексті. Це концепція додавання ігрових елементів у діяльність, щоб підвищити мотивацію та залучення користувачів або учасників. Гейміфікація – це ремесло отримання всіх цікавих і захоплюючих елементів, які є в іграх, і застосування їх у реальних чи продуктивних заходах. Гейміфікація не є новою концепцією, фактично вона використовувалася десятиліттями чи століттями [1].

Деякі поширені методи гейміфікації, які використовуються, це бали, значки та таблиці лідерів, які використовувалися тисячоліттями для навчання, розваги та залучення. Мета цих методів полягає в тому, щоб звернути увагу на певні основні чинники всередині нас, які спонукають нас до певної діяльності. У кожної людини є вісім основних драйверів, які на рисунку 1 розроблені у формі восьмикутника з окремими сторонами, що представляють основні драйвери. Порядок основних драйверів встановлено чітко, оскільки дизайн включає елементи «Лівого мозку та Правого мозку». Остаточний дизайн, який став фреймворком гейміфікації, отримав назву «Ocalysis» [2].



Рис. 1. Фреймворк гейміфікації Ocalysis

У цій кваліфікаційній роботі гейміфікація розглядатиметься як мотиваційна техніка дизайну, яка спонукає користувачів до заняття спортом. Зокрема, буде виявлено, які методи гейміфікації використовуються в популярних додатках для вправ та які цікавлять користувачів найбільше. На рисунку 1, можна помітити величезну кількість різних технік, які звертаються до різних основних драйверів, і для аналізу навіть невеликої кількості додатків для тренувань знадобиться величезна кількість часу. Тому, у цьому дослідженні буде використано лише декілька з них, які будуть доречні в контексті тренувань та занять спортом.

Для вирішення першої задачі дослідження було проведено дослідження користувачів та потенційних конкурентів. Дослідження користувачів було проведено методами інтерв'ю та опитувань, в результаті яких було визначено як різні методи гейміфікації впливають на мотивацію користувачів, та що респонденти про них думають. Під час інтерв'ю також були зібрані дані щодо власних думок про найбільш та найменш улюблені техніки гейміфікації учасників. З рейтингу видно, що функціонал, представлений в методах «Зворотній зв'язок» та «Прогрес», цікавить більшість учасників, тоді як «Челендж» та «Рівні» займають другорядні місця.

Ранжуючи методи, учасники водночас пояснювали свої рішення та надавали відгуки щодо обраних методів. Базуючись на отриманих даних, можна сформулювати інсайти щодо методів гейміфікації та вирішити які з них варто використовувати та в якому контексті. Таким чином було обрано головні методи гейміфікації, які згодом були використані у самому додатку, та потреби користувачів, щодо фітнес додатку, яким вони б хотіли користуватися. Результати досліджень були сформульовані у вигляді User Stories:

1. Як користувач, я хочу мати можливість генерувати тренування, щоб не робити це самостійно та не витратити на це час.
2. Як користувач, я хочу мати можливість відстежувати свій прогрес, щоб знати які навички мені треба покращити.
3. Як користувач, я хочу мати зв'язок з внутрішньою спільнотою, щоб ділитися своїми результатами та надихатися результатами товаришів.
4. Як користувач, я хочу мати можливість обирати інвентар для тренувань, щоб мені було зручно тренуватися вдома.

Крім User Stories, були сформовані загальні інсайти дослідження:

1. Специфічна мета. Користувач має можливість сформулювати специфічну, вимірну, досяжну, реалістичну та обмежену часом мету.
2. База теорії самовизначення. Користувачі мають відчувати контроль на своєю метою; повинні мати можливість поступово набиратися досвіду для досягнення своєї мети; повинні мати відчуття причетності до спільноти.
3. Зручність використання. Людина повинна витратити якомога менше зусиль на вивчення особливостей додатку.
4. Підтримка залученості. Окрім простоти використання, додаток має підкріплювати завдання цікавими челенджами та іграми.

5. Інформативність. Будь яка інфографіка повинна бути підкріплена поясненнями або індивідуальними висновками.

Для отримання простого та інтуїтивно зрозумілого інтерфейсу було попередньо розроблено інформаційну архітектуру додатку [3]. Вона містить в собі 5 основних блоків, які відповідають за персоналізований досвід користування додатком, готові та згенеровані тренування, відстеження прогресу, отримання зворотного зв'язку та внутрішню спільноту. Запропоновану структуру додатку в вигляді Mind Map ілюструє рисунок 2.

Після реєстрації та короткого вступного ролику користувачу запропоновано пройти невелике опитування для визначення його попереднього досвіду та мети користування додатком. Надану інформацію додаток використовує для персоналізації тренувань. Також на цьому етапі користувач вже може обрати доступне йому обладнання для тренувань та встановити час, коли йому зручно займатись вправами.

На головній сторінці користувачу відразу представлено згенероване тренування, котре буде оновлюватись відповідно його графіку тренувань. Далі йде ряд базових тренувань типу бігу, йоги або розтяжки та список готових тренувань націлених на різні групи м'язів, інтенсивність та різної тривалості. Можливість створювати своє тренування також передбачена.

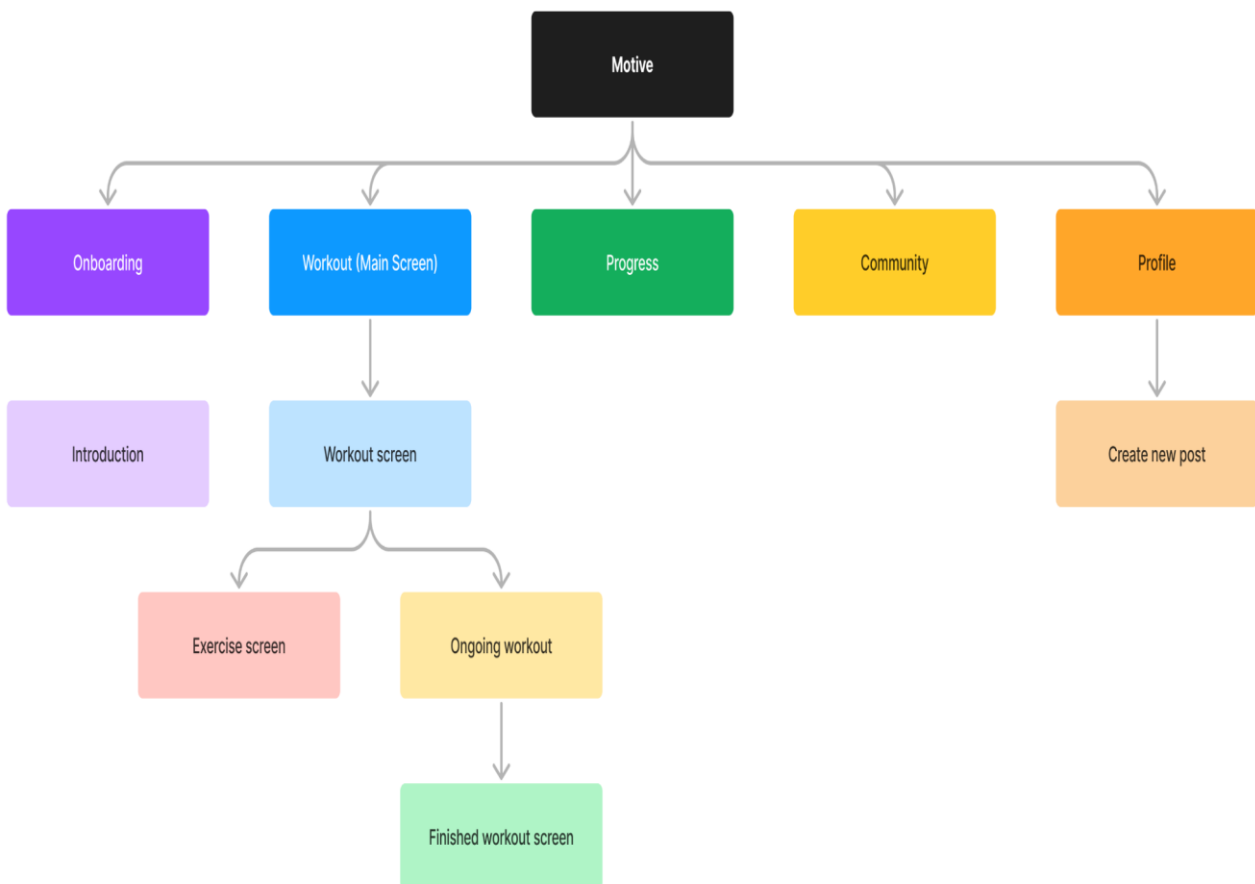


Рис. 2. Узагальнена архітектура додатку

Під час тренування у користувача є можливість редагувати вправи, кількість підходів, вагу, час, тощо. А після закінчення кожного тренування користувачу запропоновано пройти опитування з двох питань для персоналізації його досвіду.

Функціонал блоку прогресу передбачає відстеження плану та щоденника тренувань, різну статистику оснований на попередніх тренуваннях та значення маси тіла користувача.

Внутрішня спільнота дає користувачам можливість ділитися своїми досягненнями та спілкуватися з товаришами в коментарях.

В профілі користувач може оперувати даними його публічного акаунту, створювати пости та побачити досягнення, отримані ним під час тренувань та челенджів.

Дизайн додатку був спроектований спираючись на інформаційну архітектуру та інсайти отримані під час дослідження користувачів. Усього було розроблено 26 екранів додатку, які відображають його основний функціонал.

Під час дослідження стало відомо, що користувачі не знаходять статистику корисною, якщо не знають як нею користуватися, тому на сторінці прогресу користувачу, окрім необхідної статистики, надається пояснення що вона собою представляє.

В результаті дослідження також з'ясувалося, що користувачі більш схильні ділитися спортивними результатами у тематичному ком'юніті ніж у соціальних мережах, якими вони користуються щодня.

Однією з найбільш популярних функцій, котру запитували користувачі став вибір інвентарю для вправ, який можна обрати в налаштуваннях профілю. Після проектування інтерфейсу додатку було проведено юзабіліті-тестування [4], яке показало, що більшість респондентів задоволені та отримали те, що очікували від додатка.

Тестування проводилося онлайн, коридорним методом. Для тестування були запрошені користувачі переважно зі сфери дизайну, у кількості 30 чоловік. Респондентам було дозволено вільно користуватися прототипом без конкретної мети та задачі. В результаті отримано зворотній зв'язок від користувачів. Більшість респондентів відзначило, що отримали від додатка те, на що очікували. 6 з 20 учасників відзначили позитивні моменти, які їм сподобались. Ще 3 учасника відзначили моменти, які викликали в них незручності. Негативні відгуки були проаналізовані та при необхідності в дизайн були внесені правки.

Останнім етапом проектування додатку була розробка бази даних додатку, яку ілюструє рисунок 3. Для її реалізації було застосовано Microsoft Access. Усього вона нараховує 13 пов'язаних між собою таблиць, які відповідають за збереження даних про користувача, готових тренувань, вправ, інвентарю, тощо.

В центрі БД стоїть користувач, від якого йдуть зв'язки до таблиць з його особистими даними, даними для відстеження прогресу, досягненнями та інші.

Наукова новизна розробки полягає у використанні методів гейміфікації з метою підвищення мотивації та залученості потенційних користувачів фітнес-додатку до занять спортом та підтримки здорового образу життя.

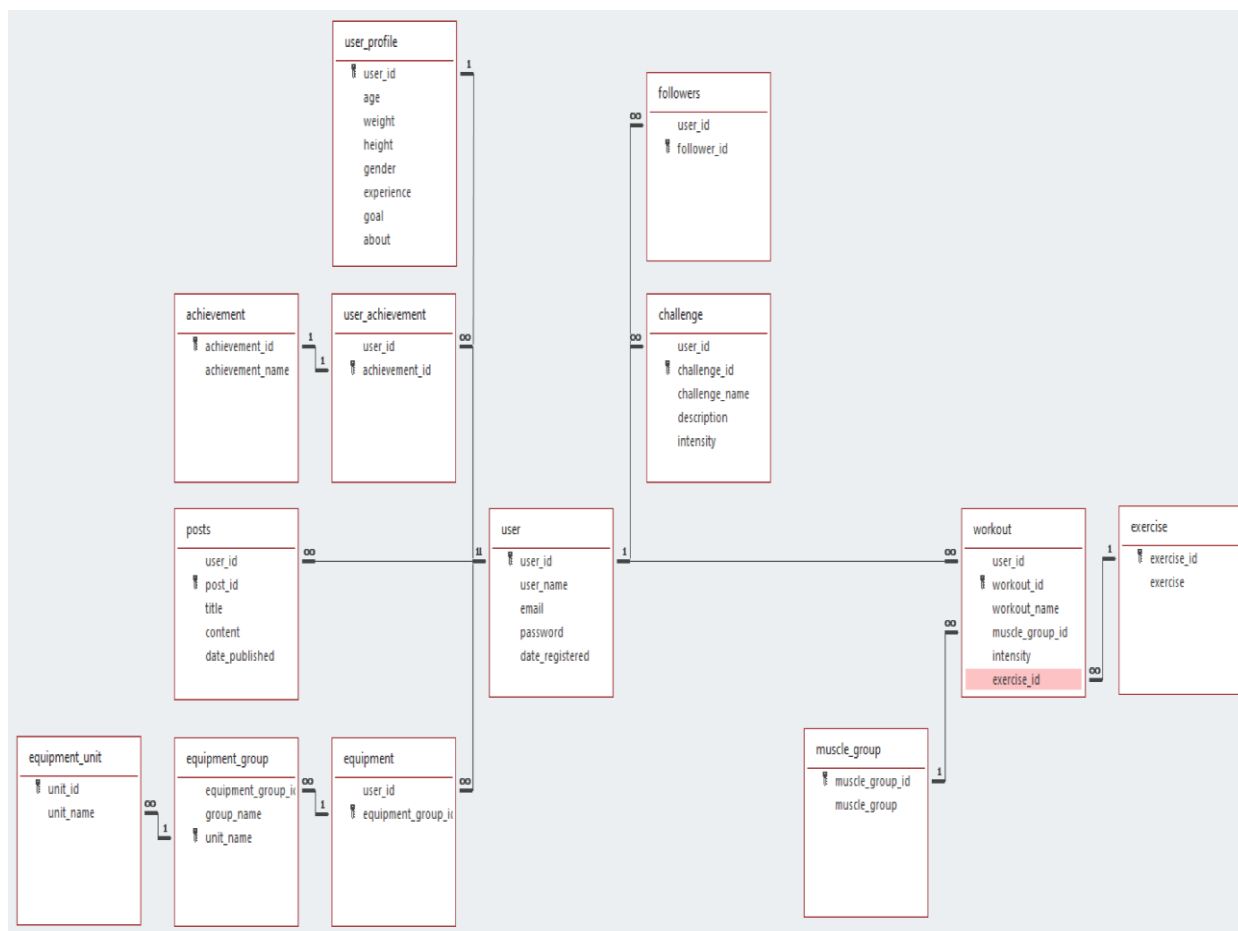


Рис. 3. Структура бази даних додатку

Висновки. В роботі розглянуто тему гейміфікації цифрових продуктів. Проведено дослідження потенційних користувачів, вивчено вплив гейміфікації на їх мотивацію до занять спортом. Як результат дослідження спроектована інформаційна архітектура мобільного додатку, зібрані вимоги користувачів, які були структуровані у вигляді користувацьких історій та інсайтів. На основі багаторівневої архітектури додатку було спроектовано дизайн інтерфейсу додатку. Також розроблена архітектура бази даних додатку, яка містить 13 таблиць, серед яких одна головна таблиця фактів, решта – допоміжні. Основна перевага додатку полягає в підтримці мотивації користувачів до занять спортом та тим самим в підвищенні залученості до використання продукту.

ПЕРЕЛІК ПОСИЛАНЬ

1. Wade Susan L. Self-Determination Theory (Sdt) : Perspective Applications and Impact. NovaScience Publishers. – 2017 – 180 p. URL: <http://site.ebrary.com/id/11332070> . Accessed 16 Feb. 2023.
2. Chou Yu-Kai. Actionable Gamification: Beyond Points Badges and Leaderboards. Octalysis Media. – 2015 – 499 p.
3. Krug Steve. Don't Make Me Think Revisited : A Common Sense Approach to Web Usability. 3rd. ed. New Riders. – 2019 – 200 p.
4. Díaz de Rada, Vidal. Strategies to improve response rates to online surveys. Papers, 107 (4), e3073. – 2022 - <https://doi.org/10.5565/rev/papers.3073>

РОЗДІЛ 4

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЯ СИСТЕМ ОБРОБКИ ІНФОРМАЦІЇ ДЛЯ ВИРІШЕННЯ ЗАВДАНЬ ОСВІТИ, НАУКИ І УПРАВЛІННЯ ВИРОБНИЦТВОМ

УДК 519.673:622.272.6

В.В. Приходько¹, Н.П. Уланова¹, К.Д. Назаренко¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ГЕОМЕХАНІЧНИХ ПРОЦЕСІВ З ВИКОРИСТАННЯМ МЕТОДА ГРАНИЧНИХ ЕЛЕМЕНТІВ

Анотація. В роботі розглянуті питання, пов'язані з моделюванням процесів видобутку корисних копалин на основі аналізу напружено-деформованого стану області породного масиву з виробками, утвореними в результаті підготовки та виїмки корисних копалин. Для визначення НДС такої області використано метод граничних елементів. Тріщинуватість вмшуючих порід моделювалась трансверсально-ізотропним середовищем. Визначено залежність висоти руйнування покрівлі вугільного пласта від геометричних параметрів виробки та послабленості порід внаслідок порушення суцільності масиву. Досліджено вплив регулярної тріщинуватості на періодичні посадки покрівлі вугільного пласта, а також змінення навантаження на кріплення в процесі просування забою.

Ключові слова: *чисельні методи, напружено-деформований стан, регулярна тріщинуватість, тонкий вугільний пласт, залишкова міцність, коефіцієнт структурного послаблення.*

Вступ. Фундаментальні дослідження та пошуки нових рішень зі створення ефективних технологій розробки родовищ корисних копалин – основа розвитку гірничодобувної промисловості, зокрема вугільної. Часто спостерігається зниження показників видобутку, пов'язане з залученням до розробки ділянок родовищ зі складними гірничо-геологічними умовами. Відомо, що найвища інтенсивність проявів гірського тиску у виробках спостерігається в моменти активної зміни напруженого стану породного масиву поблизу забою, що рухається. При цьому вирішальними факторами є зміна геометрії частини масиву та виробленого простору, а також взаємовплив очисного забою та підготовчих виробок.

Задачі, пов'язані з визначенням напруженого стану в області виробок складної геометрії у породному масиві, важко піддаються аналітичним дослідженням. Найчастіше їх розв'язують шляхом чисельного моделювання, враховуючи фізико-механічні властивості пласта і навколишніх порід. З розвитком чисельних методів суттєво розширилися можливості моделювання структурної неоднорідності породного масиву, а також розвитку гірничих робіт у просторі та часі.

Найширше використовується метод кінцевих елементів. Але для отримання рішень із прийнятною точністю застосування цього метода в тривимірній постановці суттєво ускладнює дослідження. Метод граничних елементів також дозволяє враховувати неоднорідність породного масиву – наявність зон зруйнованих порід, шаруватість. Важливо, що застосування цього методу зменшує розмірність задачі на порядок.

Метою роботи є застосування метода граничних елементів для математичного моделювання напружено-деформованого стану області породного масиву поблизу спряження підготовчих виробіток та очисного забою, а також застосування створеної моделі для розв’язання тривимірної задачі теорії пружності для середовища з регулярною тріщинуватістю.

Основний зміст роботи. У випадку тонких вугільних пластів, характерних для родовищ Донбасу, визначення напружено-деформованого стану доцільно виконувати за допомогою однієї з модифікацій методу граничних елементів [1], яка була успішно застосована в ряді робіт [2], [3], [4], [5], [6] для моделювання тривимірної області породного масиву, що містить вугільний пласт, підготовчі та очисну виробки, а також ділянки ціликів та зруйнованих порід. Частково відпрацьований вугільний пласт моделювався контактом, який містить ділянки двох типів – з вільно деформованими поверхнями та заповнені пружним матеріалом, який може мати властивості вугілля або зруйнованих порід. Підготовчі виробки представлені порожнинами аркового перерізу.

Описана область перебуває в умовах трьохосного нерівномірного стиску. Граничною поверхнею є площина, якою апроксимується тонкий вугільний пласт. Порожнини арочної форми, які відповідають підготовчим виробкам, представлені набором прямокутних елементів, нахилених під різними кутами до площини пласта.

Задача розв’язується в додаткових напруженнях. Повні напруження представлені сумою початкових $(\sigma_{ij})_0$, які мали місце до появи виробок, та додаткових $(\sigma_{ij})'$, обумовлених утворенням виробок: $\sigma_{ij} = (\sigma_{ij})_0 + (\sigma_{ij})'$.

Початковий напружений стан заданий компонентами

$$(\sigma_{xx})_0 = (\sigma_{yy})_0 = -\lambda\gamma H, (\sigma_{zz})_0 = -\gamma H,$$

де γ – об’ємна вага навколишніх порід,

H – глибина залягання пласта,

$\lambda = \nu / (1 - \nu)$ – коефіцієнт бічного тиску.

В результаті застосування методу визначаються напруги та переміщення як на граничній поверхні, так і в будь-якій точці досліджуваної області.

Межі області зруйнованих порід в покрівлі пласта, що відпрацьовується, встановлюються за допомогою критерія [7]. В якості граничної напруги

використовується величина σ^n – напруга, обчислена за пружним законом, яка відповідає межі залишкової міцності на триланковій діаграмі стиснення для матеріалу вміщуючих порід.

У випадку порушення суцільності навколишніх порід гранична напруга коригувалася множителем k – коефіцієнтом структурного послаблення, який обчислювався як відношення максимальних переміщень контура виробки у середовищі з тріщинами та без них. Тріщинуватий породний масив моделювався трансверсально ізотропним середовищем, в якому виділено головний напрям тріщин, нахилених під кутом α до площини пласта.

В якості вихідних даних для розрахунків використані такі геомеханічні характеристики: для вміщуючих порід модуль Юнга $E = 3,5 \cdot 10^4$ МПа, коефіцієнт Пуассона $\nu = 0,17$, об'ємна вага $\gamma = 2,5$ т/м³; властивості вугілля $E_g = 3,2 \cdot 10^3$ МПа, $\nu_g = 0,13$. Глибина залягання вугільного пласта $H = 450$ м, його потужність $h = 1$ м. Властивості зруйнованих порід $E_{зр} = 3,0 \cdot 10^2$ МПа, $\nu_{зр} = 0,06$. Характеристики тріщинуватості: вздовж тріщин $E_s = 3,5 \cdot 10^4$ МПа, $\nu_s = 0,17$, по нормалі до тріщин $E_n = 3,5 \cdot 10^3$ МПа, $\nu_n = 0,02$; відстань між тріщинами 0,1 м.

Значення кутів нахилу тріщин α варіювалися в інтервалі від 0° до 90°. Встановлено залежності коефіцієнта структурного послаблення k від кута α (в радіанах)

$$k = 0,068 \cdot e^{1,22\alpha}.$$

Для суцільного та тріщинуватого середовищ визначався параметр «висота руйнування» – максимальний вертикальний розмір зони зруйнованих порід у покрівлі пласта. Варіювалися геометричні параметри – відстань від розрізної печі до лінії забою l в інтервалі від 0 до 80 м при значеннях довжини лави d , що дорівнювали 72 м, 152 м і 216 м, а також значення коефіцієнта послаблення k від 0,068 до 0,37, які відповідають кутам нахилу тріщин від 0° до 90°. В результаті отримано залежність висоти руйнування покрівлі пласта від вказаних параметрів для тріщинуватих порід основної покрівлі

$$H_p^{mp} = 0,072 \cdot l^{0,75} \cdot d^{0,25} / k, \text{ м.}$$

Остання залежність дозволила оцінити шаг посадки покрівлі в процесі посування забою. При цьому користувалися припущенням, що посадка покрівлі відбувається тоді, коли зона зруйнованих порід охоплює по висоті всю основну покрівлю. Значення відстані l , при якій висота руйнування досягає потужності основної покрівлі, інтерпретувалось, як імовірний крок посадки покрівлі

пласта. Передбачалося, що навантаження на механізоване кріплення створюється вагою тієї частини порід, яка знаходиться в зоні руйнування. Навантаження на механізоване кріплення обчислювалось, як вага зруйнованих порід покрівлі, віднесена до сумарної площі всіх секцій кріплення. Шляхом узагальнення результатів отримано залежність навантаження на секцію механізованого кріплення в процесі посування забою

$$F = 0,66\gamma H_p^{mp} \cdot l / l_{кр}, \text{ т/м},$$

де H_p^{mp} – висота руйнування, м;

l – відстань від розрізної печі до лінії забою, м;

$l_{кр}$ – сумарна довжина перекриття кріплення, м;

γ – об'ємна вага навколишніх порід, т/м³.

Це дозволило сформулювати рекомендації щодо закладки виробленого простору для забезпечення безаварійної роботи кріплення.

Висновки. Математична модель, заснована на модифікації метода граничних елементів, застосована для розв'язання тривимірної задачі теорії пружності в області породного масиву поблизу спряження підготовчих виробок та очисного забою. Визначено величину кроку посадки основної покрівлі вугільного пласта в процесі посування забою в породах з регулярною тріщинуватістю. Встановлено, що вплив тріщинуватості полягає в зменшенні кроку первинної посадки покрівлі в 1,57 – 4 разів при вказаних значеннях довжини лави. Обчислено навантаження на кріплення в процесі посування забою при різних значеннях довжини лави та кутах нахилу тріщин до напластування. Для забезпечення безаварійної роботи кріплення при даному варіанті управління покрівлею запропоновано схему часткової закладки виробленого простору смугами шириною 5 м перпендикулярно лінії забою, відстань між смугами 4 м.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Новикова Л.В., Приходько В.В., Уланова Н.П. Напряжения вокруг выработки в массиве трещиноватых горных пород. Изв. вузов. Горный журнал, 1994, №7. – С. 42-45.
2. Prykhodko, V.V., Ulanova, N.P. (2018). Modeling of stress-strain state of fractured rock mass nearby of conjugated workings. *Naukovyi Visnyk Natsionalnoho Hirnychoho Universytetu*, 1, 2018, pp. 5-11. <https://doi.org/10.29202/nvngu/2018-1/15>
3. V. Prykhodko, N. Ulanova, O. Haidai, D. Klymenko (2018). Mathematical modeling of tight roof periodical falling. E3S Web Conf., Volume 60, 2018, «Ukrainian School of Mining Engineering», Berdiansk, Ukraine, September 4–8, 2018. <https://doi.org/10.1051/e3sconf/20186000>
4. Kuzmenko A.M., Ulanova N.P., Prikhodko V.V., Dotsenko V.I. Rock mass with two systems of cracks model // *Geotechnological Issues of Underground Space Use for Environmentally Protected World*, GIUS – 2001, pp. 97-100.

5. А.М. Кузьменко, В.В. Приходько, Н.П. Уланова. Влияние геологического нарушения на напряженное состояние вокруг выработки в массиве, ослабленном трещинами. – Сб. научных трудов «Вестник Криворожского технического университета», вып. 11, Кривой Рог, 2006 – С. 196-198.

6. N.Ulanova, O.Sdvyzhkova, V.Prikhodko. Optimization of room-and-pillar method parameters under conditions of limestone rocks. Progressive Technologies of Coal, Coalbed Methane, and Ores Mining. Taylor & Francis Group, London, ISBN: 978-1-138-02699-5, 2014.– pp.511-516. <https://doi.org/10.1201/b17547>

7. Критерий разрушения структурно неоднородных материалов / А. Шашенко, А. Ковров, Б. Ракишев // Розробка родовищ: Зб. наук. праць — 2016. — Т. 10, вип. 3. — С. 84-89.

УДК 004.415.3:681.6

Л.І. Мещеряков¹, Н.Н. Одновол¹, О.В. Удовик¹, М.І. Немченко¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

РОЗРОБКА ДОДАТКУ ДЛЯ АНАЛІТИКИ КОРИСТУВАЦЬКИХ ДАНИХ ТА ПОБУДОВИ СКОРИНГОВИХ МОДЕЛЕЙ

Анотація. Представлено процес створення додатку для покращення користувацького досвіду при роботі з системами та алгоритмами машинного навчання. Покращення досягається за рахунок оптимізації вхідних даних із застосуванням математичних методів лінійної та логістичної регресій.

Ключові слова: логістична регресія, лінійна регресія, база даних, скорингові моделі, машинне навчання, дерево прийняття рішень, true negative, true positive.

Вступ. У даний час широкого розповсюдження набувають різні онлайн-сервіси та онлайн-платформи. Перед нами постає велике різноманіття інтернет-магазинів, онлайн-кінотеатрів, засобів доставки їжі, платформ для навчання у багатьох освітніх напрямках, сервісів відеозв'язку. Все більше людей користується онлайн-сервісами, а користувацька інформація, яку надають люди все більше зростає в об'ємі.

За останні декілька років можна спостерігати зріст популярності професії аналітика з різними спеціалізаціями. Багато бізнес рішень, планування та стратегій розвитку базується на основі різних показників або метрик. Це зріст створює необхідність спрощення взаємодії користувачів з користувацькими даними.

Наразі входження в сферу аналітики потребує від користувача глибоких знань та експертизи з проектування математичних моделей, обробки великих об'ємів даних та прийняття рішень на основі отриманих розрахунків.

Зараз у сфері аналітики користувацьких даних не так багато готових комплексних рішень, що недостатньо для аналізу та проектування стратегій ведення бізнесу.

Постановка задачі. Для досягнення поставленої мети в роботі сформовані і вирішені такі завдання:

- визначити основні алгоритми машинного навчання;
- спроектувати та розробити програмну частину;
- визначити мінімально необхідні апаратні частини;
- зробити висновки щодо доцільності створення подібної системи.

Основний зміст роботи. Під час роботи були використані наступні методи та інструменти:

- математичні оцінки якості скорингових моделей;
- Python, matplotlib, sclearn, pandas.
- Кінцевий інтерфейс та апаратні вимоги.

Машинне навчання (англ. machine learning, ML) – клас методів штучного інтелекту, характерною рисою яких є не пряме рішення задачі, а навчання за рахунок застосування рішень безлічі подібних завдань. Для побудови таких методів використовуються засоби математичної статистики, чисельних методів, математичного аналізу, методів оптимізації, теорії ймовірностей, теорії графів, різних технік роботи з даними в цифровій формі.

Розрізняють два типи навчання:

- навчання за прецедентами, або індуктивне навчання, засноване на виявленні емпіричних закономірностей даних;
- дедуктивне навчання передбачає формалізацію знань експертів та його перенесення на комп'ютер як бази знань.

Дедуктивне навчання прийнято відносити до галузі експертних систем, тому терміни машинне навчання та навчання за прецедентами можна вважати синонімами.

Багато методів індуктивного навчання розроблялися як альтернатива класичним статистичним підходам. Багато методів тісно пов'язані із вилученням інформації (англ. information extraction, information retrieval), інтелектуальним аналізом

Є безліч об'єктів (ситуацій) та множина можливих відповідей (відгуків, реакцій). Існує певна залежність між відповідями та об'єктами, але вона невідома. Відома лише кінцева сукупність прецедентів – пара «об'єкт, відповідь», яка називається навчальною вибіркою. На основі цих даних потрібно відновити неявну залежність, тобто побудувати алгоритм, здатний для будь-якого можливого вхідного об'єкта видати досить точну відповідь, що класифікує. Ця залежність не обов'язково виражається аналітично, і тут нейромережі реалізують принцип рішення, що емпірично формується. Важливою особливістю при цьому є здатність системи, що навчається, до узагальнення, тобто до адекватного відгуку на дані, що виходять за межі наявної навчальної вибірки. Для вимірювання точності відповідей запроваджується оцінний функціонал якості.

Ця постановка є узагальненням класичних завдань апроксимації функцій. У класичних задачах апроксимації об'єктами є дійсні числа чи вектори. У

реальних прикладних завданнях вхідні дані про об'єкти можуть бути неповними, неточними, нечисловими, різнорідними. Ці особливості призводять до великої різноманітності методів машинного навчання.

Розділ машинного навчання, з одного боку, утворився в результаті поділу науки про нейромережі на методи навчання мереж та види топологій їхньої архітектури, з іншого боку – увібрав у себе методи математичної статистики. Наведені нижче способи машинного навчання засновані на застосуванні нейромереж, хоча існують і інші методи, засновані на навчальній вибірці — наприклад, дискримінантний аналіз, що оперує узагальненою дисперсією та підступністю спостережуваної статистики, або байєсовські класифікатори. Базові види нейромереж, такі як перцептрон та багат шаровий перцептрон (а також їх модифікації), можуть навчатися як з учителем, так і без вчителя, з підкріпленням та самоорганізацією. Але деякі нейромережі та більшість статистичних методів можна віднести лише до одного із способів навчання. Тому, якщо потрібно класифікувати методи машинного навчання в залежності від способу навчання, то буде некоректним відносити нейромережі до певного виду, правильніше було б типізувати алгоритми навчання нейронних мереж.

Основні характеристики для побудови – це користувацькі дані, тому для реалізації доцільно використати методи алгоритмів випадкового лісу, дерева рішень лінійної та логістичної регресій. Кожен користувач передає певний набір даних, в залежності від завдання, кожній метриці присвоюються певні коефіцієнти, тому, використовуючи цей набір змінних, генерується функцію, яка зіставляє вхідні дані з бажаними виходами. Процес навчання триває, доки модель не досягне бажаного рівня точності даних навчання.

Кінцевим результатом роботи програми повинні бути дві графічні моделі, які будовані на основі середньозважених коефіцієнтів оцінки користувача, а також кожному користувачу буде присвоєна оцінка. Результати будуть зберігатися в файлі з розширенням «.csv».

Справжній позитивний (ТР): Справжній позитивний вимірює ступінь, до якої модель правильно передбачає позитивний клас. Тобто модель передбачає, що екземпляр позитивний, а екземпляр насправді позитивний. Справжні позитиви актуальні, коли користувач хоче знати, скільки позитивних результатів правильно прогнозує дана модель. Наприклад, у задачі двійкової класифікації з класами «А» і «В», мета полягає в тому, щоб правильно передбачити клас «А», тоді справді позитивним буде кількість випадків класу «А», які модель правильно передбачила як клас «А». Якщо взяти приклад із реального світу, якщо модель розроблена для передбачення того, чи є електронний лист спамом, справді позитивний результат відбудеться, коли модель правильно передбачить, що електронний лист є спамом. Справжній позитивний показник – це відсоток усіх екземплярів, які правильно класифіковані як належні до певного класу. Справжні позитиви важливі, оскільки вони показують, наскільки добре модель працює на позитивних прикладах. У наведеній нижче матриці плутанини (рис. 1) зі 107 фактичних

позитивних результатів 104 є правильно прогнозованими позитивними результатами. Таким чином, значення True Positive дорівнює 104.

Помилково-позитивний (FP): Помилково-позитивний результат виникає, коли модель передбачає, що екземпляр належить до класу, якого він насправді не має. Помилкові спрацьовування можуть бути проблематичними, оскільки вони можуть призвести до прийняття неправильного рішення. Наприклад, якщо модель медичного діагнозу має високий рівень хибнопозитивних результатів, це може призвести до непотрібного лікування пацієнтів. Хибні спрацьовування можуть бути шкідливими для моделей класифікації, оскільки вони знижують загальну точність моделі. Є кілька способів вимірювання хибнопозитивних результатів, зокрема показники хибнопозитивних результатів. Рівень хибнопозитивних результатів – це частка всіх негативних прикладів, які прогножуються як позитивні. Хоча хибні спрацьовування можуть здатися поганими для моделі, у деяких випадках вони можуть бути бажаними. Наприклад, у медичних програмах часто краще проявити обережність і отримати кілька помилкових спрацьовувань, ніж повністю пропустити діагноз. Однак в інших програмах, таких як фільтрація спаму, помилкові спрацьовування можуть бути дуже дорогими. Тому важливо уважно розглянути компроміси, пов'язані з вибором між різними моделями класифікації. У наведеному нижче прикладі (рис. 1) хибний позитивний результат представляє кількість негативних (із 64), які хибно визначені як позитивні. З 64 фактичних негативів 3 помилково прогнозовані як позитивні. Таким чином, значення False Positive дорівнює 3.

Істинне негативне (TN): Справжні негативні результати – це результати, які модель правильно прогнозує як негативні. Наприклад, якщо модель передбачає, чи є у людини захворювання чи ні, істинно негативним буде, коли модель передбачає, що у людини немає захворювання, і вона насправді не має хвороби. Справжні негативні показники є одним із показників, які використовуються для оцінки ефективності моделі класифікації. Загалом велика кількість справжніх негативів свідчить про те, що модель працює добре. Істинне негативне значення використовується в поєднанні з помилковим негативним, істинним позитивним і помилковим позитивним для обчислення різноманітних показників ефективності, таких як точність, точність, запам'ятовування та оцінка F1. Хоча справжнє негативне значення дає цінну інформацію про ефективність моделі класифікації, його слід інтерпретувати в контексті інших показників, щоб отримати повне уявлення про точність моделі. З 64 фактичних негативів 61 є правильно прогнозованим негативом. Таким чином, значення True Negative дорівнює 61.

Хибнонегативний (FN): Хибнонегативний результат виникає, коли модель прогнозує екземпляр як негативний, хоча він насправді позитивний. Помилкові негативні результати можуть бути дуже дорогими, особливо в галузі медицини. Наприклад, якщо скринінговий тест на рак передбачає, що у пацієнта немає раку, хоча він насправді є, це може призвести до прогресування захворювання без лікування. Помилкові негативні результати також можуть

виникнути в інших сферах, наприклад у сфері безпеки чи виявленні шахрайства. У цих випадках хибний негатив може призвести до того, що хтось отримає доступ або схвалить транзакцію, яка не повинна була бути дозволена. Хибнонегативні результати часто є більш серйозними, ніж хибнопозитивні, тому важливо враховувати їх під час оцінки продуктивності моделі класифікації. Це значення представляє кількість позитивних результатів (зі 107), які помилково вважаються негативними. Зі 107 фактичних позитивних результатів 3 помилково прогнозовані як негативні. Таким чином, значення False Negative становить

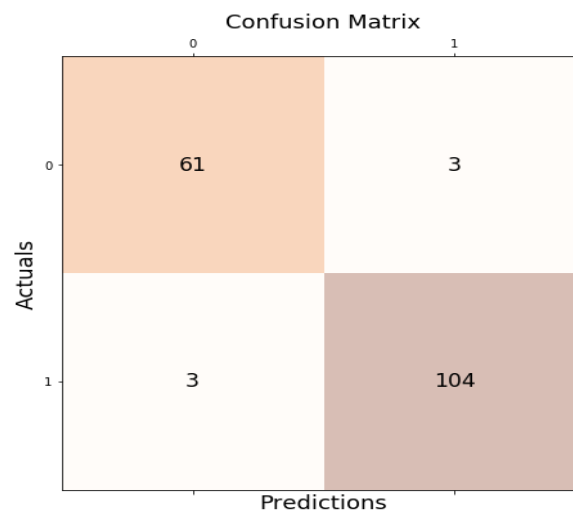


Рис.1. Матриця плутанини

Найчастіше при побудові скорингових моделей використовують мову програмування Python та її бібліотеки:

- NumPy допомагає з науковими обчисленнями;
- Matplotlib допомагає з візуалізацією;
- scikit-learn використовують для машинного навчання.

В екосистемі Python pandas є найбільш продвинутою та швидкою у роботі бібліотекою для обробки та аналізу даних. Вона побудована поверх більш низькорівневої бібліотеки NumPy (написана на C), що є великим плюсом у продуктивності.

Основні можливості бібліотеки:

- об'єкт DataFrame для маніпулювання індексованими масивами двомірних даних;
- інструменти для обміну даними між структурами пам'яті та файлами різних форматів
- вбудовані засоби розміщення даних і способи обробки відсутньої інформації;
- переформатування наборів даних, в тому числі створення зводних таблиць;
- зріз даних за значенням індексу, розширені можливості індексування, вибір із більших наборів даних;

- можливості групування дозволяють виконувати триетапні операції типу «розділення, зміна, об'єднання» (англ. split-apply-combine);
- злиття і об'єднання наборів даних;
- ієрархічне індексування дозволяє працювати з даними високої розмірності в структурах меншої розмірності;
- робота з тимчасовими рядами: формування тимчасових періодів і зміна інтервалів;

Бібліотека оптимізована для високої продуктивності, найбільш важливі частини коду написані на Cython і C.

Основне призначення додатку:

- спрощення роботи з великими об'ємами даних;
- оцінювання кожного користувача окремо;
- оптимізація роботи call-центру.

Для досягнення поставленої задачі додаток повинен уміти оброблювати великі об'єми користувацьких даних та видавати результат у графічному та текстовому вигляді.

Для підтримки стабільної роботи додатку, слід дотримуватися таким технічним вимогам:

- операційна система: Windows 10, Windows Server 2016 або Windows Server 2019;
- оперативної пам'яті 4 гігабайт;
- процесор x64 з тактовою частотою 1,4 ГГц;
- 100 мегабайт вільного місця на диску.

Наукова новизна розробки складається в удосконаленні математичної моделі процесів обробки користувацьких даних та проєктування процесу скорингу.

Висновки. В результаті було спроектовано надійний додаток для аналітики користувацьких даних. У тому числі, були розглянуті шаблони обчислення та програмні можливості Python бібліотек для побудови математичних моделей.

ПЕРЕЛІК ПОСИЛАНЬ

1. Imran Ahmad. 40 Algorithms Every Programmer Should Know: Hone your problem-solving skills by learning different algorithms and their implementation in Python, 1st Edition. - Packt Publishing, 2020. – 384с.
 2. Hong Zhou. Learn Data Mining Through Excel: A Step-by-Step Approach for Understanding Machine Learning Methods, 1st Edition. Apress, 2020. – 236с.
 3. Stephen Klosterman. Data Science Projects with Python: A case study approach to successful data science projects using Python, pandas, and scikit-learn, 1st Edition. - Packt Publishing, 2019. – 374с.
 4. Matt Harrison. Pandas 1.x Cookbook: Practical recipes for scientific computing, time series analysis, and exploratory data analysis using Python, 2nd Edition. - Packt Publishing, 2020. – 816с.
- Jaime Buelta. Python Automation Cookbook: 75 Python automation ideas for web scraping, data wrangling, and processing Excel, reports, emails, and more, 2nd Edition. - Packt Publishing, 2020. – 528с

МЕТОДИКА КАРТОГРАФУВАННЯ ПОЖЕЖ НА ЦИФРОВИХ ЗНІМКАХ З ВИКОРИСТАННЯМ МАШИННОГО НАВЧАННЯ

Анотація. Запропоновано автоматичний напівконтрольований підхід машинного навчання для картографування пожеж за допомогою багатоспектральних зображень. Проведено аналіз «традиційних» методів та на основі машинного навчання виявлення порушень покриву землі. Експериментальні результати нашого методу доводять його цінність у нагляді за лісовими пожежами.

Ключові слова: *машинне навчання, картографування, багатоспектральні зображення, пожежі.*

Вступ. Пожежа є важливим процесом у екосистемі та відіграє важливу роль при формуванні ландшафтів, наземних екосистем і атмосферного середовища. Крім того, лісові пожежі спричиняють великі людські втрати та матеріальні збитки, забруднення атмосфери, пошкодження ґрунту тощо. Існуючі дослідження показують, що орієнтовна глобальна щорічна площа горіння становить приблизно 420 мільйонів гектарів [1]. Тому, щоб зменшити негативний вплив пожежі, слід проводити виявлення активних пожеж у режимі реального часу, що може забезпечити своєчасну та цінну інформацію для управління пожежами. З розвитком технологій супутникового дистанційного зондування все більше дослідників обирають використання супутникових багатоспектральних зображень для виявлення лісових пожеж [2, 3]. Загальними ознаками пожеж є яскраве полум'я та дим, що утворюються під час горіння, а також високі температури на вогненних поверхнях, які відрізняються від навколишнього середовища. Дим і полум'я, що утворюються під час горіння, можна виявити у смугах видимого світла на зображеннях дистанційного зондування, а високі температури на поверхні пожеж легко виявити в середньому інфрачервоному, короткохвильовому інфрачервоному та тепловому інфрачервоному діапазонах.

Алгоритми машинного навчання стали життєво важливим інструментом у сучасному багатоспектральному аналізі зображень, особливо в класифікації земельного покриву, через їх безпрецедентну прогностичну силу [4, 5]. Однак точність класифікації на основі машинного навчання повністю залежить від набору даних. Таким чином, ановані набори даних стали важливою передумовою для розробки та оцінки класифікацій на основі машинного навчання.

Постановка задачі. Розробити методику виявлення пожежі з використанням згорткової нейронної мережі (Fire_CNN) на основі супутникових зображень Sentinel-2. Запропонована методика використовує багатомасштабну згортку та остаточну схему прийняття, яка може ефективно отримувати точні характеристики пожежних

точок і покращувати точність виявлення пожежі. Для реалізації методики потрібно розв'язати наступні задачі:

- провести аналіз програмних аналогів та наукової літератури у предметній галузі;
- підготувати навчальну та тестову вибірки;
- розробити нейронну мережу для розпізнавання пожежі;
- навчити та протестувати нейронну мережу;
- виконати тестування запропонованої методики на знімку супутника Sentinel-2.

Основний зміст роботи. Сучасні алгоритми машинного навчання (МН), засновані на загорткових нейронних мережах, були розроблені в середині минулого десятиліття. Проте вже накопичено значний досвід їх застосування у завданнях сегментації супутникових знімків, зокрема маскуванню хмарності [] чи класифікації типів землекористування []. Але до теперішнього часу опубліковано порівняно небагато робіт, де розглядається застосування цих алгоритмів для виявлення порушень лісового покриву []. Отже, завдання виявлення порушень лісового покриву із застосуванням цих алгоритмів має високий рівень новизни.

У роботі запропоновано методику картографування пожеж на цифрових знімках супутника Sentinel-2 на основі CNN – мереж та складається з п'яти кроків:

1. Завантажити цифрові супутникові знімки.
2. Провести попередню обробку даних з пункту 1. На цьому етапі пропонуємо використати комбінований метод на основі еквалізації, головних компонентів:

$$X = \begin{vmatrix} x_{11}, x_{21}, \dots, x_{n1} \\ \dots\dots\dots\dots\dots\dots\dots\dots \\ x_{1n}, x_{2n}, \dots, x_{nn} \end{vmatrix} \quad (1)$$

де n – спостережень, тобто. вихідний масив геоданих представлений матрицею X розмірністю $n * N$, N – може бути число профілів зйомки.

3. На третьому етапі пропонується використати кластерний аналіз з метою розбиття пожеж (згарищ) на спектрально однорідні ділянки. Заснований даний етап на властивостях вегетаційних індексів. Це дозволить виявити порушення лісового покриву. В роботі пропонується використовувати нормалізований індекс гарі (NBR – Normalized Burn Ratio) для виявлення вигорілих територій та має високу кореляцію зі ступенем пошкодження лісів пожежами [2]:

$$NBR = \frac{Band8A - Band12}{Band8A + Band12} \quad (2)$$

де Band8A та Band12 – значення спектральної яскравості пікселя (коефіцієнта спектральної яскравості), отримані в ближньому інфрачервоному та середньому інфрачервоному діапазонах супутника Sentinel-2.

Для виділення нещодавніх меж гарей пропонується використати різницю індексу dNBR (Differenced Normalized Burn Ratio):

$$dNBR = (NBR_{prefire} - NBR_{postfire}), \quad (3)$$

де $NBR_{prefire}$ – значення до пожежу; $NBR_{postfire}$ – після пожежу.

4. Розробка моделі нейронної мережі. В даній роботі запропоновано використати архітектуру згорткової нейронної мережі на рис. 1. Основною ідеєю роботи таких нейронних мереж є чергування шарів згортки та шарів підвибірки. Архітектура містить 3 згорткових шари, 3 шари підвибірки та 2 повнозв'язкові шари. Повністю зв'язаний рівень приймає об'єднані характеристики як вхідні дані, враховуючи всі особливості, і, нарешті, виводить ймовірність того, що точка вибірки є пожежною/не пожежною точкою; вибрали двійкову крос-ентропійну функцію як функцію втрат. Двійкова крос-ентропійна функція визначається таким чином [4]:

$$g(y, \hat{y}) = y \cdot \log \hat{y} + (1 - y) \log (1 - \hat{y}), \quad (4)$$

де y – прогнозоване значення, а \hat{y} – справжнє значення. Якщо точка є осередком пожежі, $\hat{y} = 1$, якщо точка не є осередком пожежі, $\hat{y} = 0$, а y є ймовірністю того, що точка є осередком пожежі, $0 \leq y \leq 1$.

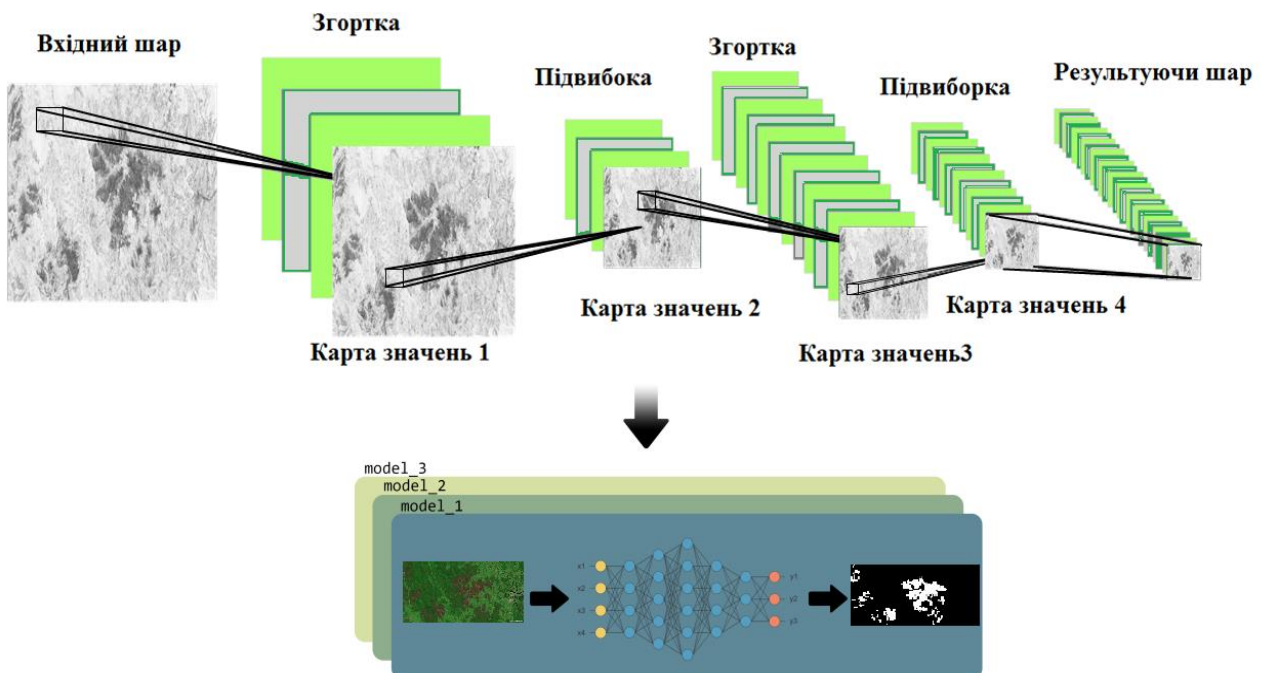


Рис. 1. Архітектура нейронної мережі пошуку пожеж

5. На завершальному етапі виділяємо зону пожежі на основі полігону. Для оперативного виділення місця пожежу необхідно провести аналіз динаміки горіння території на основі «гарячих» пікселів. На цьому етапі важливим є період отримання цифрових космічних знімків (до та після пожежу). Це дозволило створити тимчасові серії полігонів пожеж, згарищ. Алгоритм містить процедуру прийняття рішення щодо перевірки кожного нового полігону пожежі (згарища) частиною вже існуючої пожежі або необхідно додати до бази даних як нову зону пожежі. Для реалізації виділення полігону було використано критерії радіаційної температури космічних знімках (тобто інформація, яка міститься в ІЧ-канал). Математично це можна представити як [5]:

$$x_i = a \cdot y_i + b + E_i \quad (4)$$

де x_i – яскравість i -го пікселя зображення до пожежі. y_i – яскравість i -го пікселя зображення після пожежі, E_i – випадкова помилка.

Реалізація методики виконана в програмному середовищі Python. На рис. 2 представлено структурну схему реалізації методики картографування пожеж на цифрових знімках.

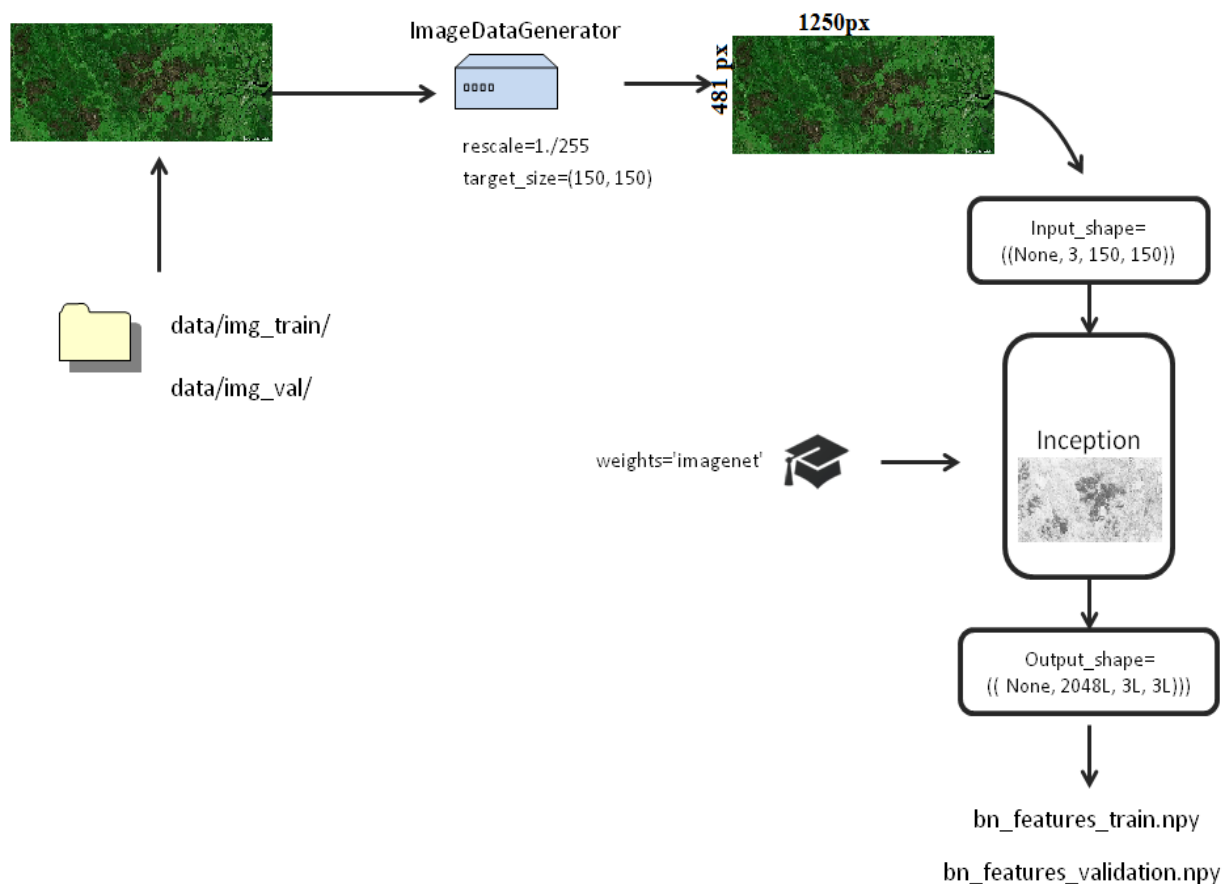


Рис. 2. Структурна схема створення нейронної мережі мовою Python



a)



б)



в)

Рис. 3. Цифрові знімки супутника Sentinel-2 Австралії регіону Coffs Harbour: а) первинні; б) після обробки; в) запропонована методика

Тестування проводились на цифрових зображеннях супутника Sentinel-2, від 03.01.2020 р. регіону Австралії. На рис. 3 представлено первинні дані, після корекції та картографування пожеж. Потім ми провели аналіз існуючих методів із запропонованим та вияснили, що не всі гарі були зафіксовані існуючими методами. Низові пожежі, як правило, погано виділяються на знімках. Матриця помилок являє собою інструмент, що використовує крос-табуляцію для показу того, як співвідносяться значення класів, що збігаються, отримані з різних

джерел. Розрахунок помилки матриці наведено в табл.1. Під час навчання мережа показала значення точності 97%, а функція втрат становила 0,092.

Таблиця 1

Точності масок, побудованих за допомогою спектральних індексів та запропонованої методики

Маска	Точність, %
NBR	70,24
dNBR	76,58
MIRBI	59,14
SWVI	29,25
Запропонована методика	93,88

Наукова новизна полягає у представленні модифікованої згорткової нейронної мережі, що містить багато масштабну згортку та остаточну прийнятну схему рішення. Це дозволить ефективно отримувати точні характеристики місця вогнищ та картографувати на цифрових зображеннях.

Висновки. У цій статті було представлено методику виявлення пожежі за допомогою модифікованої CNN (Fire_CNN). Запропонована методика використовує багатомасштабну структуру згортки, яка може враховувати взаємозв'язок між функціями різних масштабів, щоб мережа могла ефективно виділяти функції «гарячих» точок (які не є пожежними). Ці структури покращують здатність до навчання мережі та швидкість навчання. Щоб оцінити ефективність запропонованого алгоритму, його було протестовано на супутникових зображеннях Sentinel-2 регіону Австралії та показано, що запропонована методика може виявляти лісові пожежі з точністю на 29,2% вище, ніж традиційні методи.

ПЕРЕЛІК ПОСИЛАНЬ

1. Giglio, L., Boschetti, L., Roy, D. P., Humber, M. L., and Justice, C. O. (2018). The Collection 6 MODIS Burned Area Mapping Algorithm and Product. *Remote sensing Environ.* 217, 72–85. doi:10.1016/j.rse.2018.08.005
2. Kaku, K. (2019). Satellite Remote Sensing for Disaster Management Support: A Holistic and Staged Approach Based on Case Studies in Sentinel Asia. *Int. J. Disaster Risk Reduction.* 33, 417–432. doi:10.1016/j.ijdrr.2018.09.015.
3. Barmoutis, P., Papaioannou, P., Dimitropoulos, K., and Grammalidis, N. (2020). A Review on Early forest Fire Detection Systems Using Optical Remote Sensing. *Sensors.* 20 (22), 6442. doi:10.3390/s20226442.
4. Gewali, U.B.; Monteiro, S.T.; Saber, E. Machine Learning Based Hyperspectral Image Analysis: A Survey. February 2018. Available online: <http://arxiv.org/abs/1802.08701>.
5. Gewali, U.B.; Monteiro, S.T.; Saber, E. Machine Learning Based Hyperspectral Image Analysis: A Survey. February 2018. Available online: <http://arxiv.org/abs/1802.08701>.

КЛАСИФІКАЦІЯ СИГНАЛІВ ОБ'ЄКТІВ КЕРУВАННЯ З ВИКОРИСТАННЯМ ПЕРЕБУДОВАНИХ МАТРИЧНИХ СПЕКТРАЛЬНИХ ОПЕРАТОРІВ

Анотація. В роботі пропонується критерій класифікації сигналу, що супроводжує функціонування об'єкта керування, за якістю розкладання випадкового вектора в базисах, що перебудовуються, які відповідають безлічі класів випадкових сигналів.

Ключові слова: випадкові сигнали, класифікація, перебудовані оператори.

Вступ. При контролі параметрів об'єкта управління за вектором вихідних даних необхідно вирішити такі задачі [1]:

- подання вихідних даних як результату вимірювань будь-якої характеристики об'єкта;
- виділення характерних (інформативних) ознак та зниження розмірності вектора вихідних даних;
- знаходження вирішальних процедур, необхідних при класифікації сигналу з метою контролю параметрів об'єктів управління та підвищення ймовірності правильного вибору зразка з кодової книги.

Постановка задачі. Нехай стиску підлягають M дискретних реалізацій, кожна з яких належить одному з класів m випадкового процесу за умови наявності апріорних відомостей про клас процесів у вигляді еталона. До таких процесів відносяться періодично-корельовані випадкові процеси з періодичним повторенням статистичних характеристик (математичні очікування, дисперсії, автокореляційні функції). Ця модель досить адекватно визначає процеси, що супроводжують функціонування роторних об'єктів управління. Нехай \mathbf{X}^m - вектор розміром $N, m = \overline{1, M}$. Вважаємо, що вектори належать N -мірному евклідовому простору. и $\|\mathbf{X}^m\| = 1$. Вважаємо також, що для кожного m відома класифікована (навчальна) вибірка $\{\mathbf{X}_l^m\}, l = \overline{1, K}$, де K – обсяг вибірки. Тоді незміщеною обґрунтованою оцінкою є середня за безліччю класифікованої вибірки

$$\mathbf{X}_{\text{sr}}^m = \frac{1}{K} \sum_{l=1}^K \mathbf{X}_l^m . \quad (1)$$

Зазначимо, що спектральний оператор субоптимальний по Карунену-Лоеву, що перебудовується [2], пристосований до еталону класу m тому сенсі, що у спектральній області перебудованого базису еталону зіставляється лише один коефіцієнт, відмінний від нуля. Базис, що перебудовується, є

оптимальним для еталона в сенсі прийнятого критерію інформативності спектральних ознак. За процедурою побудови базису, що перебудовується $\mathbf{X}_{\text{ст}}$ є перший рядок матриці спектрального оператора. Позначимо матрицю спектрального оператора, що перебудовується, для класу m через $\mathbf{V}^m = [\mathbf{V}_1^m, \mathbf{V}_2^m, \dots, \mathbf{V}_N^m]^T$. Для рядків \mathbf{V}_i^m матриці \mathbf{V}^m виконуються умови ортонормованості.

Для формування вектора інформативних ознак за вектором вихідних даних \mathbf{X} пропонується сформувати систему спектральних ознак $\{\mathbf{Y}^m\}$, $m = \overline{1, M}$:

$$\mathbf{Y}^m = \frac{1}{N} \mathbf{V}^m \mathbf{X} . \quad (2)$$

Кожен вектор \mathbf{Y}^m являє собою вектор спектральних коефіцієнтів по системі ортогональних функцій, що перебудовуються для класу m по системі \mathbf{V}^m ортогональних функцій. У запропонованому методі формування вектора інформативних ознак паралельно формуються M систем інформативних ознак (M спектрів в базисах, що перебудовуються) по N ознак (N спектральних коефіцієнтів) у кожній системі.

Компоненти y_i^m вектора \mathbf{Y}^m визначаються таким чином:

$$y_i^m = \frac{1}{N} \sum_{n=1}^N B_{i_n}^m x_n , \quad (3)$$

де $B_{i_n}^m$ - компоненти рядка \mathbf{V}_i^m матриці \mathbf{V}^m .

Щоб оцінити якість розкладання випадкового вектора в базисі, що перебудовується, використовуємо точність опису вектора частиною ортогональних функцій. Для цього обчислюється сума квадратів коефіцієнтів розкладання за вказаними функціями:

$$A_p^m(\mathbf{Y}^m) = \sum_{i=1}^p (y_i^m)^2 . \quad (4)$$

Величина $A_p^m(\mathbf{Y}^m)$ визначається як квадрат норми проекції вектора \mathbf{Y}^m , принадлежащего класу m на підпространство, визначаєме як

$$\{\mathbf{V}_1^m, \mathbf{V}_2^m, \mathbf{V}_3^m, \dots, \mathbf{V}_p^m\} , p < N$$

Кожна величина y_i^m , згідно з теоремою Парсеваля, є потужністю, що припадає на ортогональну функцію з номером i . Співвідношення (4) визначає потужність перших p спектральних складових розкладання (2).

Якщо вектор розкладається в базисі того класу m , до якого належить сам, то в середньому за вибіркою $\{\mathbf{X}_l^m\}, l = \overline{1, K}$ значення квадрата проєкції на перші p ортогональних функцій більше, ніж те значення для вектора, що належить іншому класу s . Значення квадратів проєкцій на перші p ортогональних функцій тим більше відрізнятимуться для різних векторів, чим сильніше відрізняються між собою класи, до яких належать ці вектори.

Таким чином при класифікації спектральних ознак, отриманих за допомогою безлічі базисів, що перебудовуються, кожен з яких побудований для конкретного класу сигналів, можна визначити критерій по максимуму норми проєкції вектора на підпростір спектральних ознак, за яким випадкові вектори \mathbf{X} відносяться до класу m :

$$\begin{aligned} \text{якщо } A^{ms} = A_p^m - A_p^s > 0 \\ \text{для всіх } s = \overline{1, M} (s \neq m), \text{ то } \mathbf{X} \in \{\mathbf{X}^m\}. \end{aligned} \quad (5)$$

Наукова новизна складається в розробці критерію класифікації векторів за якістю розкладання випадкового вектора в базисах, що перебудовуються, відповідних безлічі класів випадкових сигналів.

Висновок. Розроблено критерії класифікації сигналу (5), що супроводжує функціонування об'єкта за спектрами в базисах, що перебудовуються, з метою підвищення точності контролю параметрів об'єкта керування. Критерії класифікації випадкових векторів розроблені виходячи із способу оцінювання якості розкладання векторів, що класифікуються, за першими ортогональними функціями перебудованих базисів. Вирішальні правила, що використовують розроблені критерії, дозволяють забезпечити кращу роздільність в ознаковому просторі порівняно з загальноприйнятим при класифікації векторів критерієм мінімуму евклідова відстані від еталонного представника класу векторів.

ПЕРЕЛІК ПОСИЛАНЬ

1. Power Engineering, Control and Information Technologies in Geotechnical Systems / [під ред. Genadiy Pivnyak, Oleksandr Beshta, Mykhaylo Alekseyev]. - Leiden: CRC Press/Balkema, 2015. - 220 с.
2. Energy Efficiency Improvement of Geotechnical Systems: International Forum on Energy Efficiency / [під ред. Genadiy Pivnyak, Oleksandr Beshta, Mykhaylo Alekseyev]. - Leiden : CRC Press/Balkema, 2013. - 244 с.

АНАЛІЗ ФУНКЦІОНАЛЬНИХ МОЖЛИВОСТЕЙ ПАКЕТІВ ПРИКЛАДНИХ ПРОГРАМ МОДЕЛЮВАННЯ БЕЗДРОТОВИХ СЕНСОРНИХ МЕРЕЖ

Анотація. На сьогоднішній день методи комп'ютерного та імітаційного моделювання є досить ефективним і дієвим інструментом у дослідницькій та інженерній діяльності, що обумовлює актуальність і доцільність їх використання під час розробки прикладних інформаційних технологій. Основна мета статті полягає в критичному аналізі та логічному узагальненні відомих наукових і прикладних результатів щодо методів і програмно-технічних засобів моделювання бездротових сенсорних мереж. Основні результати роботи: проаналізовано існуючі середовища комп'ютерного моделювання та їх характерні особливості; встановлено основні критерії щодо порівняльної характеристики популярних моделюючих середовищ; описано та доведено основні характерні особливості, а також переваги і недоліки середовищ моделювання.

Ключові слова: бездротова мережа, середовище моделювання, Suprcarbon, Matlab, Op-net.

Вступ. Бездротова сенсорна мережа реалізується мобільними пристроями, які взаємодіють між собою засобами радіоканалів. Проведення експериментальних досліджень із реальними пристроями та технологіями не завжди є можливим, отже, методи комп'ютерного моделювання і симуляції являють собою реальну і доцільну альтернативу.

Серед множини важливих властивостей і параметрів, які мають бути визначені та прийняті до обліку під час імітаційних комп'ютерних експериментів, основними є: модель мобільності та модель розповсюдження радіохвиль. Обидва зазначені параметри мають значний вплив на продуктивність мобільних бездротових мереж, наприклад, продуктивність протоколів маршрутизації залежить від цих моделей. На сьогоднішній день у науково-аналітичних джерелах запропоновано та досліджено широкий спектр моделей мобільності та поширення радіохвиль. Проте варто зазначити, що більшість таких моделей була розроблена для конкретних прикладних цілей і не може бути однозначно адаптована і використана для кожного фізичного сценарію [1].

Моделі розповсюдження радіохвиль, які використовуються в симуляторах звичайних бездротових мереж, зазвичай базуються на простих моделях розповсюдження радіохвиль і не враховують наявності перешкод у середовищі розповсюдження [2].

Постановка задачі. Основна мета статті полягає в критичному аналізі та логічному узагальненні відомих наукових і прикладних результатів щодо методів і програмно-технічних засобів моделювання бездротових сенсорних

мереж. Для досягнення поставленої мети в роботі сформовані і вирішені такі завдання:

- проаналізувати існуючі середовища моделювання та їх характерні особливості;
- встановити основні критерії щодо порівняльної характеристики моделюючих середовищ;
- описати та довести основні характерні особливості, а також переваги і недоліки середовищ моделювання.

Основний зміст роботи. Під час досліджень цієї статті було проаналізовано наступні популярні в світовій науковій спільноті середовища моделювання, які можуть бути використані для симуляції бездротових сенсорних мереж:

- CupCarbon;
- MATLAB;
- OP-NET.

CupCarbon – це багатоагентний симулятор бездротової мережі датчиків із дискретними подіями, який базується на засадах геолокації. Це дозволяє моделювати та імітувати сенсорні мережі в цифровому географічному інтерфейсі OpenStreetMap. Для цієї мети CupCarbon надає набір простих у використанні та конфігурованих об'єктів. CupCarbon складається з трьох основних компонентів: мультиагентне середовище моделювання, засоби мобільного моделювання та вбудований симулятор WSN (WiSen).

CupCarbon реалізує середовище для цілей багатоагентного моделювання. Використання багатоагентної системи дозволяє відтворити середовище, яке ефективно імітує фізичні системи реального світу, де кожен об'єкт у системі працює автономно, обмінюючись інформацією з іншими об'єктами в тому самому середовищі. Інтеграція OpenStreetMap до середовища CupCarbon забезпечує інтерфейс і базу даних оцифрованих даних, що пов'язані зі задачами геолокації. Серед основних агентів у середовищі моделювання CupCarbon можна знайти бездротові сенсорні вузли датчиків і базових станцій для надсилання та отримання пакетів даних. Також є моделі генераторів аналогових подій, мобільних об'єктів та інше. Приклад моделі інфокомунікаційної мережі у середовищі CupCarbon представлено на рис. 1.

CupCarbon пропонує можливість створювати шляхи, вільні або пов'язані з реальними дорогами, та призначати їм агентів, щоб зробити їх мобільними. Кожен шлях визначається набором опорних точок на карті OSM, а кожна точка пов'язана з датою, що відповідає точній даті прибуття об'єкта до неї. Із міркувань оптимізації пам'яті до оперативної пам'яті завантажуються лише траєкторії, які пов'язані з об'єктами, що інтегровані до процесу симуляції [3].

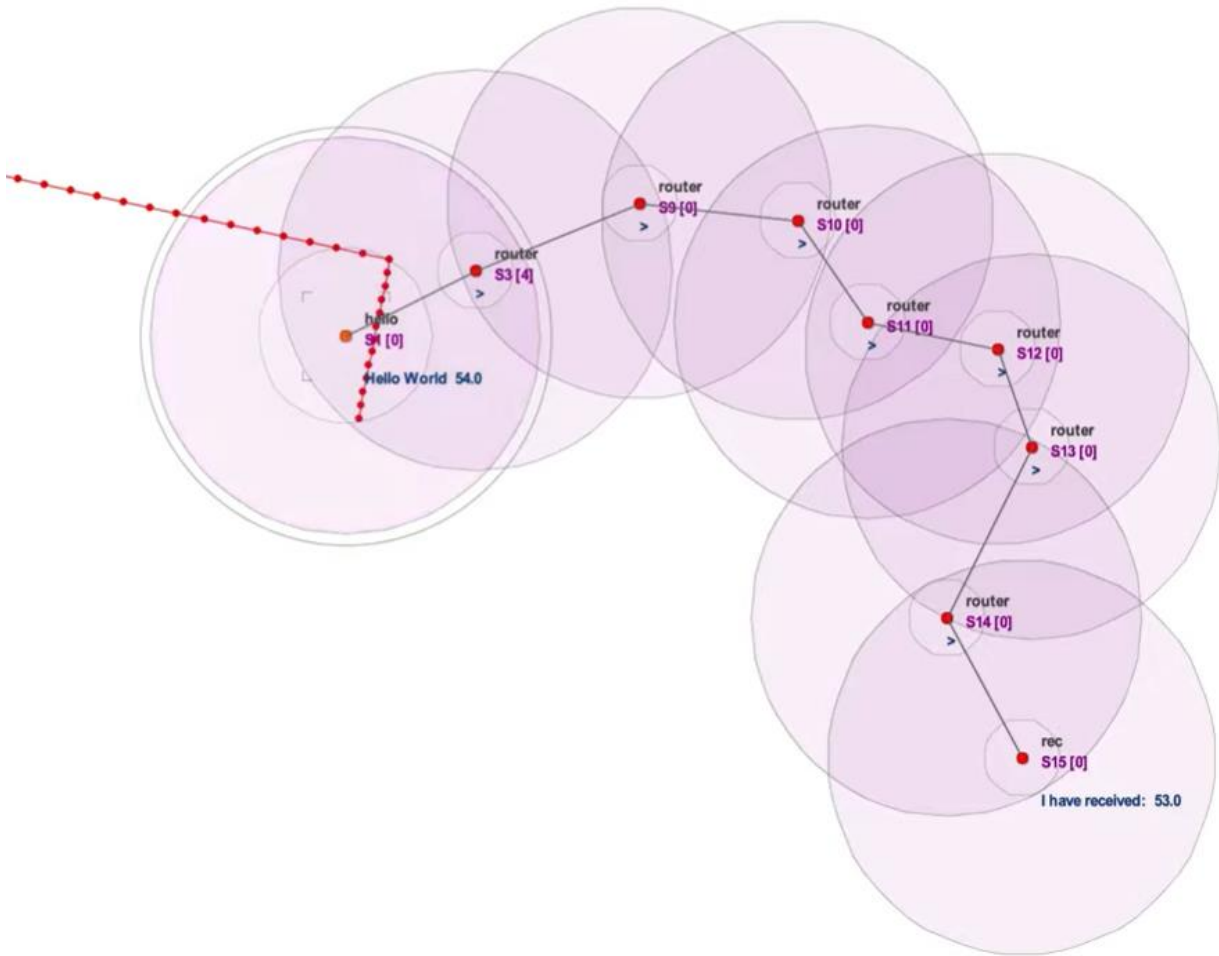


Рис. 1. Приклад типової моделі мережі у CupCarbon

Пакет прикладних програм Matlab є найбільш універсальним та продуктивним програмним забезпеченням для створення та моделювання широкого спектру технологічних рішень, у тому числі, бездротових сенсорних мереж. Matlab забезпечує природне представлення сигналів і має широкий спектр інтегрованого функціоналу для обробки сигналів. Основною особливістю є широкий функціонал можливостей, до яких входить [4]:

- розробка та оцінка методологій для перевірки дотримання стандартів на кожному етапі проектування;
- комбінація багатьох компонентів і моделювання на початкових етапах науково-дослідної діяльності;
- модельне проектування, що може бути використане в якості стратегії для скорочення циклів проектування;
- наявний додаток для бездротового моніторингу та прототип зі стандартними програмно-визначеними радіосистемами (SDR).

Приклад моделі мережі в середовищі Matlab наведений на рисунку 2.

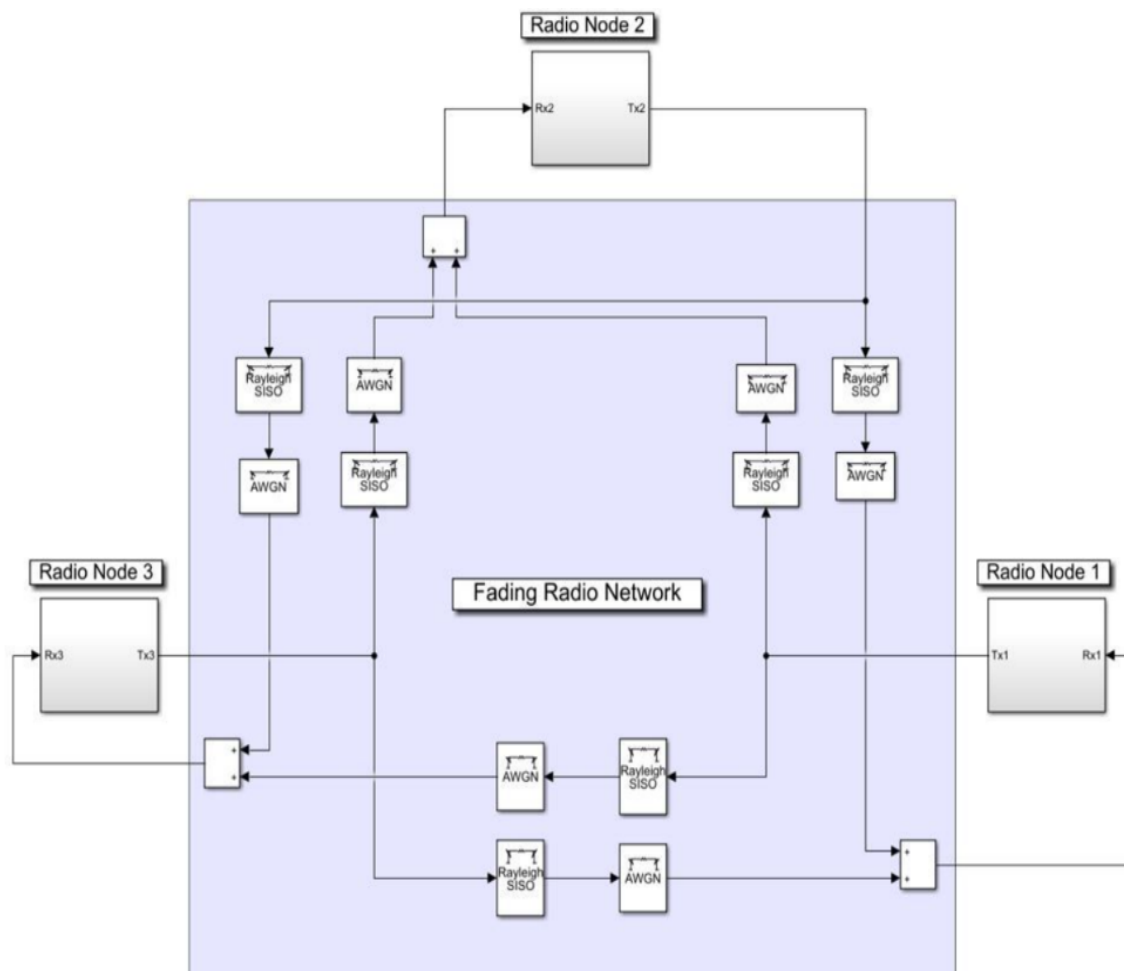


Рис. 2. Приклад моделі в пакеті прикладних програм Matlab [5]

Зазначені вище особливості поліпшують процес моделювання комплексних рішень та виконання глибокої декомпозиції мережевої системи. Типовими базовими моделями елементів вузлів мережевих структур є:

- моделювання на рівні мережі за допомогою радіочастот;
- цифрова смуга частот;
- цифро-аналоговий перетворювач;
- RF-симулятор;
- можливість приймання / передавання сигналів із використанням антени.

Пакет прикладних програм OP-NET використовує метод об'єктного моделювання та графічний редактор, що забезпечує комплексне середовище розробки під час моделювання мережі зв'язку та моделювання розподілених систем. За рахунок використання основного механізму, керованого подіями, у порівнянні з процесом, що керується часом, обчислювальна ефективність була значно покращена. OP-NET забезпечує три механізми моделювання: нижній – модель процесу, що реалізує алгоритм узгодження; середній – модель вузла, яка використовує модель процесу для реалізації відповідної функції обладнання; верхній – модель мережі, яка використовує модель вузла для побудови структури топології реальної мережі. Тривірневі моделі повністю відповідають

фактичному протоколу та обладнанню, які повністю відображають особливості мережі. Приклад моделі мережі в середовищі Matlab наведений на рис. 3.

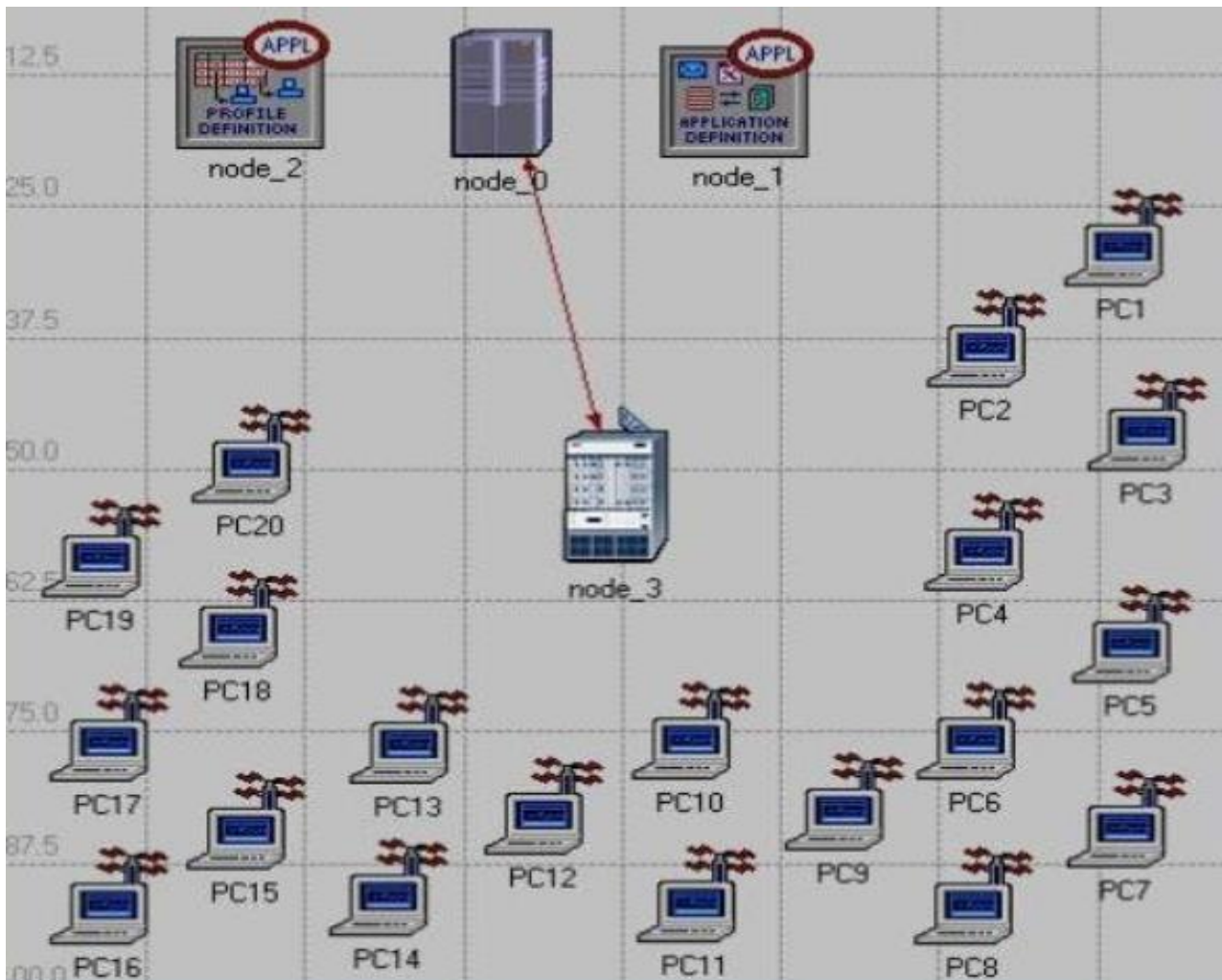


Рис. 3. Приклад мережі у середовищі OP-NET [6]

Відповідно до результатів моделювання однією з основних задач є аналіз продуктивності за багатьма параметрами, включаючи різні режими та умови навантаження. На основі моделювання є можливим отримання відповідних рішень для аналізу та покращення продуктивності мережі.

Зазначені вище середовища дозволяють провести комп'ютерне моделювання систем бездротового зв'язку. При цьому важливим є проведення порівняльної характеристики, як це наведено в таблиці 1.

Виходячи з аналізу інформації, яку наведено в таблиці 1, було зроблено висновок, що Matlab є найбільш оптимальним середовищем, із точки зору, побудови комплексних систем, що потребують високий рівень наближеності до реальних умов експерименту та під час вирішення завдань, що потребують декомпозиції системи. На базі середовища Matlab можливим є проектування різноманітних бездротових рішень із подальшим тестуванням та моніторингом. У, свою чергу, CupCarbon є оптимальним для мереж датчиків, mesh-мереж та

систем рухомого зв'язку, що дозволяє досить швидко провести експеримент та отримати основні показники вузлів системи. OP-NET найбільш вдало підходить для вирішення задач моделювання статичних мереж комп'ютерного сегменту.

Таблиця 1.

Порівняльна характеристика середовищ моделювання бездротових мереж

Характеристика середовища моделювання	CupCarbon	MATLAB	OPNET
Особливості побудови системи	побудова на базі геолокації	створення моделі у вигляді блоків	імітація мережі на логічному рівні
Налаштування вузлів	автономне	загальне	загальне
Ступінь декомпозиції	середній	високий	низький
Базова направленість	мережі датчиків	загальні системи передачі	комп'ютерні мережі
Можливості моніторингу	процес маршрутизації, енергетичні рівні	усі можливі параметри	навантаження системи
Ступінь інтуїтивності системи	високий	низький	високий
Ступінь інтеграції з іншими системами	низький	високий	низький
Ступінь універсальності	середній	високий	низький

Наукова новизна отриманих результатів полягає у встановленні критеріїв та систематизації показників щодо аналізу середовищ моделювання бездротових систем, що дає змогу оптимізувати вибір підходу під час моделювання конкретних типів систем із обліком критеріїв експерименту, що є важливим етапом проектування технічних систем.

Висновки. Проведено аналіз середовищ комп'ютерного моделювання для організації бездротової передачі інформації. У результаті отримано критеріальні оцінки кожного з середовищ (CupCarbon, Matlab і Op-net) відповідно до зазначених особливостей та виявлено найбільш оптимальні сценарії використання для кожного випадку зі зазначенням відповідних характерних особливостей.

ПЕРЕЛІК ПОСИЛАНЬ

1. What-is-iot-the-internet-of-things. URL: <https://www.networkworld.com/article/3207535/what-is-iot-the-internet-of-things> (дата звернення: 14.11.2022).
2. Iot-architecture-topology-and-edge-compute. URL: <https://www.digi.com/blog/post/iot-architecture-topology-and-edge-compute> (дата звернення: 15.11.2022).
3. Saha S., Acharjee U., Tahzib-Ul-Islam Md. A Survey on Wireless Mesh Network and its Challenges at the Transport Layer. Int. Journal of Computer Engineering and Technology, 2014. Vol. 5 (8). P. 169 – 177.
4. Wireless Communications. MathWorks. URL: <https://www.mathworks.com/solutions/wireless-communications/standards.html> (дата звернення: 16.11.2022).

5. Wireless Transceiver Design and Network Modeling in Simulink. URL: <https://www.mathworks.com/company/newsletters/articles/wireless-transceiver-design-and-network-modeling-in-simulink.html> (дата звернення: 14.11.2022).

6. INTEGRATING OPNET PROJECTS IN WIRELESS MESH NETWORK. URL: <https://opnetprojects.com/opnet-wireless/> (дата звернення: 15.11.2022).

УДК 330.46

М.А. Демиденко¹, О.Ю. Чуріканова¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

МОДЕЛЬ ОПТИМІЗАЦІЇ МАРКЕТИНГОВИХ КОМПАНІЙ В ПОШУКОВИХ СИСТЕМАХ

Анотація. У статті пропонуються теоретичні та практичні рекомендації щодо вирішення економічної проблеми ведення науково-обґрунтованої маркетингової компанії за допомогою інструментів реклами Google з використанням математичного моделювання. Теоретичною і методологічною основою дослідження є фундаментальні положення сучасної економічної теорії, наукові праці вчених. У процесі дослідження використовувалися такі методи: економіко-математичне моделювання; графічний; аналітичні та структурно-логічний. Пропонується комплексний підхід до пошуку такої рекламної компанії, яка в умовах невизначеності, використовуючи моделювання, задовольняє вимогам клієнта і забезпечує брокеру мінімальний ризик отримання прибутку. На підставі проведених досліджень запропоновано економіко-математичну модель для визначення набору ключових слів, за допомогою яких брокер може утримувати прибуткову рекламну компанію з мінімальними ризиками.

Ключові слова: метод, модель, оптимізація, критерій, Google Ads, ключові слова, цифрова економіка, контекстна реклама, пошукова оптимізація, реклама.

Вступ. В сучасній цифровій економіці маркетинг в пошукових системах, є одним з найбільш поширених рекламних технологій. В пошуковиках демонстрація рекламних оголошень відбувається разом із результатами запитів які співпадають з ключовими словами оголошень. За сценарієм роботи пошукових машин реклама з'являється на сторінках, повернутих після пошукових запитів. Рекламодавці сплачують пошуковим системам оплату за системою "плата за клік" (Pay-per-click, PPC). Кожен рекламодавець має обліковий запис в системі, в якому для продукту що рекламується сформовано портфель ключових слів, з відповідними ставками за клік на словах які ініціюють показ їх оголошень та загальний щоденний бюджет.

Формування прибуткового портфеля може виявитися складним для багатьох рекламодавців, головним чином через брак часу, ресурсів або досвіду на рекламних ринках. Тому створення таких портфелів доручається стороннім організаціям-реklamним агентствам та брокерам. Взаємодія між сторонніми партнерами та рекламодавцями створює вторинний ринок пошукового маркетингу (на відміну від первинного ринку, на якому рекламодавці

безпосередньо взаємодіють з пошуковими системами), де треті сторони продають послуги з оптимізації рекламних кампаній, а рекламодавці - виступають як покупці цих послуг. Тому доцільно розробити економіко-математичну модель оптимізації рекламної кампанії на вторинному ринку. В роботі представлено модель яка дозволяє мінімізувати ризики брокера і отримати запланований прибуток.

Аналіз останніх досліджень і публікацій. Вирішенням проблем оптимізації рекламних компаній в Інтернеті займалась низка науковців. В роботі [1] запропоновано методика оптимізації ключових слів. В роботі [2] була розроблена модель для коефіцієнта конверсії в оголошеннях Google. Автори [3] запропонували метод оптимізації оголошень Google який використовував семантичні знання. Автори [4] вивчали взаємозв'язок між характеристиками оголошень Google, позицією реклами та рейтингом пошукової системи Google. У [5] і [6] була розроблена багатозадачна рекомендаційна система, яка показала, що специфічні текстові структури можуть призвести до високого рейтингу кліків на рекламі в пошуковому маркетингу. Інші роботи [7] проаналізували поведінку користувачів в пошуковому маркетингу і запропонували механізми максимізації доходу

Проте більшість досліджень авторів спрямована на формування портфеля ключових слів в пошуковому маркетингу, і ніхто не аналізує ефективність отриманого портфеля і методику його формування.

Постановка задачі. Метою даної роботи є дослідження теоретичних основ та прикладних проблем оптимізації пошукового маркетингу що зумовлює постановку таких цілей:

- розробити економіко-математичну модель, яка дозволяє знайти оптимальний варіант впровадження пошукового маркетингу який забезпечує бажану прибутковість при мінімальних ризиках збитків для брокера із задовольняє вимогам замовника.

- розробити математичну модель пошукового маркетингу, яку можливо реалізувати в прикладному програмному забезпеченні.

Основний зміст роботи. В сучасні цифровій економіці проведення маркетингу через оголошення Google є особливо ефективним, оскільки пошукова система Google обробляє великий обсяг трафіку і, таким чином, забезпечує найбільшу кількість показів рекламних оголошень.

Оголошення Google (Google Ads) є найпопулярнішою в світі системою реклами. Платформа Google Ads дозволяє компаніям створювати оголошення, які відображаються в пошуковій системі Google та інших продуктах Google.

Google Ads працюють за моделлю "плата за клік" (Pay-per-click, PPC), за якою користувачі встановлюють ставки за ключовими словами та сплачують за кожен клік на рекламі. Клік у контексті пошукової оптимізації означає перехід користувача на сайт за посиланням, розміщеної на сторінці пошуковика або сторонньому сайті. Щоразу, коли розпочинається пошук, Google виконує пошук у сукупності оголошень рекламодавців і вибирає набір оголошень-переможців, які відобразатимуться на перших місцях на сторінці результатів

пошуку. "Переможці" вибираються за рейтингом на основі комбінації факторів, включаючи якість і релевантність ключових слів і рекламних кампаній, а також розмір ставок для ключових слів.

Рейтинг або позиція на сторінці пошуку розраховується за формулою

$$\text{Рейтинг реклами} = \text{Максимальна ставка} \times \text{Показник якості} \times \text{Очікуваний вплив додаткової інформації (розширень)} \quad (1)$$

Максимальна ставка це найбільша ціна, яку рекламодавець згідний платити за клік на ключовому слові. Показник якості це значення, яке розраховує Google з урахуванням якості цільової сторінки і відповідності ключових слів цій сторінці. Очікуваний вплив розширень оголошень - це величина яка враховує ефект від застосування додаткової інформації до оголошення, наприклад номер телефону або інші посилання на певні сторінки сайту.

Ця система дозволяє рекламодавцям залучати потенційних клієнтів за ціною, яка відповідає їхньому бюджету. Це по суті своєрідний аукціон.

Розглянемо функціонування вторинного ринку Google Ads, який працює за наступною схемою. На ринку взаємодіють клієнти і брокери. Клієнти доручають брокерам провести рекламну компанію їх продукту. На практиці клієнт довіряє сторонньому брокеру побудувати портфель ключових слів Google Ads для ефективною реклами в пошуковиках. Клієнт сплачує брокеру плату за послуги. Брокер взаємодіє безпосередньо з Google Ads, встановлюючи максимальний розмір ставок, враховуючи загальний бюджет, і сплачує витрати на показ оголошень Google. Прибуток брокера – це різниця між платою за його послуги та витратами на рекламу, сплаченими Google. Різні рекламодавці можуть мати різні цілі для своїх рекламних кампаній, тому брокер повинен, узгоджувати свою мету прибутковості з потребами його клієнта. Як правило, між рекламодавцем і брокером можливі два типи договорів:

1. Рекламодавець має мету $D(a)$ для свого оголошення і брокер пропонує загальний бюджет досягнення цієї мети за визначений час;
2. Рекламодавець має загальний бюджет для своєї компанії, і брокер зобов'язується досягти цільової мети $D(a)$ для цього бюджету за узгоджений з клієнтом проміжок часу.

Незалежно від типу контракту, ефективність брокера зводиться до мінімізації витрат, сплачених Google. Вартість оплати Google Ads залежить від ціни за клік (CPC), яка визначається через механізми онлайн-аукціону (1).

Тому є доцільним розробити економіко-математичну модель для визначення сукупності ключових слів за допомогою, яких брокер може провести найбільш прибуткову рекламну компанію.

Нехай рекламуємо товар X , з яким у Google Ads пов'язані ключові слова $i = 1, 2, \dots, N$, які розшукувались в Google у наступні проміжки часу $t = 1, 2, \dots, T$.

Брокер пропонує свої послуги за ціною P_{it} за клік на i -му слові у проміжок часу t . У свою чергу Google стягує плату C_{it} за кліки на ключових словах $i=1,2, \dots, N$ у проміжки часу $t=1,2, \dots, T$.

Тоді прибуток брокера становитиме

$$\Pi_t = \sum_{i=1}^N \sum_{t=1}^T (P_{it} - C_{it}). \quad (2)$$

Середня вартість кліка на ключовому слові $\overline{C_{it}}$ в рекламній компанії продукту a в пошуковикі Google становитиме

$$\overline{C_{it}} = \frac{\sum_{i=1}^N \sum_{t=1}^T C_{it}}{S}, \quad (3)$$

де S – загальна кількість кліків на $i=1,2, \dots, N$ ключових словах за час $t=1,2, \dots, T$ проведення рекламній компанії продукту a .

Тоді прибутковість рекламної компанії продукту, а для брокера обчислиться за формулою

$$R(a) = \frac{\overline{P_{it}} - \overline{C_{it}}}{\overline{C_{it}}}, \quad (4)$$

де $\overline{P_{it}}$ – середня вартість кліка за ставками послуг брокера, яку можна розрахувати за формулою

$$\overline{P_{it}} = \frac{\sum_{i=1}^N \sum_{t=1}^T P_{it}}{S}, \quad (5)$$

Брокер має сформувати такий набір ключових слів, який забезпечить заданий рівень прибутку при мінімальному ризику отримати збитки.

Тоді критерій оптимізації рекламної компанії a для брокера можна сформулювати як мінімізацію ризику того, що прибуток брокера буде нижчим, ніж ціна яку він сплачує Google за кліки на ключових словах. Розрахунок критерію виконується за формулою

$$\sigma_a = \sqrt{\frac{\sum_{i=1}^N \sum_{t=1}^T (P_{it} - C_{it})^2}{\tau \mu}} \rightarrow \min, \quad (6)$$

де τ – загальна кількість проміжків часу на яких виконувався пошук в рекламній компанії; μ – загальна кількість ключових слів.

Ключові слова $i = 1, 2, \dots, N$ в рекламній компанії мають різну відповідність рекламуємому продукту a і різну запитуваність у пошуках в Google, тому доцільно присвоїти їм різні коефіцієнти ваги. Тоді ω_i коефіцієнт ваги i -го ключового слова. При цьому виконуються умови

$$0 \leq \omega_i \leq 1,$$

$$\sum_{i=1}^N \omega_i = 1.$$

На практиці коефіцієнти ваги ω_i є випадковими величинами які змінюються в часі.

Тоді модель оптимальної рекламної компанії брокера можна сформулювати таким чином:

$$\sqrt{\frac{\sum_{i=1}^N \sum_{t=1}^T (P_t(\omega_i) - C_t(\omega_i))^2}{\tau \mu}} \rightarrow \min$$

$$\left. \begin{array}{l} \sum_{i=1}^N \sum_{t=1}^T (P_t(\omega_i) - C_t(\omega_i)) = \rho \\ \sum_{i=1}^N \omega_i = 1 \end{array} \right\} \quad (7)$$

$$\omega_i \geq 0$$

де ρ – прибуток брокера, за проведення рекламної компанії.

Модель (7) може розглядатися як різновид моделі Марковіца, яка відрізняється від відомої моделі тим, що вагові коефіцієнти є випадковими величинами, а ризик оцінюється середньоквадратичним відхиленням прибутку (збитку) брокера. Практичне розв'язання моделі може бути здійснено із застосуванням відомих методів [6, 7]

Наукова новизна. Розроблено модель формування портфеля ключових слів реклами, яка забезпечує оптимальний компроміс між прибутковістю та ризиком брокера, в умовах невизначеності поточної плати за клік і кількості кліків у маркетинговому пошуку. Отримані результати дослідження дозволять

учасникам пошукового маркетингу підвищити прибутковість інвестицій у рекламну кампанію, запропонована модель може бути реалізована в додатках для комп'ютерів та мобільних пристроїв для автоматизації процесу оптимізації пошукового маркетингу.

Висновки. Сформульована проблема формування портфелю ключових слів пошукового маркетингу і необхідності розробки моделі для підвищення прибутковості інвестицій у рекламну кампанію. Обґрунтовано критерій оптимізації рекламної кампанії для брокера на вторинному ринку пошукового маркетингу. На основі проведених досліджень запропоновано економіко-математичну модель для визначення сукупності ключових слів за допомогою, яких брокер може провести прибуткову рекламну компанію з мінімальними ризиками. Модель може бути застосована для замовника, якщо його розглядати як брокера його рекламних оголошень.

ПЕРЕЛІК ПОСИЛАНЬ

1. Amruta J., Rajeev M. Keyword generation for search engine advertising. *Sixth IEEE International Conference. Data Mining Workshops* : works of Sci.-Tech. conf., New York, IEEE, 2016. ICDM Workshops 2016. New York, P. 490–496.
2. Anindya G. and Sha Y. An empirical analysis of search engine advertising: Sponsored search in electronic markets. *Management Science*. 2009. № 55(10). P.1605–1622,
3. Thomaidou S., Vazirgiannis M. Multiword keyword recommendation system for online advertising. In *Advances in Social Networks Analysis and Mining*: works of Sci.-Tech. conf., New York, IEEE 2011 ASONAM 2011. New York, P. 423–427.
4. Demydenko M. Method of selection of ERP systems using multi-criterial optimization models. *Naukovyi Visnyk NHU*. 2018. №5. С. 132–137.
5. Demydenko M. Economic-mathematical model of optimal pricing of suppliers and processors of raw materials for the sustainable development of the mining and metallurgical complex of Ukraine. *International scientific conference Anti-crisis management: state, region, enterprise* : works of Sci.-Tech. conf., Le Mans, 17 November 2017. Le Mans Université, France, 2017. №3. С. 140–143.
6. Демиденко М. А. Системи підтримки прийняття рішень : монографія. Дніпропетровськ: Національний гірничий університет, 2016, 106 с.
7. Демиденко М. А. Управління проектами інформатизації за методологією SCRUM. : монографія. Дніпропетровськ: Національний гірничий університет, 80 с

В.Ю. Каштан¹, К.Л. Сергєєва¹, О.В. Коробко¹, Д.В. Іванов¹, А.Р. Кучер¹
¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

МЕТОДИКА АВТОМАТИЗОВАНОГО РЕГУЛЮВАННЯ ДОРОЖНЬОГО РУХУ НА ОСНОВІ НЕЙРОННОЇ МЕРЕЖІ

Анотація. Робота присвячена актуальній задачі розробці методу регулювання дорожнього руху на основі нейронної мережі, що підвищить повноту та достовірність інформації, точність розрахунків та пропорційний розвиток транспортних систем, заснованих на репрезентативних даних про параметри транспортного потоку.

Ключові слова: *дорожній рух, YOLO-мережа, інтенсивність руху, щільність трафіку.*

Вступ. Інтенсивність руху є головним показником при визначенні рівня завантаженості автомобільних доріг. При вивченні інтенсивності руху визначають такий параметр, як нерівномірність транспортного потоку – його розподіл за часом і напрямками. Щодня величезна кількість машин стоїть у пробках та паралізує рух у місті. Часом час, загублений у пробках, сягає кількох годин, що виливається на шкоду економічній, психологічній та екологічній.

Для контролю та управління рухом транспорту нині встановлюються автоматизовані системи управління дорожнього руху, що дозволяє централізовано керувати мережею дорожніх контролерів (світлофорів) за допомогою пристроїв зв'язку з дорожніми контролерами.

Постановка задачі. Розробити методику керування дорожнім рухом на основі нейронної мережі. Ця нова система полегшує рух автомобілів на перехрестях, що призводить до зменшення заторів, менших викидів CO₂ тощо.

Для досягнення цієї мети необхідно вирішити наступні завдання:

- виконати аналіз проблематики методів моніторингу транспортного потоку у великих містах;
- обґрунтувати теоретичні передумови дослідження транспортного потоку на основі нейронної мережі під час проектування вулично-дорожньої мережі;
- виконати експериментальні дослідження щодо визначення інтенсивності руху як основного параметра транспортних потоків для проектування нових транспортних об'єктів та реконструкції існуючих вулиць та доріг.

Основний зміст роботи. Завдання апроксимації невідомої функції з відомим набором даних вхідних і очікуваних вихідних даних може бути вирішено шляхом застосування нейронних мереж [1]. Ідея навчання штучної нейронної мережі полягає в тому, щоб надати набір даних із вхідними та очікуваними виходами, а також використовувати алгоритм для встановлення

ваги кожного вхідного нейрона таким чином, щоб помилка цього набору даних була мінімальною.

Алгоритм регулювання дорожнього руху на основі нейронної мережі наведено на рис. 1 та складається з таких етапів: отримати зображення кожної смуги в реальному часі; сканувати та визначити щільність трафіку; ввести вищевказані дані в модуль розподілу часу; результатом будуть часові інтервали для кожної смуги відповідно.

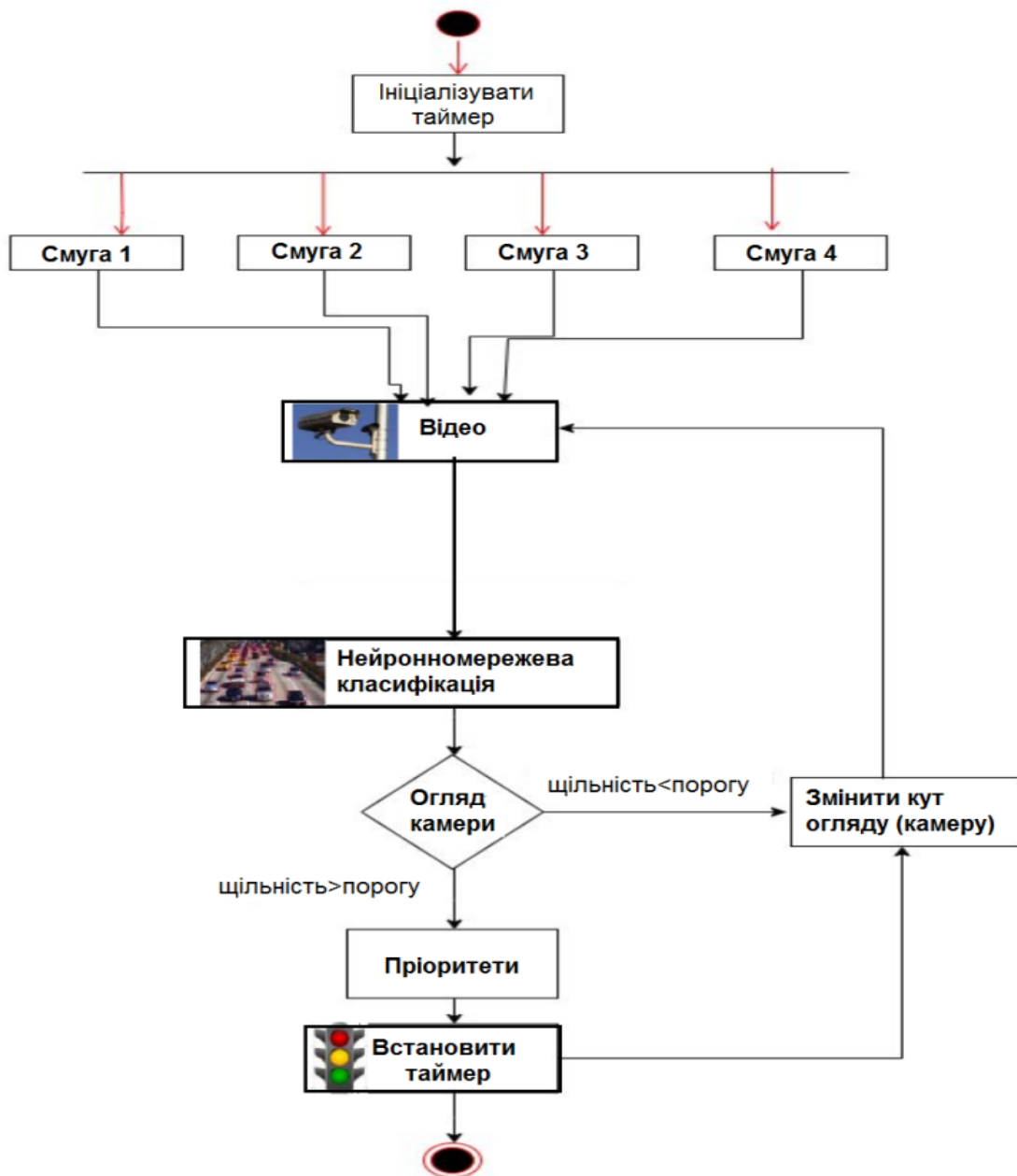


Рис. 1. Алгоритм регулювання дорожнього руху

Кожному світлофору на перехресті, що розглядається, поставимо у відповідність нейрон і об'єднаємо всі світлофори перехрестя в нейронну мережу. Особливістю нейронної мережі є те, що їй належить вирішувати

комбіновану задачу оптимізації та кластеризації. Тому мережа, що проектується, складається з двох частин. Перша частина нейронної мережі вирішуватиме задачу оптимізації, її виходи будуть входами для другої частини мережі, яка призначена для вирішення задачі кластеризації.

Першим етапом побудови нейромережових моделей є відбір вхідних даних, які впливають на очікуваний результат. Тому, вхідними даними для проектованої нейронної мережі буде оцінка дорожньої ситуації дорожнього перехрестя, що залежить від завантаженості кожного з напрямків руху в даний час, а також статистики про щільність потоків руху на перехресті у певний час доби, у певний день тижня [2]. Так як, для вихідних даних мережі необхідно отримати номер стану перехрестя, в якому він повинен знаходитись в наступний момент часу, тому пропонуємо використати багат шарову нейронну мережу [1, 2], що складається з трьох шарів. Вхідний шар: кількість нейронів цього шару залежить від кількості світлофорів. Для перехрестя вхідний шар містить 12 нейронів. Функціями цього шару є отримання даних, посилення та поширення сигналу.

Основною функцією середнього шару є знаходження рішення задачі оптимізації руху транспортних засобів через перехрестя. Вхідний та середній шари, а також матриця з вагами між ними становитимуть першу частину всієї нейронної мережі. На виходах необхідно отримати рішення мережі рівень активності кожного світлофора, тобто рішення про те, які світлофори мають бути увімкнені. Вихідний (результуючий) шар є другою складовою нейронної мережі. На виходах цього шару необхідно отримати розв'язання задачі класифікації, ці самі виходи і будуть виходами всієї мережі загалом. Кількість нейронів вихідного шару залежить від кількості станів, в які може переключитися перехрестя. Для перехрестя вихідний шар буде містити 4 нейрони, значення одного з яких буде дорівнює 1 (номер цього нейрона визначить номер обраного стану перехрестя), а значення інших – 0.

Запропонована нами система отримує зображення з камер відеоспостереження на транспортних розв'язках як вхідні дані для розрахунку інтенсивності руху в реальному часі на основі нейромережової обробки даних. Цю систему можна розділити на 3 модулі: модуль виявлення транспортних засобів, алгоритм перемикання сигналів і модуль моделювання. В модулі виявлення транспортного засобу використовуємо YOLO мережу [3]. Це найсучасніша система виявлення об'єктів у реальному часі. Попередня робота з виявлення об'єктів перепрофілює класифікатори для виконання розпізнавання транспортного засобу. Одна нейронна мережа прогнозує обмежувальні прямокутники та ймовірності класу безпосередньо з повних зображень в одній оцінці. Далі визначається кількість транспортних засобів кожного класу, таких як автомобіль, велосипед, автобус і вантажівка, для розрахунку щільності руху. Алгоритм перемикання сигналів використовує цю щільність, щоб встановити таймер зеленого сигналу для кожної смуги.

При реалізації алгоритму враховувались наступні фактори:

1. Час обробки алгоритму розрахунку щільності трафіку і потім зелений тривалість світла – це визначає, у який час потрібно отримати зображення

2. Кількість смуг.

3. Загальна кількість транспортних засобів кожного класу, таких як автомобілі, вантажівки, мотоцикли тощо.

4. Щільність руху, розрахована з використанням наведених вище факторів

5. Час, доданий через затримку, яку страждає кожен транспортний засіб під час запуску, і нелінійне збільшення затримки, яку зазнають транспортні засоби, які знаходяться позаду.

6. Середня швидкість кожного класу транспортного засобу, коли починається зелене світло, тобто середній час, необхідний для перетину сигналу кожним класом транспортного засобу [4].

7. Мінімальний і максимальний ліміт тривалості зеленого світла – для запобігання затор.

Коли алгоритм запускається вперше, час за замовчуванням встановлюється для першого сигналу першого циклу, а час для всіх інших сигналів першого циклу та всіх сигналів наступних циклів встановлюється алгоритмом. Запускається окремий потік, який обробляє виявлення транспортних засобів для кожного напрямку, а основний потік обробляє таймер поточного сигналу. Коли таймер зеленого світла поточного сигналу (або таймер червоного світла наступного зеленого сигналу) досягає 0 секунд, потоки виявлення роблять знімок наступного напрямку. Потім результат аналізується та встановлюється таймер наступного зеленого сигналу. Все це відбувається у фоновому режимі, поки основний потік відраховує таймер поточного зеленого сигналу. Це дозволяє призначати таймер без проблем і, отже, запобігає будь-якій затримці. Коли зелений таймер поточного сигналу стає нульовим, наступний сигнал стає зелений протягом часу, встановленого алгоритмом.

Зображення буде знято, коли сигнал, який має стати зеленим, становить 0 секунд. Це дає системі загалом 5 секунд (що дорівнює значенню таймера жовтого сигналу) для обробки зображення, визначення кількості транспортних засобів кожного класу, присутніх на зображенні, обчислення часу зеленого сигналу та відповідно встановлення часу цього сигнал, а також час червоного сигналу наступного сигналу. Щоб знайти оптимальний час зеленого сигналу на основі кількості транспортних засобів кожного класу на сигналі, були використані середні швидкості транспортних засобів під час запуску та час їх прискорення, на основі яких оцінюється середній час, який потрібен кожному класу транспортного засобу для перетину перехрестя знайдено. Потім час зеленого сигналу розраховується за формулою [4]:

$$GST = \frac{\sum (вiдсутнiсть\ транспортних\ засобiв \cdot середнiй\ час)}{(Lane + 1)} \quad (1)$$

де *GST* – зелений сигнальний час; відсутність транспортних засобів – це кількість транспортних засобів кожного класу транспортного засобу на сигнал, визначений модулем виявлення автомобіля; середній час, потрібний транспортним засобам цього класу для перетину перетин, і *Lanes* – кількість смуг на перехресті.

Середній час, потрібний кожному класу транспортного засобу для перетину перехрестя, можна встановити відповідно до місця розташування, тобто залежно від регіону, міста, місцевості або навіть перехрестя на основі характеристик перехрестя, щоб збільшити рух транспорту. ефективніше управління. Для цього можна проаналізувати дані відповідних транспортних органів.

Наукова новизна полягає у представленні методу для визначення інтенсивності руху з метою підвищення ефективності функціонування транспортних потоків на основі YOLO-мережі.

Висновки. В даній роботі було представлено методику керування дорожнім рухом на основі нейронної мережі. Спочатку камера надсилає зображення через регулярні короткі проміжки часу. Далі методика визначає кількість автомобілів на смузі, а отже обчислює їх відносну щільність відносно інших смуг. Модуль розподілу часу приймає дані (як щільність трафіку) від цієї системи та визначає оптимізований та ефективний часовий інтервал. Потім мікропроцесор передає це значення на відповідний світлофор. Таким чином, запропонована методика полегшує рух автомобілів на перехрестях, що призводить до зменшення заторів, менших викидів CO₂.

ПЕРЕЛІК ПОСИЛАНЬ

1. Li, Z.-X.; Qin, B.; Wang, M.-Q. Real-time Detection and Recognition of Traffic Signal Lights Based on Time-space Relationship Model. *Comput. Sci.* 2018, 45, 314–319.
2. Wang, Z.-Y.; Miao, D.-Q.; Zhao, C.-R.; Luo, S.; Wei, Z.-H. Pedestrian tracking and detection combined algorithm based on multi-granularity features. *Comput. Res. Dev.* 2020, 57, 996–1002.
3. K. Zaatouri and T. Ezzedine, "A Self-Adaptive Traffic Light Control System Based on YOLO," 2018 International Conference on Internet of Things, Embedded Systems and Communications (IINTEC), Hammamet, Tunisia, 2018, pp. 16-19.
4. Redmon J. You only look once: Unified, realtime object detection / J. Redmon, R. Girshick, A. Farhadi // The IEEE Conference on Computer Vision and Pattern Recognition, 2016. — P. 779788

НАВЧАННЯ АДАПТИВНОЇ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ СЕГМЕНТАЦІЇ ЗОБРАЖЕНЬ

Анотація. Нейронні мережі показують високу якість роботи в задачах комп'ютерного зору та, зокрема, в задачі сегментації зображень. Виникає потреба виконання мережі, як на потужних, так і на слабких пристроях. Однак, зміна часу виконання мережі передбачає зміну архітектури та перенавчання мережі, що є довгим та коштовним процесом. Для вирішення цієї проблеми в роботі запропоновано нову адаптивну після навчання архітектуру нейронної мережі, що підвищує якість сегментації та дозволяє ефективно адаптувати мережу до пристроїв із різною обчислювальною потужністю.

Ключові слова: адаптивні нейронні мережі, сегментація зображень, мобільні обчислення.

Вступ. Найефективніше задачу сегментації семантичного зображення вирішують згорткові нейронні мережі. Загалом, розв'язання проблеми потребує великої кількості обчислень. Особливо це помітно на мобільних пристроях. Такі пристрої мають обмежені обчислювальні ресурси та зазвичай працюють від акумулятора. У [1] було реалізовано кілька алгоритмів пошуку об'єктів для роботи на мобільних пристроях. Ці алгоритми включають як статистичні, так і нейро-мережеві. У таблиці 1 статті [1] наведено результати експерименту, де порівнювали продуктивність 2 пристроїв високого класу. Один із пристроїв був випущений у 2013 році, а інший – у 2018. Як виявилось, старі пристрої можуть бути від 2,13 до 5,64 разів повільнішими за новіші. Фактична відносна різниця залежить від розглянутого алгоритму.

Ця величезна різниця в продуктивності є проблематичною, особливо враховуючи той факт, що архітектуру нейронної мережі (і, як наслідок, час передбачення) неможливо змінити динамічно без повторного навчання нейронної мережі. Однак, на практиці необхідно розробляти застосунки для обох категорій пристроїв. Отже, метою є розробка адаптивної після навчання мережі для вирішення задачі сегментації.

Постановка задачі. Для досягнення поставленої мети в роботі сформовані і вирішені такі завдання:

- розроблено власну реалізацію алгоритму нейро-мережевої сегментації зображень U-Net;
- розроблено модифікацію підходу адаптивних після навчання нейронних мереж (РТА), що є застосовним для мережі U-Net та може ефективно вирішувати задачу сегментації зображень;
- експериментально підтверджено ефективність розробленого підходу із врахуванням як якості вирішення задачі, так і часу, необхідного для передбачення маски сегментації.

Основний зміст роботи. Кілька останніх робіт пропонують модифікації процедури навчання нейронної мережі, щоб результуюча мережа стала адаптивною [2, 3]. Зокрема, в [2] вперше було запропоновано підхід навчання та побудови адаптивної після навчання мережі (РТА), який дозволяє реалізувати динамічну нейронну мережу. В оригінальній роботі [2] запропоновану мережу було застосовано до проблеми проти анти-спуфінгу обличчя. Згідно до авторів, підхід РТА надає кілька переваг: 1) після завершення навчання мережі можна миттєво налаштувати час виконання нейронної мережі. На основі конфігурації час виконання становить від 80% до 107% часу роботи нединамічної нейронної мережі; 2) загальний час навчання зменшився на 14%; 3) точність прогнозування зросла з 96,74% для підходу без РТА до 97,85% для підходу РТА.

Проте запропонований підхід не застосовувався до завдання семантичної сегментації. У цій роботі побудовано нейронну мережу U-Net+РТА, яка здатна вирішити проблему сегментації зображення, а її конфігурацію можна динамічно змінювати після завершення навчання нейронної мережі.

Наш підхід базується на широко використовуваній моделі U-Net, яка показала хороші результати в задачі сегментації зображення. Подібно до [2], ми замінюємо 2 пари обернених залишкових блоків на блоки РТА. Кожен блок РТА має легкі та важкі гілки, які мають різну кількість операцій для обчислення. Легка гілка в 2 рази швидша за важку. Перемикання між гілками дозволяє змінювати час виконання всієї мережі. Щоб обидві гілки давали значущі прогнози, застосовано стратегію навчання із випадковою вибіркою конфігурацій РТА.

Застосування підходу РТА до задачі сегментації дало нам покращення як у якості роботи мережі, так і в часі виконання. Отриманий час виконання мережі можна адаптивно змінювати залежно від використовуваних потреб, а саме від 96,86% до 102,98% відносно часу виконання оригінальної мережі U-Net. Результуюча якість прогнозування мережі зросла з 85,82% для оригінальної моделі U-Net до 86,70% для нової запропонованої моделі U-Net+РТА. Обидва покращення були досягнуті без збільшення загального часу навчання.

Наукова новизна полягає у запропонованій новій адаптивній після навчання архітектурі нейронної мережі для вирішення задачі сегментації зображень. Запропонована мережа покращує якість сегментації, дозволяє динамічно змінювати архітектуру мережі та адаптувати її до пристроїв із різними обчислювальними можливостями.

Висновки. На відміну від існуючих згорткових нейронних мереж, підход РТА дозволяє динамічно змінювати швидкість і архітектуру нейронної мережі. Навчання моделі нейронної мережі займає багато часу. За допомогою адаптивності нейронної мережі після завершення навчання, можна додатково налаштувати нейронну мережу, не витрачаючи більше часу на навчання. Крім того, моделі на основі РТА можуть ефективно орієнтуватися на пристрої з різними обчислювальними ресурсами: від повільних пристроїв до більш

потужних, і все в одній моделі. У цій роботі застосовано підхід РТА до завдання семантичної сегментації. Наш підхід базується на архітектурі U-Net, яку модифіковано за допомогою блоків та стратегії навчання РТА. Наші експериментальні результати показують, що окрім динамічного налаштування архітектури, нова запропонована мережа також краща за якість, ніж оригінальна U-Net.

ПЕРЕЛІК ПОСИЛАНЬ

1. Khabarлак К. Face detection on mobile: Five implementations and analysis / К. Khabarлак // CoRR. – 2022. – Vol. abs/2205.05572.
2. Khabarлак К. Post-Train Adaptive MobileNet for Fast Anti-Spoofing / К. Khabarлак // CEUR Workshop Proceedings – CEUR-WS.org, 2022. – Vol. 3156. – P. 44-53.
3. Khabarлак К. Faster optimization-based meta-learning adaptation phase / К. Khabarлак // Radio Electronics, Computer Science, Control. – 2022. – № 1. – P. 82-92.

УДК 681.5.011(075.8)

В.М. Куваєв¹, О.О. Бойко¹, Є.К. Воскобойник¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ВІДТВОРЕННЯ СИНТЕЗОВАНОЇ МОДЕЛІ СИСТЕМИ КЕРУВАННЯ НА БАЗІ ПРОМИСЛОВОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Анотація. У роботі розглянуте питання розробки програмного забезпечення промислових системи керування на мові програмування Structured Text виходячи з їх моделей та перевірки правильності його функціонування.

Ключові слова: промислові системи, моделі функціонування, програмне забезпечення, мова програмування Structured Text.

Вступ. За останні десятиліття для вирішення питань синтезу систем керування у світі була створена велика кількість спеціалізованого програмного забезпечення. Основним напрямком розвитку даних продуктів є підвищення кількості запропонованих структур, типів регуляторів та варіантів розрахунків їх параметрів, впровадження технологій нечіткої логіки та нейронних мереж. Однак більшість цих розробок не дають можливості прямого переносу отриманих таким чином моделей до промислових систем, що зменшує їх значущість.

Окрім того при розробці програмного забезпечення сучасних промислових систем здебільшого використовуються мови програмування стандарту МЕК 61131-3. Їх аналіз показує, що для переносу отриманих моделей найбільш підходить мова програмування Structured Text, яка є схожою за можливостями з мовою програмування Pascal. Таким чином при прямому переносі синтезованих систем з мов програмування високого рівня, що

використовують об'єктно орієнтований підхід до промислових систем, що використовують структурно орієнтований підхід виникає велика кількість додаткових завдань. Використання системного підходу до вирішення таких завдань дозволить спростити процес переносу синтезованих моделей, зменшити кількість помилок у програмному забезпеченні та спростити процес його відладки.

Основний зміст роботи. У Національному технічному університеті «Дніпровська політехніка» в якості програмного забезпечення синтезу та моделювання систем керування використовується математичний пакет MATLAB, а в якості промислового програмного забезпечення SCADA система zenon Supervisor. Кожний з цих продуктів займає більше 30 % світового ринку у своїй галузі таким чином їх можна розглядати в якості типових прикладів. Однією з переваг SCADA системи zenon є наявність у її складі програмного програмованого логічного контролера zenon Logic, мови програмування якого відповідає стандарту МЕК 61131-3, а сам контролер може використовуватися при симуляції роботи апаратного забезпечення системи керування.

Здебільшого синтез систем керування з використанням математичного пакету MATLAB проходить у графічному середовищі імітаційного моделювання Simulink. При цьому використовуються безперервні моделі типових передавальних ланок, арифметичні та алгебраїчні операції, нелінійні елементи. У деяких випадках розробляються користувацькі моделі на мові програмування MATLAB. Таким чином синтезована модель системи керування складається з великої кількості елементів, які необхідно перенести до програмного програмованого логічного контролера zenon Logic для отримання відповідного програмного забезпечення.

Враховуючи вище зазначене нами пропонується розглядати синтезовану модель систему керування як набір з трьох структурних елементів: моделі об'єкта керування, моделі регулятора та моделі збурення.

Модель об'єкта керування у більшості випадків складається з: безперервних передавальних функцій, які замінюються відповідними цифровими передавальними функціями, а їх параметри розраховуються за z-перетворенням, затримок, які реалізуються як черги FIFO, нелінійностей, які задаються за допомогою умовних операторів та арифметичних операцій. Таким чином для отримання тексту програми моделі об'єкта керування у математичному пакеті MATLAB необхідно виконати заміну всіх безперервних функцій на цифрові та згенерувати код на мові Structured Text. Нажаль існують випадки коли генератор не може виконати поставлене завдання, тоді текст програми формується для окремих елементів відповідної моделі, після чого на його підставі розробляється загальне програмне забезпечення моделі. Враховуючи це рекомендується давати для всіх блоків моделі та їх входів і виходів зрозумілі назви. Створення програмного забезпечення моделі у zenon Logic Workbench виконується на підставі отриманого тексту програми у вигляді User Defined Function Block який реалізує об'єктно орієнтований підхід.

Якщо модель регулятора розроблена на мові програмування MATLAB у неї виділяються вхідні, вихідні та внутрішні змінні. На підставі отриманого опису створюється відповідний User Defined Function Block у якому самостійно відтворюється алгоритм роботи регулятора за наявним кодом. Це пов'язано з тим, що мова Structured Text є жорстко типізованою, а крім того присутня не відповідність між її алгоритмічними функціями та функціями мови MATLAB.

Реалізація моделі збурення залежить від форми його завдання. При використанні генератора псевдовипадкових значень бажано записати їх до файлу та використовувати у процесі перевірки функціонування отриманого програмного забезпечення.

На підставі отриманих функціональних блоків створюється загальне програмне забезпечення синтезованої системи керування на мові програмування Function Block Diagram, яка наочно відображає структуру отриманої моделі.

Для розробленого програмного забезпечення синтезованої системи керування створюється людино машинний інтерфейс який записує результати роботи системи керування у текстовий файл. Отримані результати порівнюється з результатами моделювання у математичному пакеті MATLAB на підставі чого робиться висновок про можливість використання отриманого регулятора в якості елемента системи керування.

Висновки. Використання розглянутого підходу забезпечує відтворення програмне забезпечення синтезованої моделі системи керування на базі промислового програмного забезпечення та дозволяє перевіряти правильність його функціонування відповідно до похідної моделі.

ПЕРЕЛІК ПОСИЛАНЬ

1. Бойко О.О., Проектування автоматизованих систем керування технологічними комплексами. Конспект лекцій для студентів спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології». / О.О. Бойко, Є.К. Воскобойник; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Електрон. Текст. Дані – Дніпро : НТУ «ДП», 2020. – 99 с.

2. Бойко О.О., Комп'ютерна логіка. Конспект лекцій для студентів напряму підготовки 151 «Автоматизація та комп'ютерно-інтегровані технології». [Електронний ресурс] / О.О. Бойко, Є.К. Воскобойник; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Електрон. Текст. Дані – Дніпро : НТУ «ДП», 2020. – 48 с.

3. Методичні рекомендації до практичних робіт по курсу «Системи диспетчерського управління та збору даних SCADA» для здобувачів першого рівня освіти (бакалаврського) спеціальності 122 «Комп'ютерні науки»/ Проценко С.Н., Бойко О.А., Куваєв В.М., Голінько О.В.; М-во освіти і науки України, Нац. техн. ун-т «Дніпровська політехніка». – Дніпро : НТУ «ДП», – 2022. – 179 с

РОЗДІЛ 5

ІНФОРМАЦІЙНА БЕЗПЕКА

УДК 004.056:004.94

В.І. Корнієнко¹, Б.О. Статива¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ВДОСКОНАЛЕННЯ МЕТОДУ ЗАХИСТУ ХМАРНИХ ДАНИХ ІЗ КОНТРОЛЕМ ЛІНІЙ ЗВ'ЯЗКУ ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ СИСТЕМ

Анотація. Розглянуто вдосконалення методу антивірусного захисту даних за рахунок безпечної маршрутизації метаданих у антивірусні хмарні системи. Основними складовими рішеннями методу є алгоритми формування безлічі маршрутів передачі метаданих, контролю ліній зв'язку та нейромережевої безпечної маршрутизації. Проведено розрахунки для підтвердження достовірності одержаних результатів та представлена структурна схема алгоритму формування базової множини маршрутів передачі метаданих.

Ключові слова: інформаційно-комунікаційна система (ІКС), захист інформації, лінії зв'язку, антивірус, хмарні дані

Вступ. Стрімкий розвиток інформаційно-комунікаційних систем і мереж (ІКС) та інформаційних технологій викликає ряд проблем, пов'язаних з безпекою мережевих ресурсів. Тому актуальною є розробка та вдосконалення систем, методів та засобів антивірусного захисту даних і мінімізації наслідків кіберзлочинів, своєчасне виявлення і локалізація комп'ютерних вірусів є вкрай важливим і разом з тим складним завданням.

Тому мета роботи: дослідження та вдосконалення методу антивірусного захисту даних в ІКС з використанням сучасних хмарних обчислювальних систем.

Постановка задачі. Аналіз процесу функціонування ІКС, а також дослідження процесів формування, передачі та обробки метаданих у хмарних антивірусних системах за дослідженнями авторів праць [1-4], дозволили визначити щільність розподілу ймовірностей часу передачі хешфайлу метаданих у хмарні антивірусні системи, а також обробки та доставки команд передачі управління, сформувані та математично формалізувати знання про зміни та характер поведінки основних імовірно-часових показників якості обслуговування в ІКС.

Для формування множини маршрутів потрібно ефективного виконання основних процесів, що реалізують такі алгоритми:

1. Алгоритм пошуку найкоротших шляхів між вузлами в ІКС.
2. Алгоритм формування базової множини маршрутів передачі даних.

3. Алгоритм безпечної маршрутизації на базовій безлічі шляхів передачі метаданих у програмний сервер.

Основний зміст роботи. Для алгоритму пошуку найкоротших шляхів між вузлами були проведені дослідження, що показали розв'язання цієї задачі у площині вирішення загального завдання маршрутизації метаданих у хмарні антивірусні системи. Тому є необхідність вибору базового алгоритму пошуку найкоротших шляхів.

З аналізу відомих алгоритмів пошуку найкоротших шляхів був обраний один з найоперативніших алгоритмів, що задовольняв заданим вимогам ($O(n^2^n)$) – алгоритм D'Esopo-Pape. Даний алгоритм рекомендується в результатах досліджень ряду авторів, а також підкріплені результатами експериментів, що проводились за допомогою імітаційної моделі [4].

Результати досліджень імітаційного моделювання підтверджують достовірність результатів аналізу алгоритмів пошуку найкоротших шляхів. Отже, таким чином, можна відзначити доцільність використання даного алгоритму (D'Esopo-Pape), як базовий при пошуку найкоротших шляхів між вузлами в ІКС.

Наступним кроком є алгоритм формування базової множини маршрутів передачі метаданих, для знаходження множини маршрутів, виключаючи «петлі». При вирішенні даної задачі пошуку найкоротших шляхів з використання даного алгоритму у більшості випадків є можливість зустрітися з проблемою «зациклювання» даних у знайдених шляхах, що називається «петлями», і це призводить до збільшення часу, а іноді й втрати, передачі інформаційних пакетів. Позбутися «петель» можна за допомогою додавання часових обмежень.

Після формування базової множини маршрутів передачі метаданих виникає необхідність проводити постійний моніторинг каналів зв'язку та адаптивно змінювати таблиці базової множини маршрутів у разі аномальних змін у показниках тестових сигналів. Отже, саме для вирішення цього завдання призначений алгоритм безпечної маршрутизації на базовій множині шляхів передачі метаданих у програмний сервер

З проведених досліджень попередніх підрозділів з'ясовано, що виникає необхідність у знаходженні такої множини маршрутів, використання якої в умовах обмежень, що накладаються, дозволили б забезпечити максимальну можливу інформаційну безпеку, а саме у моніторингу каналів зв'язку та виборі з всієї знайденої множини шляхів оптимальної сукупності маршрутів.

Основні вимоги до якості наданих послуг в ІКС задаються в параметричному вигляді, системою обмежень на допустиму ймовірність спотворення інформаційних пакетів у процесі передачі та допустиму ймовірність прийому інформаційного пакету за визначений час.

Головною особливістю розглянутого алгоритму є можливість постійного моніторингу та облік каналів зв'язку ІКС на маршрутах вузлів програмного сервера, що дають можливість виявляти можливості кібератак та несанкціонованого доступу до ІКС.

Для того щоб маршрутизатор міг функціонувати, необхідно сформувати інформацію про стан з'єднань, що виходять з даного вузла. Кожному з'єднанню надається певний вектор параметрів, компоненти якого характеризують певну складову фізичної сполуки.

Слід зауважити, що одним із найважливіших параметрів, які необхідно враховувати при виборі подальшого шляху маршрутизації інформації, є тип каналу зв'язку, його пропускна здатність, швидкість передачі та функціональна безпека.

Витік інформації може бути здійснено шляхом прямого приєднання до каналу зв'язку та зчитування її за допомогою технічних засобів. Внаслідок цього має бути можливість визначення спроб підключення до каналу.

Щоб система системи нейромережових експертів безпечної маршрутизації нормально функціонувала необхідно підготувати та систематизувати дані, за якими проводиться навчання його окремих нейромережових компонентів. Для вирішення цього завдання частина цієї системи формування навчальної та тестової вибірки формує дані для навчання нейронної мережі, впорядковує та організує її з метою забезпечення можливості їх подальшої обробки за допомогою нейромережових технологій.

Для нормального функціонування системи необхідно сформувати їх початкові стани, виражені попередньою установкою вагових коефіцієнтів нейронних мереж. Як було зазначено раніше для вирішення поставленого завдання доцільно використати імунний алгоритм оптимізації, побудований на основі принципів імунітету живих організмів.

Отже, в результаті, для кожного нейромережового експерта створюється окремий комплекс популяцій антитіл, усередині кожної популяції проводиться розвиток антитіл, мутація та видалення.

На основі досліджень та оцінки визначено, що у порівнянні з методом одноколіїної маршрутизації застосування антивірусного захисту даних у ІКС за рахунок багатознакової маршрутизації метаданих у хмарні антивірусні системи скорочує час передачі інформаційних пакетів метаданих у 1,9-3 рази. А це дозволить скоротити час передачі інформаційних пакетів до 15 разів.

На базі проведених досліджень проведено імітаційне моделювання передачі сигнатур в ІКС з наступними параметрами:

1) використовуються аналізатори ліній зв'язку та інтелектуальні засоби прийняття рішень про компрометації маршрутів.

2) можливість обслуговувати в одиницю часу таку кількість пакетів, яка відповідає кількості пакетів, що зберігаються в одному блоку пам'яті буфера.

3) довжина інформаційного пакету $l_p = 1024$ біт.

4) кількість експериментів $N^* = 100$.

На основі цих даних шляхом імітаційного моделювання отримані результати, що представлені на гістограмі часу передачі хеш-файлу метаданих у хмарні антивірусні системи (рис. 1).

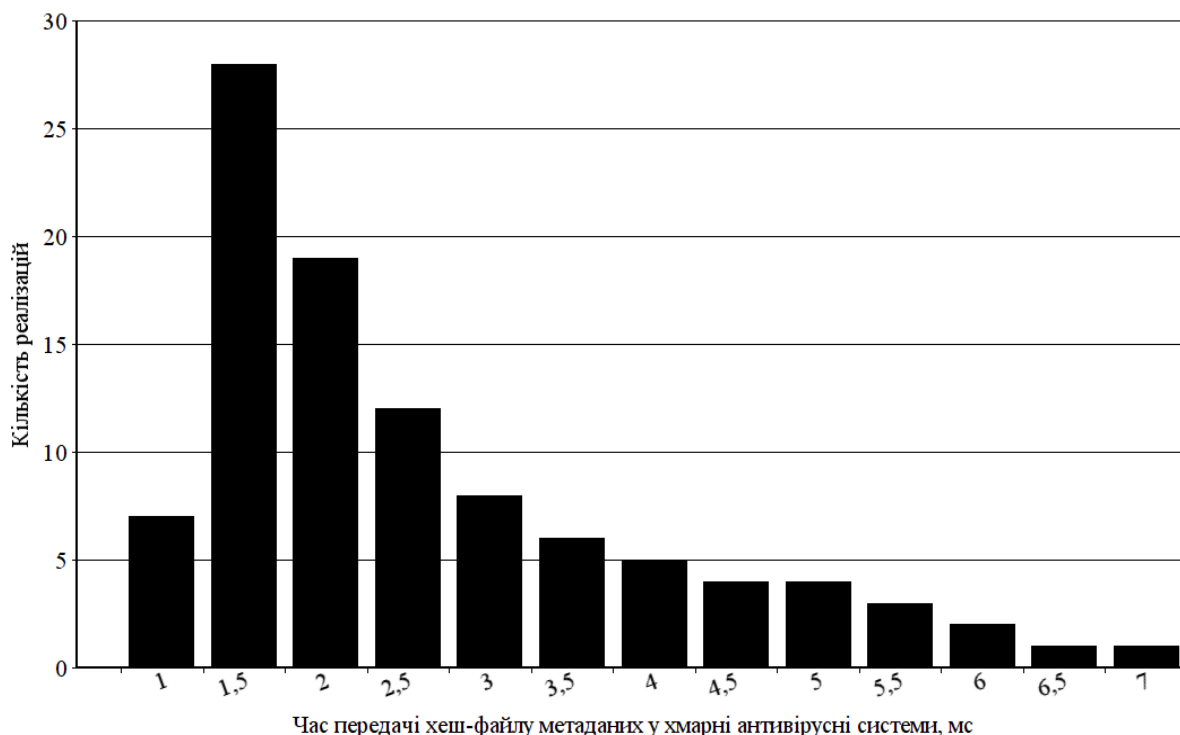


Рис 1. Гістограма часу передачі хеш-файлу метаданих у хмарні антивірусні системи

Була проведена перевірка нормального розподілу цієї випадкової величини з критерії згоди Пірсона, яка довела правдоподібність гіпотези, що величина часу передачі хеш-файлу метаданих у хмарні антивірусні системи розподілено за нормальним законом.

Також визначено довірчу ймовірність того, що значення часу передачі хешфайлу метаданих в антивірусні хмарні системи «не відхилилося» від математичного очікування більше ніж на 1. Довірча ймовірність дорівнює 0.95.

Таким чином, використання вдосконалених моделей та антивірусного захисту даних в ІКС в умовах модернізації мережевого телекомунікаційного обладнання дозволить підвищити рівень інформаційної безпеки.

Наукова новизна розробки полягає в удосконаленні методу антивірусного захисту даних в ІКС за рахунок безпечної маршрутизації метаданих у антивірусні хмарні системи.

Висновки. Проаналізовані загрози, що впливають на стан захисту даних в ІКС та за цими даними описані вимоги до інформаційної безпеки та антивірусного захисту, обрані засоби та методи забезпечення цих вимог. Проведено дослідження вдосконалення методу антивірусного захисту даних в ІКС за рахунок безпечної маршрутизації метаданих у антивірусні хмарні системи та обґрунтована доцільність його запровадження. Основними складовими рішеннями методу є алгоритми формування безлічі маршрутів передачі метаданих, метод контролю ліній зв'язку ІКС та моделі системи нейромережевих експертів безпечної маршрутизації.

ПЕРЕЛІК ПОСИЛАНЬ

1. Браницкий А.А., Котенко И.В. Анализ и классификация методов обнаружения сетевых атак // Труды СПИИРАН. 2016. Вып. 2(45). ISSN 2078-9181 (печ.), ISSN 2078-9599 (онлайн) www.proceedings.spiiras.nw.ru. С. 207-244.
2. Носенко К.М., Півторак О.І., Ліхоузова Т.А. Огляд систем виявлення атак в мережевому трафіку // Міжвідомчий науково-технічний збірник «Адаптивні системи автоматичного управління», 2014, № 1(24). - С. 67-75.
3. Лукова-Чуйко Н., Наконечний В., Толюпа С., Зюбіна Р. Проблеми захисту критично важливих об'єктів інфраструктури // Безпека інформаційних систем і технологій № 1(2), 2020. С. 31-39.
4. Довбешко С.В., Толюпа С.В., Шестак Я.В. Застосування методів інтелектуального аналізу даних для побудови систем виявлення атак // Сучасний захист інформації №1(37), 2019. С. 6-15.

УДК 004.415.3:681.6

Л.І. Мещеряков¹, А.В. Кожевников¹, С.М. Мацюк¹, А.А. Стешенко¹
¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

АНАЛІЗ БЕЗПЕЧНОСТІ ТА ШВИДКОСТІ СУЧАСНИХ АЛГОРИТМІВ ШИФРУВАННЯ

Анотація. Розглянуто важливість процесу криптографії та його поширеність в сучасних технологіях. Зроблено порівняння існуючих алгоритмів та ключів з аналізом щодо безпеки їх використання.

Ключові слова: шифрування, обчислювальне середовище, brute force, NIST, JDK.

Вступ. Криптографія є основою безпеки сучасних інформаційних систем. Вона оточує нас повсюди: зберігання облікових даних, зв'язок електронною поштою, мобільні платежі та цифрова готівка – це лише частина найпоширеніших випадків.

Криптографія є однією з найскладніших тем інформаційної безпеки, не можна піддаватися спокусі запровадити власні бібліотеки, оскільки неможливо обмеженими силами перевірити новий алгоритм на всі можливі загрози та постійно слідкувати за оновленням ризиків безпеки даних. Більшість сучасних мов мають реалізовані крипто бібліотеки та модулі, тому для прийняття оптимального рішення щодо кращого способу захисту інформації необхідно виконати детальний аналіз та порівняння.

Постановка задачі. Задля досягнення поставленої мети в роботі сформовані і вирішені наступні завдання:

- викласти принципи вибору алгоритму для найбільш безпечного шифрування;
- визначити різницю швидкості роботи найпопулярніших алгоритмів;
- виконати їх аналіз та порівняння за безпекою шифрування даних;

- визначити основний, мінімально необхідний розмір ключа; зробити висновки щодо оптимального використання та вибору.

Основний зміст роботи. Шифрування – це процес кодування або шифрування даних, щоб їх міг прочитати лише той, хто має засоби для повернення їх до початкового стану. Технології шифрування є одним із важливих елементів будь-якого безпечного обчислювального середовища.

Безпека шифрування полягає в здатності алгоритму генерувати зашифрований текст, який нелегко повернути до початкового відкритого стану. Для налаштування будь-якої базової схеми шифрування, дуже важливо, щоб наступні питання були вирішені правильно:

- наскільки безпечним алгоритм вважається на даний момент у криптографічній літературі;
- характеристики продуктивності алгоритму (наприклад чи підтримує паралельне шифрування та його «чиста швидкість»);
- наскільки політично безпечним є рішення використовувати певний алгоритм;
- використання правильно підібраних ключів і їх розмірів.

Важливо пильно налаштувати всі параметри, адже навіть незначний прорахунок в безпеці конфігурації зробить всю криптосистему відкритою для атак.

В рамках дослідження за допомогою бібліотеки `javax.crypto` було реалізовано шифрування найпопулярнішими у JDK Sun алгоритмами: AES128, AES256, SHA256, SHA512, DES, Triple DES і Blowfish. Для кожної реалізації було виміряно швидкість шифрування у мілісекундах (рис. 1) та описано основні характеристики, важливі для безпеки даних (табл. 1).

Важливо зазначити, що AES розшифровується як Advanced Encryption Standard і є алгоритмом, який більшість людей зрештою використає, якщо у них немає вагомих причин використовувати щось інше. Це рішення є політично безпечним, адже стандарт шифрування Національного інституту стандартів і технологій США (NIST) схвалює AES із 128 або 256-бітними ключами для шифрування секретних даних.

Окрім вибору алгоритму потрібно визначити розмір ключа, що забезпечує «міцність» шифрування, припускаючи, що фактичний алгоритм в інших аспектах безпечний. Розмір ключа зазвичай корелює з кількістю припущень, які зловмиснику потрібно буде зробити, щоб знайти ключ за допомогою «brute force», припускаючи, що всі можливі ключі однаково ймовірні (рис. 1).

Використання ключів додає новий рівень безпеки до методів захисту інформації. Існує дві загальні категорії алгоритмів на основі ключів:

- симетричні алгоритми шифрування: симетричні алгоритми використовують той самий ключ для шифрування та дешифрування. Ці алгоритми можуть працювати в потоковому режимі (з бітами або байтами даних) або в блочному режимі (з блоками даних фіксованого розміру). Вони зазвичай використовуються для таких програм, як шифрування даних, файлів і

особливо даних у мережах зв'язку (наприклад електронні листи, TLS, миттєві повідомлення тощо);

Таблиця 1

Порівняльна характеристика алгоритмів

Алгоритм	Розміри ключа	Швидкість	Залежність швидкості від розміру ключа	Коментар
AES	128-256	Середня	Так	Безпечний, його перевага полягає в тому, що він дозволяє використовувати 256-бітний розмір ключа, який має захистити від певних майбутніх атак (collision attacks і потенційних алгоритмів квантового обчислення)
DES	56	Повільна	Ні	Дуже небезпечно: навіть одна машина Сорасобана вартістю 10000 USD може знайти ключ DES приблизно за тиждень.
Triple DES	112-168	Дуже повільна	Ні	Доволі безпечний, особливо для невеликих розмірів даних. 168-бітний варіант, за оцінками NIST, забезпечує безпеку даних до 20304 року, проте AES пропонує вищий рівень безпеки за меншої вартості CPU.
Blowfish	128-448	Швидка	Ні	Вважається безпечним, але пройшов менше крипто аналізу, ніж інші алгоритми. Blowfish було замінено Twofish, але останній не підтримується як стандарт у Java.
SHA	256-512	Дуже швидка	Так	Проблема полягає в тому, що лише SHA256 та SHA512 (тип SHA2, який генерує 256/512-бітове хеш-значення) залишається безпечним — SHA-1 (перша версія SHA, яка генерує 160-бітове хеш-значення) має відомі вразливості.

- асиметричні алгоритми шифрування: на відміну від симетричних алгоритмів, які використовують той самий ключ для обох операцій шифрування та дешифрування, асиметричні алгоритми використовують два окремих ключі для цих двох операцій. Ці алгоритми використовуються для обчислення цифрових підписів і протоколів встановлення ключів.

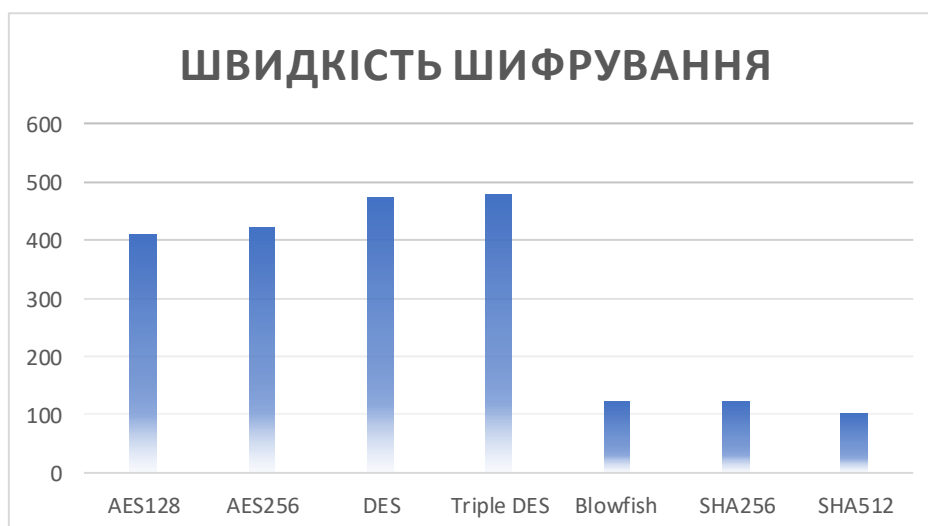


Рис. 1. Різниця швидкості роботи алгоритмів

Як обрати розмір ключа? Загалом, мінімальний необхідний розмір ключа – це деяка комбінація максимального розміру ключа, який зараз можна використати для заданого алгоритму, екстрапольованого до кількості років, які потрібно зберігати конфіденційними зашифровані дані. Наприклад, NIST рекомендує мінімум «128 біт міцності», щоб зберегти конфіденційність даних «після 2030 року». А за посібником із найкращих методів шифрування даних компанії Visa, що випускає кредитні картки, з метою передачі номерів кредитних карток «ключі повинні мати потужність принаймні 112 еквівалентних бітів». Вони фактично вважають 128 біт (як у мінімальному розмірі ключа AES) «сильнішими, ніж потрібно», хоч можливо на це впливає відносно короткий термін служби кредитних карток.

Висновки. Доклавши достатніх зусиль, практично будь-яку криптографічну систему можна успішно атакувати. Головне питання полягає в тому, скільки часу та ресурсів необхідно щоб її зламати. Як видно з цього аналізу, є багато деталей, на які слід звернути увагу, щоб зробити все правильно під час розробки та реалізації схеми шифрування.

Найбільш перевіреним та збалансованим алгоритмом шифрування зараз можна вважати AES. Для того щоб зберегти дані в безпеці принаймні 20 або 30 років краще використовувати 256-бітний ключ; 128-бітний ключ може підійти, якщо ретельно продумано захист від brute force, а також якщо протягом життя даних мало ймовірна загроза квантових обчислень. Наприклад, для даних кредитних карток можна бути впевненим, що термін дії всіх існуючих закінчиться до того часу, коли квантові комп'ютери стануть нашою реальністю.

Проведений аналіз допоможе забезпечити розумний рівень безпеки для захисту криптосистем від відомих на даний момент атак, а також захистити їх у майбутньому.

ПЕРЕЛІК ПОСИЛАНЬ

1. N. Ferguson, B. Schneider and T. Kohno Cryptography Engineering / John Wiley & Sons, 2010 – 384 с.
2. NIST Recommendation for Key Management: Part 1 - 2020. - С.53-56. - [Електронний ресурс] Режим доступу: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-57pt1r4.pdf>
3. Side-Channel Attacks on Symmetric Encryption Schemes: The Case for authenticated Encryption. - [Електронний ресурс] Режим доступу: <https://home.cs.colorado.edu/~jrblack/papers/padding.pdf>
4. C. Paar and J. Pelzi - Understanding Cryptography - 2010. - 382 с. - [Електронний ресурс] Режим доступу: <https://swarm.cs.pub.ro/~mbarbulescu/cripto/Understanding%20Cryptography%20by%20Christof%20Paar%20.pdf>
5. Java Cryptography Architecture (JCA) - [Електронний ресурс] Режим доступу: <https://docs.oracle.com/javase/8/docs/technotes/guides/security/crypto/CryptoSpec.html>

УДК 004.056:004.94

В.І. Корнієнко¹, О.І. Нікольська¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

РОЗРОБКА ОРГАНІЗАЦІЙНО-ТЕХНІЧНИХ ЗАХОДІВ ЗАХИСТУ ІНФОРМАЦІЇ ПРИ ВИКОРИСТАННІ ЦИФРОВОЇ АТС EWSD

Анотація. Проаналізовано загрози для інформації та моделі порушників, організацію та порядок технічного захисту інформації в ЦАТС. Розроблено план захисту та політику безпеки інформації на АТС EWSD, розроблено заходи захисту від витоків інформації технічними каналами, розроблено систему захисту в мережі SS7.

Ключові слова: *цифрова автоматична телефонна станція (ЦАТС), політика безпеки, захист інформації, модель порушника, рівень загрози, технічний захист інформації (ТЗІ), комплексна система захисту інформації (КСЗІ)*

Вступ. При впровадженні нових технологій передачі інформації важливим етапом є створення системи інформаційної безпеки з певним рівнем захищеності інформаційних ресурсів. В результаті досягається зниження ризиків інформаційної безпеки до прийнятного рівня як у телекомунікаційних системах, так і в інформаційно-телекомунікаційних мережах, до яких вони входять.

Забезпечення конфіденційного зв'язку є однією з важливих задач захисту інформації в телекомунікація [1-2]. Серед телекомунікаційних операторів поширення отримали цифрові АТС, що агрегують передачу голосової інформації та даних [2-3].

Таким чином, розробка організаційно-технічних заходів щодо захисту інформації ЦАТС, зокрема на базі EWSD, є актуальним завданням.

Постановка задачі. Метою роботи є забезпечення захисту інформації при використанні цифрової АТС EWSD шляхом розробки та впровадження відповідних організаційно-технічних заходів.

Для реалізації поставленої мети необхідно вирішити такі задачі:

1. Розробити модель АТС EWSD з позицій технічного захисту інформації, виконати аналіз загроз для інформації та розробити модель порушника
2. Розробити політику безпеки інформації та вимоги до календарного плану робіт із захисту інформації АТС EWSD
3. Розробити заходи щодо захисту мережі сигналізації SS7
4. Сформулювати рекомендації щодо обмеження фізичного доступу до обладнання зв'язку в абонентській мережі

Основний зміст роботи. Розроблені модель інфраструктури цифрового вузла комутації з позицій захисту інформації та структурна схема станційної частини програмно-керованої АТС EWSD з позицій ТЗІ. При цьому визначені загрози інформаційній безпеці при забезпеченні:

- 1) конфіденційності:
 - крадіжка (копіювання) інформації та засобів її обробки;
 - втрата (ненавмисна втрата, витік) інформації та засобів її обробки;
- 2) доступності:
 - блокування інформації;
 - знищення інформації та засобів її обробки;
- 3) цілісності:
 - модифікація (спотворення) інформації;
 - заперечення справжності інформації;
 - нав'язування хибної інформації.
- 4) спостережності:
 - блокування;
 - модифікація інформації;
 - маскуванню інформації;
- 5) порядку маршрутизації трафіка:
 - крадіжка трафіка;
 - несанкціоноване використання послуг та інформаційних ресурсів телекомунікаційних мереж.

Потенційним порушником безпеки інформації ЦАТС EWSD є особа, яка помилково, внаслідок необізнаності, цілеспрямовано, за злим наміром або без нього, використовуючи різні можливості, методи та засоби, здійснила спробу виконати операції, що призвели або можуть призвести до порушення конфіденційності, цілісності й доступності інформації.

Порушники:

- сторонні особи, що знаходяться за межами контрольованої зони вузлів ЦАТС EWSD ;
- відвідувачі;
- представники організацій, що взаємодіють з питань обслуговування ЦАТС EWSD , технічного забезпечення та підтримки її функціональності.

- системний адміністратор ЦАТС EWSD ;
- адміністратор безпеки;
- користувачі;
- технічний персонал, який обслуговує будівлю та приміщення, в яких розташовані компоненти ЦАТС EWSD ;
- персонал, який обслуговує технічні засоби (інженери, техніки).

Загальні положення безпеки інформаційних ресурсів у програмно-керованих ЦАТС наступні:

1. Об'єктом технічного захисту на програмно-керованих ЦАТС, а також на установчих (відомчих, корпоративних) ЦАТС є конфіденційна, а також відкрита важлива для особи, суспільства і держави інформація, яка зберігається та циркулює на цих ЦАТС.

2. Метою ТЗІ на програмно-керованих ЦАТС загального користування, а також на установчих (відомчих, корпоративних) ЦАТС є запобігання за допомогою інженерно-технічних заходів здійсненню загроз інформаційним ресурсам (далі - загроз для інформації) ЦАТС.

3. Перелік задач, спрямованих на реалізацію мети дипломної роботи:

- розробка вимог щодо захисту інформації на ЦАТС;
- створення захищених ЦАТС;
- створення систем і засобів ТЗІ на ЦАТС;
- створення систем керування комплексами засобів ТЗІ на ЦАТС;
- оцінка захищеності інформації на ЦАТС;
- оцінка ефективності систем і засобів ТЗІ на ЦАТС.

Політика безпеки визначає такі аспекти забезпечення захищеності інформаційних ресурсів у технологічному середовищі АТС EWSD:

- гарантії безпеки середовища персоналу;
- гарантії стандартизації технологічного середовища;
- гарантії забезпечення спостереження та керованості технологічного середовища;
- гарантії забезпечення конфіденційності і цілісності інформаційних ресурсів технологічного середовища;
- гарантії якості документації.

Нормальне функціонування АТС EWSD на всіх стадіях її життєвого циклу забезпечує персонал, тому необхідно забезпечити захищеність інформаційних ресурсів від їх помилкових і зловмисних дій.

Вимоги до персоналу включають у себе:

- вимоги до системи організації праці – забезпечення відповідності кваліфікації персоналу складу виконуваних робіт, система підвищення кваліфікації тощо;
- вимоги до контролю системи організації праці – контроль кваліфікації персоналу, аналіз і контроль системи підвищення кваліфікації, аналіз системи підбору кадрів, аналіз розподілу повноважень, аналіз системи оцінки якості праці тощо;

– вимоги до поведінки персоналу в робочий час – підтримання загальної дисципліни праці, виконання правил роботи з секретною інформацією та інше;

– вимоги до контролю поведінки персоналу в робочий час – контроль виконання технологічної дисципліни і поведінка персоналу у робочий час тощо;

– вимоги до поведінки персоналу в неробочий час – відсутність фактів антигромадської діяльності, аномалій у психофізичному стані організму і таке інше;

– вимоги до контролю поведінки персоналу у неробочий час – контроль достовірності даних про персонал, перевірка стану здоров'я персоналу.

Календарний план робіт із захисту інформації має такі розділи:

1. Організаційні заходи – це комплекс адміністративних та обмежувальних заходів, спрямованих на оперативне розв'язання задач захисту шляхом регламентації діяльності персоналу і порядку функціонування систем забезпечення інформаційної діяльності та засобів забезпечення ТЗІ.

2. Контрольно-правові заходи:

– контроль за виконанням персоналом (користувачами) вимог, відповідних інструкцій, розпоряджень, наказів;

– контроль за виконанням заходів, розроблених за результатами попередніх перевірок;

– контроль за станом зберігання та обігу носіїв інформації на робочих місцях.

3. Профілактичні заходи – заходи, спрямовані на формування у персоналу та користувачів мотивів поведінки, які спонукають їх до безумовного виконання у повному обсязі вимог режиму, правил проведення робіт тощо.

4. Робота з кадрами – заходи з підбору та навчання персоналу і користувачів встановленим правилам безпеки інформації, новим методам захисту, підвищення їхньої кваліфікації.

5. Інженерно-технічні заходи – заходи, спрямовані на налагодження, випробовування і введення в експлуатацію, супроводження й технічне обслуговування апаратних і програмних засобів захисту інформації від несанкціонованого доступу комплексів захисту інформації від загроз технічними каналами та каналами спеціального впливу, інженерного обладнання споруд і приміщень, в яких розміщуються засоби обробки інформації тощо.

Організація та порядок технічного захисту інформації в ЦАТС.

Для успішної технічної експлуатації КСЗІ на ЦАТС з досягненням заданого рівня захищеності інформаційних ресурсів та рівня гарантій захисту необхідно правильно організувати заходи з ТЗІ на всіх попередніх етапах створення КСЗІ, зокрема на стадіях побудови та здавання в експлуатацію:

– організація ТЗІ на стадії побудови ЦАТС;

– організація ТЗІ на стадії вводу в експлуатацію ЦАТС;

– організація ТЗІ на етапі технічної експлуатації ЦАТС;

– організація управління інформаційною безпекою;

– розробка документації щодо повноважень та відповідальності суб'єктів взаємовідносин при реалізації задач забезпечення інформаційної безпеки в ЦАТС.

Збереження інформаційних ресурсів АТС EWSD визначається умовами забезпечення їх захищеності на станції та в мережі. Для рішення цієї задачі розробляється система безпеки, основними функціями якої є захист інформації при її обробці та передачі каналами зв'язку від НСД до неї, від різноманітних програмно-технічних впливів, а також від витоку технічними каналами за рахунок побічних електромагнітних випромінювань та наводок (ПЕМВН).

Відносно принципів роботи мережі сигналізації були прийняті наступні заходи безпеки

1. Протоколи SS7 на відміну від IP-протоколів не підтримують функції шифрування й автентифікації. Тому шлюз у складі транзитного пункту сигналізації маршрутизатора мережі SS7 виконує сканування повідомлень, що надходять у мережу, щоб запобігти надходженню повідомлень з неавторизованих внутрішніх і зовнішніх вузлів мережі.

2. Мережі сигналізації не підтримують функції шифрування й автентифікації, за допомогою яких можна гарантувати достовірність вузлів зовнішньої мережі, що посилають повідомлення. Елементи самої мережі сигналізації також не захищені. Якщо зловмисник зможе відправити на деякий вузол мережі SS7 трафік, що перевищує той, на який цей вузол розрахований, останній вийде з ладу, і управління викликами в цьому секторі мережі буде порушено. Тому на транзитних пунктах рекомендується встановлювати спеціальні програми, які дозволяють операторам ідентифікувати повідомлення, що випадають із загального контексту або виявляють себе нетиповим поведінням.

3. Одним з важливих способів підвищення безпеки мережі є побудова правильної архітектури на стику IP-мережі. Кількість вузлів з'єднання цих мереж повинне бути мінімальним, повідомлення з мережі SS7 повинні надходити на IP-шлюзи централізовано. Замість численних шлюзів використовується єдиний високопродуктивний шлюз. Тоді за рахунок скорочення точок взаємодії між різнорідними частинами мережі підвищується загальна безпека мережі.

4. На сучасних мережах широко використовується централізована обробка і управління мережею SS7. У зв'язку з цим використовуються спеціальні функції, розроблені для дистанційного експлуатаційного управління АТС, заміни версії програмного забезпечення тощо. Вони також являють собою загрозу інформаційної безпеки, тому що дані функції можуть збігатися з цілями зловмисника. У зв'язку з цим для процедур віддаленого доступу необхідний моніторинг міжстанційної і міжмережної інформації, захист від загрози пересилання по мережі сигналізації спеціальних директив шляхом їх фільтрування при вході у фрагмент мережі, що захищається.

5. Нормальна робота мережі багато в чому також залежить від навантаження, створюваного на мережі SS7, тому таке навантаження необхідно контролювати.

Метод обмеження фізичного доступу до устаткування зв'язку спрямований на те, щоб унеможливити для зловмисника фізичне сприйняття інформативних сигналів, які існують у лінії зв'язку, колах апаратури та навколишньому просторі. Для досягнення такої мети слід застосовувати апаратуру, перевірену на відсутність упроваджених «закладок», пломбувати експлуатовану апаратуру, ремонт апаратури робити лише з залученням довірених фахівців під контролем власника чи співробітника служби безпеки підприємства.

Необхідно виключити будь-які ініціативні переробки впровадженої до експлуатації апаратури обслуговуючим персоналом чи ремонтниками. Особливу увагу слід звертати на легко замінювані елементи. Такі елементи слід додатково закріплювати й маркувати. Додаткове кріплення й маркування повинні бути непомітні для стороннього спостерігача, але легко перевіритися власником терміналу чи допущеним обслуговуючим персоналом.

Для унеможливлення перехоплення інформації з електромагнітних полів бажано застосовувати сертифіковану апаратуру, виконуючи вказівки щодо її розміщення.

При використанні іншої апаратури бажано провести інструментальну перевірку можливості приймання сигналів захищеної інформації у безпосередній близькості (10...15 см) від апаратури.

У цілому при організації робочого місця абонента захищеного зв'язку слід дотримуватися правил:

- на робочому місці має бути мінімум апаратури та устаткування;
- встановлення всього устаткування та елементів інтер'єра має утруднювати їхнє переміщення та заміну чи впровадження сторонніх предметів;
- на випадок, якщо відбудуться порушення розташування, заміна чи впровадження нового предмета, тоді слід вжити заходів задля виявлення й знешкодження певних дій;
- повинно бути максимально утруднене для зловмисника спостерігання за робочим процесом зв'язку й ознайомлення з системою та апаратурою захисту інформації.

Наукова новизна розробки полягає у визначенні моделі цифрової АТС EWSD з позицій технічного захисту інформації, розробці плану захисту цифрової АТС та політики безпеки інформації на ЦАТС EWSD.

Висновки. Розроблено план захисту цифрової АТС типу EWSD, який включає: опис основних об'єктів захисту, загрози інформації в ЦАТС та моделі порушників, політику безпеки інформації на ЦАТС EWSD, вимоги до календарного плану захисту інформації на ЦАТС.

На основі запропонованої схеми організації та забезпечення ТЗІ ЦАТС та мереж розроблено заходи щодо організації та порядку технічного захисту інформації в ЦАТС EWSD.

Запропоновано заходи щодо захисту мережі сигналізації SS7 та сформульовано рекомендації щодо обмеження фізичного доступу до обладнання зв'язку в абонентській мережі.

ПЕРЕЛІК ПОСИЛАНЬ

1. Корнієнко В.І. Ідентифікація і прогнозування трафіку в телекомунікаційних системах / В.І. Корнієнко, Л.В. Будкова // Системи обробки інформації – 2011. – № 8(98). – С. 208-211. – ISSN 1681-7710.
2. Будкова Л.В. Моделювання самоподібного трафіку в інформаційних телекомунікаційних мережах / Л.В. Будкова, В.І. Корнієнко // Вісник Кременчуцького національного університету імені Михайла Остроградського – 2013. – № 4(81). – С. 101-108.
3. Беллами Дж. Цифровая телефония: пер. с англ. / Дж. Беллами. – М.: Эко-Тренд, 2004. – 640 с.

УДК 519.876.5

Т.А. Желдак¹, Ю.С. Соловей¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

БІЗНЕС-АНАЛІЗ ДОДАТКА ДЛЯ ІДЕНТИФІКАЦІЇ КІБЕРЗЛОЧИНІВ У БАНКІВСЬКІЙ СФЕРІ

Анотація. Описано процес розробки алгоритму класифікації транзакції з виявленням підозрілих на шахрайство, ефективнішого за існуючі. Підвищення ефективності досягається шляхом використання сучасних алгоритмів автоматизованої класифікації великих даних та автоматизованих методів обробки вимог до програмного забезпечення.

Ключові слова: банківські картки, транзакції, класифікація, шахрайство, точність, навчання, бустінг, класифікатор, додаток, ефективність, вимоги, проектування.

Вступ. Кіберпростір створює неймовірні можливості, розширює свободу, стимулює розвиток інновацій та збагачує суспільство. Однак, паралельно з позитивними тенденціями, набуває розвитку і кіберзлочинність, що, зрозуміло, завдає значної шкоди інтересам наших громадян та держави в цілому. З метою протидії такій кримінальній протиправності в Національній поліції функціонує підрозділ кіберполіції.

Дана робота присвячена дослідженню та оптимізації діяльності управління протидії кіберзлочинам в Дніпропетровській області, яка в свою чергу входить до складу апарату центрального органу управління поліції, а також впровадженню новітніх технологій для щоденного використання в розслідуванні злочинів та запобіганні шахрайських дій в банківському секторі.

Метою роботи установи є захист населення від кіберзлочинів, запобігання неправомірних дій та розслідування і викриття вже скоєних злочинів. Основні задачі даного відділку досить широкий: поліціанти займаються боротьбою з вірусами, DdoS-атаками, спамом, шахрайством з банківськими системами і крадіжкою особистих даних.

Об'єктом дослідження є правозахисна діяльність управління протидії кіберзлочинам в Дніпропетровській області ДКП НПУ, а саме виявлення неправомірних дій в банківській сфері.

Предметом дослідження є використання методів бізнес-аналізу для розробки додатка ідентифікації кіберзлочинів у банківській сфері управлінням протидії кіберзлочинам в Дніпропетровській області ДКП НПУ.

Метою роботи є підвищення ефективності роботи кіберполіціантів при розпізнаванні шахрайських дій та подальшого пошуку злочинців, можливого прогнозування неправомірних дій в банківських системах за допомогою розробленої Antifraud detection-системи.

Постановка задачі. Для досягнення поставленої мети в роботі визначені та вирішені наступні задачі дослідження:

- дослідити методи, за допомогою яких можна побудувати класифікатор банківських транзакцій;
- розробка та побудова класифікатора на незбалансованих даних;
- аналіз отриманих результатів та вибір оптимального методу;
- проведення бізнес-аналізу додатка для ідентифікації кіберзлочинів у банківській сфері, а саме збір та обробка вимог до додатка, розробка логічної схеми БД, моделювання елементів алгоритму роботи, а також результати застосування та оцінка ефективності додатка.

Основний зміст роботи. Для розв'язання поставлених задач в роботі запропоновано використовувати наступні методи дослідження:

- методи Undersampling та Oversampling для збалансування незбалансованої вибірки [1];
- методи XGBoost, AdaBoost та градієнтного бустингу та ансамблі відповідних моделей для побудови класифікатора банківських транзакцій з метою виявлення шахрайської діяльності [2];
- різноманітні техніки виконання бізнес-аналізу додатка, тощо.

Авторами була розроблена програмна реалізація машинного навчання, а саме покрокова реалізація алгоритмів XGBoost, AdaBoost, градієнтного бустингу та ансамблів відповідних алгоритмів [3]. В якості вхідних даних було взято вибірку з відкритого джерела, так як доступ до персональних даних, скачування і подальше використання і вивчення для осіб, що не мають спеціального дозволу, заборонено чинним законодавством. Набір даних містить транзакції, здійснені кредитними картками у вересні 2013 року українськими власниками карток, що відбулися за два дні. В ході аналізу вхідних даних було виявлено, що між даними існує великий дисбаланс, адже кількість шахрайських транзакцій сягала менше 1% від усієї вибірки.

Було проведено чотири експериментальних навчань за різних умов, задля виявлення оптимальних значень. В результаті було отримано наступні результати:

Експеримент №1. Побудова класифікатора на незбалансованій вибірці:

- Метод градієнтного бустингу: TN = 94 778, FP = 36, FN = 45, TP = 77;
- Метод XGBoost: TN = 94 810, FP = 4, FN = 37, TP = 85;
- Метод AdaBoost: TN = 94 802, FP = 12, FN = 38, TP = 84.

Експеримент №2. Побудова класифікатора на вибірці, поділеній на 5 рівних частин, з метою зменшення дисбалансу в вибірці :

- Метод XGBoost: TN = 94 756, FP = 58, FN = 28, TP = 94, а оцінки точності моделі складають : recall = 0.77, precision = 0.62, accuracy = 1.00;
- Метод AdaBoost: TN = 94 380, FP = 434, FN = 27, TP = 95 а оцінки точності моделі складають : recall = 0.81, precision = 0.01, accuracy = 0.92.

Експеримент №3. Побудова класифікатора на збалансованій вибірці методом Undersampling (NNK):

- Метод XGBoost: TN = 93 640, FP = 1174, FN = 24, TP = 98, а оцінки точності моделі складають : recall = 0.80, precision = 0.08, accuracy = 0.99;
- Метод AdaBoost: TN = 80 423, FP = 14 391, FN = 15, TP = 107 а оцінки точності моделі складають : recall = 0.88, precision = 0.01, accuracy = 0.85.

Експеримент №4. Побудова класифікатора на збалансованій вибірці методом Oversampling (Random):

- Метод XGBoost: TN = 94 656, FP = 158, FN = 23, TP = 99, а оцінки точності моделі складають : recall = 0.81, precision = 0.39, accuracy = 1.00;
- Метод AdaBoost: TN = 92 825, FP = 1 989, FN = 23, TP = 99 а оцінки точності моделі складають : recall = 0.81, precision = 0.05, accuracy = 0.98.

Аналізуючи результати машинного навчання варто відзначити, що метод градієнтного бустингу виявився не оптимальним, адже давав низькі за точністю результати і час виконання навчання був дуже довгим в порівнянні з іншими двома методами, тому його було виключено з подальших розрахунків. Також в ході реалізації двох методів було застосовано методи семплінгу, що посприяли покращенню результатів машинного навчання. Спираючись на результати експериментів можна вважати експеримент № 3 і №4 результативними, проте вибір конкретного методу бустингу і семплінгу може варіюватися.

Алгоритм XGBoost має більш вищий показник точності класифікатора (більш точно відокремлює «хороші» транзакції), а також має більш швидкий час виконання алгоритму навчання, а у алгоритму AdaBoost кращий показник глибини класифікатора (більш точно відокремлює «погані» транзакції), зумовлено тим, що алгоритм AdaBoost навчається на «поганих» результатах попереднього навчання. Тобто з кожним кроком все зменшується кількість поганих результатів навчання. Проте за рахунок цього достатньо велика кількість, можливо хороших транзакцій потрапляють до «підозрілих».

В такому випадку можна відштовхуватися від побажань замовника подібних класифікаторів або ж доречно використовувати ці два алгоритми разом. Вони, так би мовити, будуть збалансовувати результати навчання одне одного.

В нашому випадку замовником є державна правоохоронна установа, метою якої є виявити якомога більше підозрілих транзакцій, так як вони в подальшому можуть скласти більшу загрозу. Тому можна вважати, що в даному випадку оцінка глибини є більш опорною, ніж точність моделі.

В практичному розділі було проведено бізнес-аналіз додатку протидії кіберзлочинам в банківській сфері. Тобто це можна вважати вже цілісним

проектом по впровадженню Antifraud detection-системи, адже розробки самого функціоналу недостатньо. Для ефективного його використання необхідно розробити зручний функціонал, який буде відповідати усім вимогам користувача, та буде містити увесь необхідний функціонал.

Особливості побудованої Antifraud detection-системи:

- Обробка подій у реальному часі.
- Система відповідає найжорсткішим вимогам до швидкості – тисячі подій обробляються в режимі real-time.
- Максимум виявленого шахрайства при мінімумі хибних спрацьовувань.
- Одночасне застосування експертних правил та самонавчальних моделей.
- Відкрита технологія керування для коригування політик та моделей.
- Рішення побудоване на відкритій технології управління.
- Зручний інтерфейс для швидкого прийняття рішень.
- Можливість створення та наповнення даними різних бізнес-об'єктів.

Усе це стало можливим завдяки методам бізнес-аналізу [4], що дозволили виконати збір необхідних вимог до додатку шляхом проведення інтерв'ю користувачів та опитувань. Також вагому роль відіграло визначення потенційних стейкхолдерів (зацікавлених осіб) продукту, адже це дозволило побудувати всебічну модель додатку, яка враховує усі вимоги, очікування та інтереси та протистояти обмеженням, які були визначені у Stakeholder map.

Також важливим кроком на етапі розробки додатку було побудова інформаційної архітектури додатку, а також відтворення user flow [5], що допомогло зрозуміти чи відбулися покращення у шляху юзера по досягненні однієї і тієї ж цілі в порівнянні з шляхом до впровадження додатку.

Практична цінність отриманих результатів полягає у отриманні готового додатку класифікатора банківських транзакцій, який дозволить визначати до якого класу відноситься та чи інша транзакція, тобто чи є вона «хороша», або «погана», блокувати їх, а також з використанням внутрішніх та зовнішніх банків даних та зручного функціоналу відшукувати злочинців, швидко отримувати необхідні дані про них та попереджати подальші неправомірні злочини. Тестування додатку вже за три місяці показало, що раніше в середньому на збір необхідної інформації по справі йшло 9-10 днів, наразі це займає чотири дні, це з урахуванням того факту, що зазвичай шахраї не обходяться одним злочином, а мають велику кількість жертв, і тому саме збір доказів по всім випадкам подовжує час обробки та збору інформації по справі.

Економічний ефект від отриманих результатів очікується позитивним завдяки тому, що зменшується щорічний збиток від фінансового шахрайства в сфері онлайн платежів: впровадження на три місяці готового функціоналу допомогло нам заощадити майже 40 млн. грн у 2022 році і становить 151 млн. грн, а прогнозоване значення на 2023 рік становить 99 млн грн, що на 35% менше ніж у 2022 році та на майже 50% менше ніж у 2020 році.

Соціальний ефект від результатів роботи очікується позитивним через скорочення кількості злочинів в фінансовій сфері, захист населення від

неправомірних транзакцій, тобто призводить до підвищення довіри населення до органів Національної поліції України. Складність впровадження такої системи полягає у тому, що необхідно залучати команду спеціалістів, що мають реалізувати впровадження і забезпечити подальше супроводження продукту, проводити аналізування результатів класифікатора, задля виявлення поламок функціоналу і швидкого їх усунення, адже така система потребує знань та навичок в багатьох сферах інформаційних технологій, якими не може володіти кіберполіціант. З іншого ж підтримка з боку держави, яке є одним зі стейкхолдерів та залучення інших меценатів мають допомогти в зборі необхідних коштів та персоналу, адже це значно покращуватиме положення захисту населення в банківській сфері і підвищуватиме довіру до НПУ та влади.

Наукова новизна розробки полягає у застосуванні сучасних алгоритмів класифікації транзакцій для підвищення ефективності роботи кіберполіціантів при розпізнаванні шахрайських дій та подальшого пошуку злочинців, можливого прогнозування неправомірних дій в банківських системах.

Висновки. В результаті було у отримано готовий додаток класифікатора банківських транзакцій, який дозволяє визначати до якого класу відноситься та чи інша транзакція, тобто чи є вона «хороша», або «погана», блокувати їх, а також з використанням внутрішніх та зовнішніх банків даних та зручного функціоналу відшукувати злочинців, швидко отримувати необхідні дані про них та попереджати подальші неправомірні злочини. Це має призвести до скорочення кількості злочинів в фінансовій сфері, захист населення від неправомірних транзакцій, відтак до підвищення довіри населення до органів Національної поліції України.

ПЕРЕЛІК ПОСИЛАНЬ

1. Паклін, М.Б. Побудова класифікаторів на незбалансованих вибірках на прикладі кредитного скорингу / М.Б. Паклін, С.В. Уланов, С.В. Царьков // Штучний інтелект. – 2010. В. 3. – с. 528-534.
2. Schapire R.E. The Boosting Approach to Machine Learning: An Overview [Архівовано 20 вересня 2020 у Wayback Machine.], MSRI (Mathematical Sciences Research Institute) Workshop on Nonlinear Estimation and Classification. – 2003. – 23 p.
3. Zhou, Zh.H. On the doubt about margin explanation of boosting / Wei Gao, Zhi-Hua Zhou // Artificial Intelligence 203: 1–18. – 2013. doi:10.1016/j.artint.2013.07.002.
4. Laplante, Ph. Requirements Engineering for Software and Systems (вид. 1st). Redmond, WA: CRC Press. - 2009. – 264 p.
5. Berenbach B. Software & Systems Requirements Engineering: In Practice. / Brian Berenbach, Daniel Paulish, Juergen Katzmeier, Arnold Rudorfer // New York: McGraw-Hill Professional. – 2009. – 546 p.

РОЗДІЛ 6

ПРОБЛЕМИ ДИСТАНЦІЙНОЇ ОСВІТИ

УДК 159.95

М.О. Алексеев¹, А.В. М'якенький¹, О.М. Алексеев¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

МОДЕЛЬ ОЦІНКИ КОГНІТИВНИХ НАВИЧОК СТУДЕНТА

Анотація. Розроблено модель оцінки когнітивних навичок студента, на підставі аналізу якої робиться висновок щодо необхідності зміни навчальної методики, що дозволило адаптивно змінювати методику викладання матеріалу під час навчання студента.

Ключові слова: когнітивне моделювання, навчальна методика, випадковий ліс, розумова діяльність, здібність до навчання.

Вступ. Процес пізнання належить до розумової діяльності людини, поєднуючи у собі процеси сприйняття, уваги, пам'яті, мислення, розуміння, розв'язання проблем та прийняття рішень та відіграє важливу роль у взаємодії людини та світу, що її оточує.

Дослідження цього процесу, а також інших аспектів розумової діяльності людини є темою досліджень когнітивної науки.

Аналіз та дослідження результатів когнітивного моделювання дозволяє встановлювати зміни у розумовій та ментальній моделі студента під час навчання та робити висновки щодо ефективності навчальної методики та необхідності її зміни, що зумовлює актуальність обраної теми.

Мета роботи – підвищення ефективності навчання студента через адаптивне формування навчальних методик на основі аналізу його когнітивної моделі.

Постановка задачі: У якості моделі, на підставі якої робиться висновки щодо добору навчальних методик, була побудована та навчена модель класифікації з використанням алгоритму випадкового лісу.

Об'єктом класифікації є результат виконання студентом контрольних або проміжних завдань з дисципліни за окремою навчальною методикою. На виконання завдання студентові надається лімітована кількість спроб, а результат визначається станом «склав» або «не склав». Таким чином вихідними є класи, отримані комбінацією станів завдання «склав» або «не склав» та з якої спроби був отриманий стан завдання.

Ознаками класифікації є обчислені когнітивні навички студентів у результаті виконання завдання та теми, які охоплюються завданнями тестових матеріалів. Кожне завдання у розробленій моделі охоплює три теми з набору тем дисципліни. Даний набір ознак дозволяє класифікувати результати

тестування окремих студентів та робити висновки щодо покращення навчальної методики.

Щоб отримати уявлення про початкові когнітивні навички, студенти проходять вступне тестове завдання дисципліни. Отриманий набір даних використовується для навчання моделі. Після цього студенти починають вивчати матеріали дисципліни за стандартною навчальною методикою курсу, за якими надається наступний тестовий проміжний контроль. Результати тестування використовуються для прогнозування успіхів студенту за попередньою моделлю, а після цього порівнюються з реальними результатами. На основі порівняння робиться висновок щодо ефективності обраної методики для кожного студента. У результаті висновків навчальна методика знову змінюється, модель доповнюється отриманими даними і процедура повторюється до моменту задовільних результатів.

Запропонована методика була застосована на наборі даних результатів тестувань студентів. На рис. 1 представлений графік прогнозованих успіхів окремого студента у виконанні окремого завдання на заданий набір тем, а на рис. 2 представлені реальні результати, отримані студентом під час виконання цього завдання.



Рис. 1. Графік прогнозованих успіхів студента після зміни навчальних методик

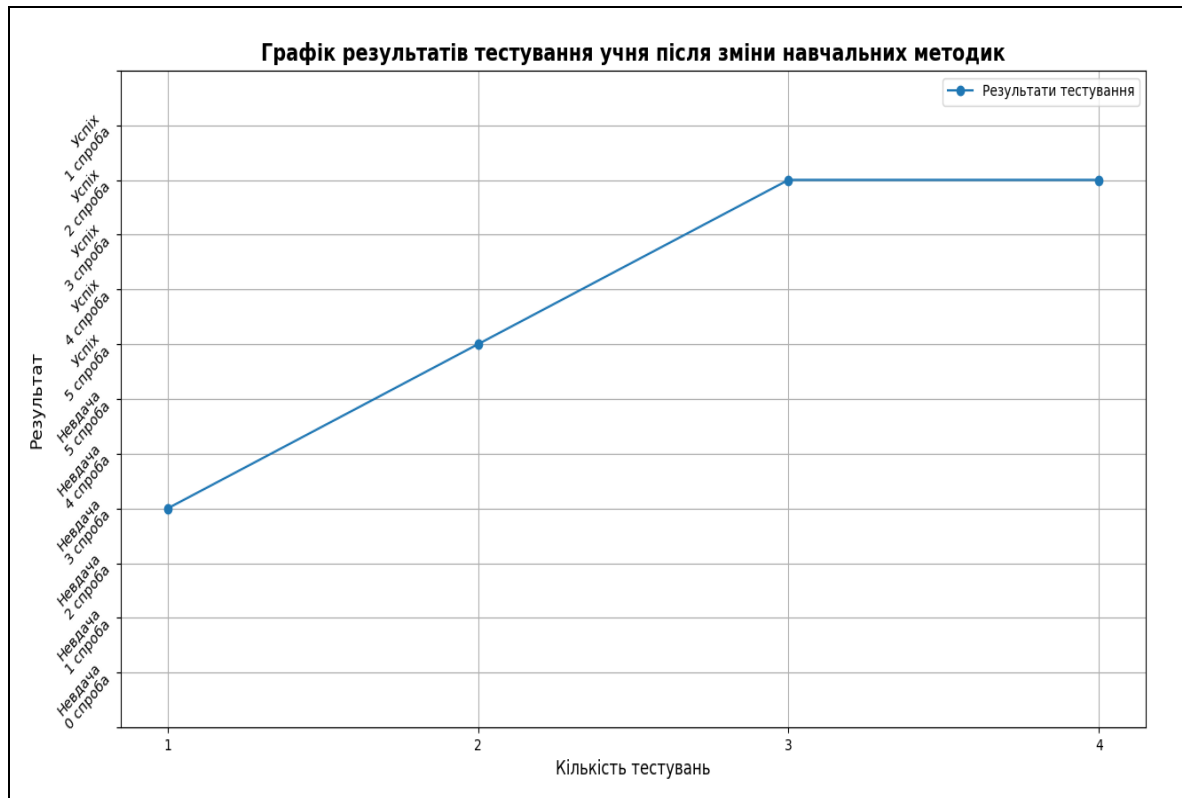


Рис. 2. Графік результатів тестування студента після зміни навчальних методик

Після кожного тестування результати моделі порівнювались с реальними результатами та робився висновок щодо покращення навчальної методики. Тенденцію ефективності добору навчальних методик показує графік порівняння результатів моделі та реальних результатів, зображений на рис. 3.

Після першого тестування прогнозований та реальні результати показують невдачу у виконанні студентом поставленого завдання. З цього можна зробити висновок, що початкова методика навчання не є ефективною і потребує змін. Після корегування навчальних методик на наступному проміжному тестуванні за прогнозованими результатами студент все ще не мав скласти завдання успішно, але реальні результати показують, що студент використав усі спроби, але виконав завдання успішно. На основі цього можна зробити висновок, що зміна навчальної методики для цього студента покращила його когнітивні навички та його результат з даних тем. Після доповнення навчальної методики, прогнозований та реальний результат показали успіх з невеликою різницею у спробах, що означає, що модифікована навчальна методика значно покращила опанування матеріалу студентом. Подальші вдосконалення методики не надали результатів покращення. Причиною цього може бути або досягнення межі можливостей когнітивної системи студента, або відсутність мотивації досягнення максимального результату або комбінація цих факторів.

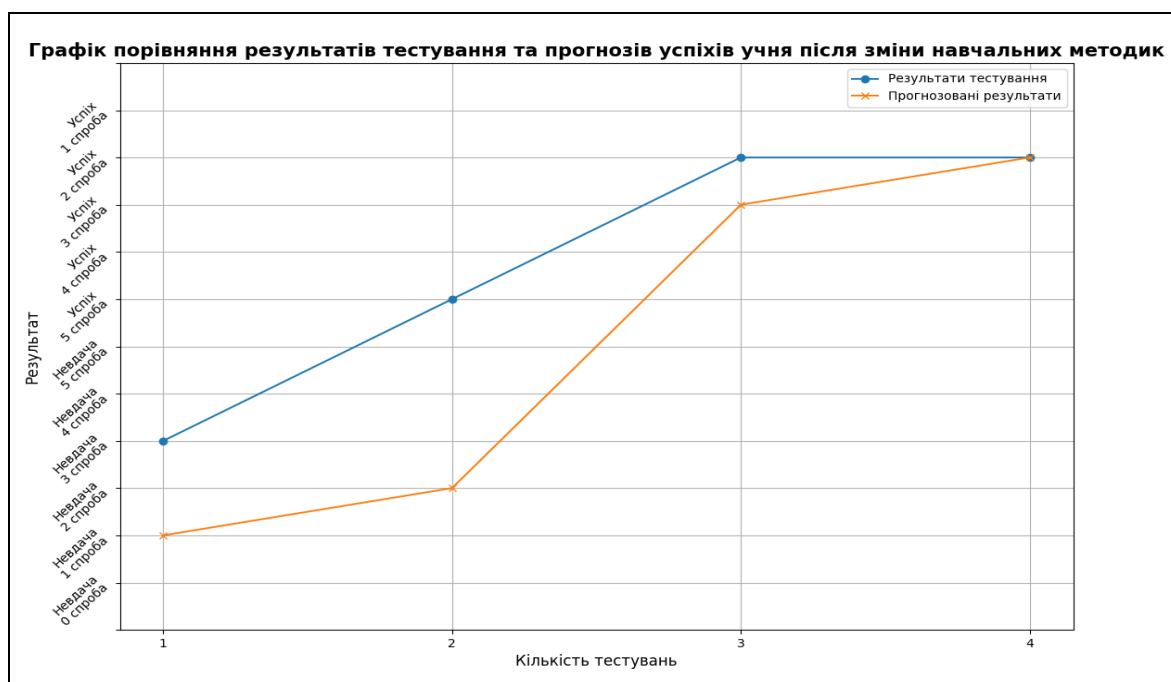


Рис. 3. Графік порівняння результатів тестування та прогнозів успіхів студента після зміни навчальних методик

Висновки. Запропоновано метод добору навчальних методик на основі аналізу когнітивних навичок студента, обчислених за результатами виконання контрольних заходів дисципліни. З цією метою були описані набір даних з експертними оцінками когнітивних навичок для кожного завдання з контрольних матеріалів та тем, що охоплюють ці завдання, а також набір даних про результати виконання групою студентів тестових завдань з обрахованими оцінками когнітивних навичок на основі системи штрафів. Отримані набори даних використовувались для навчання моделі прогнозування успіхів студента, на підставі аналізу яких приймалось рішення про зміну або доповнення навчальної методики. Отримані результати показують ефективність запропонованого методу, оскільки вони показують покращення результатів контролю знань студента та відповідно зростання його когнітивних навичок.

ПЕРЕЛІК ПОСИЛАНЬ

1. Bender A., Beller S. Cognition and cognitive science / A. Bender, S. Beller // *Frontiers in Psychology*. – 2016. – Vol. 7. – P. 24-30.
2. Sun R. Cognitive science meets multi-agent systems: A prolegomenon. / R. Sun // *Philosophical Psychology*. – 2001. – Vol. 14. – P. 5-28.
3. Wooldridge M., Jennings N. R. Intelligent agents: theory and practice / M. Wooldridge, N. R. Jennings // *The Knowledge Engineering Review*. – 1995 – Vol. 14. – P. 115 – 152.
4. Sun R. Introduction to computational cognitive modeling / R. Sun // *The Cambridge handbook of computational psychology* – Cambridge University Press, 2008. – P. 3-19.
5. Frischkorn G.T., Schubert A. L. Cognitive Models in Intelligence Research: Advantages and Recommendations for Their Application / G.T. Frischkorn, A.L. Schubert // *Journal of Intelligence*. – 2018. – Vol. 6, № 3. – P. 1-22.

А.Т. Харь¹, В.Н. Куваев¹, О.В. Реута¹, Д.В. Мельніков¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ДИСТАНЦІЙНА ОСВІТА ПІД ЧАС ВІЙНИ

Анотація. У статті розкрито основні аспекти онлайн-навчання в умовах війни на прикладі студентів спеціальності 121 «Інженерія програмного забезпечення». Визначено, що в умовах воєнного стану закладам вищої освіти необхідно виконати свої першочергові обов'язки по наданню знань всім здобувачам освіти незалежно від їх місцезнаходження для вирішення проблеми доступу до високоякісної освіти.

Ключові слова: дистанційне навчання, навчання під час війни, синхронна форма, асинхронна форма.

Вступ. З березня 2020 року оголошено подовжений карантин через поширення COVID-19, що призвело до переходу освіти в Україні у дистанційну форму. Наступні два навчальні роки проходили у гібридному форматі (оскільки в населених пунктах та окремих навчальних закладах регулярно запроваджувався карантин).

24 лютого 2022 року Росія здійснила широкомасштабне вторгнення на українську територію, внаслідок чого навчання знову було переведено у дистанційний режим. На даний момент викладачі вже мають певний досвід організації дистанційного навчання, але підготовка під час війни мала свої особливості, що, звичайно, також ускладнювало її організацію. Механізм здійснення дистанційного навчання в умовах воєнного стану кидає низку викликів освітній системі, які були немислимими під час ковідних карантинів. Література про освіту за умов воєнного стану зазвичай застаріла й подекуди стосується ще Другої світової війни та освітньої галузі того періоду.

Постановка задачі. Під час війни сформувалося кілька груп студентів: студенти, що знаходяться вдома (на неокупованих та окупованих територіях), і студенти, що були внутрішньо або зовнішньо переміщені. Звичайно, ситуація кожного студента індивідуальна, але ці групи студентів мають певні схожі характеристики, що дозволяє виконати узагальнення вимог та сформулювати рекомендації щодо організації дистанційного навчання для кожної групи.

Основний зміст роботи. Дистанційне навчання – форма організації навчального процесу, за якої весь курс або його частина проводиться в межах територіальної відстані викладачів і студентів з використанням сучасних інформаційно-телекомунікаційних технологій. Дистанційне навчання допускає гнучкість і різноманітність навчання, що допоможе повніше розкрити потенціал студента і надати можливість тим здобувачам, хто пропускає заняття з поважних причин (змагання, хвороба тощо), та особам, які за станом здоров'я навчаються індивідуально, якісно задовольнити власні потреби в здобутті освіти.

Дистанційне навчання під час війни може відбуватися в синхронному та асинхронному режимах [1]. У процесі вивчення питання було встановлено переваги та недоліки обох з цих режимів (Рис.1) [2].

Синхронне навчання – майже безпосередній зворотний зв'язок з обох сторін. Викладач може опитати студента, оцінити його відповідь, задати додаткові питання. Така взаємодія краща для засвоєння матеріалу, тому що студенти можуть задати питання по складним для опанування темам. Звичайно, є і мінуси: переривання зв'язку, технологічні негаразди, неможливість колективного обговорення та групової роботи.

Перевагою асинхронного режиму є гнучкість графіку роботи, можливість зручного розпорядку дня та роботи у власному темпі. Водночас, недоліком є зниження навчальної дисципліни, зменшення соціалізації, важкість зворотного зв'язку. Асинхронне навчання вимагає більше зусиль від студентів, які повинні навчитись правильно організувати свій час, серйозніше ставитись до завдання, мати навички самодисципліни.

<p>ПЕРЕВАГИ ОНЛАЙН-НАВЧАННЯ ДЛЯ ВИКЛАДАЧА:</p> <ul style="list-style-type: none">– високий рівень ефективності через використання інтерактивних інструментів;– легкість у відстежуванні прогресу кожного здобувача освіти;– можна практично будь-де проводити заняття;– на заняття ніхто не запізниться – завдяки мобільній версії та додатку є можливість вчасно приєднатися до заняття.	<p>НЕДОЛІКИ ОНЛАЙН-НАВЧАННЯ ДЛЯ ВИКЛАДАЧА:</p> <ul style="list-style-type: none">– потрібні знання у користуванні платформою для онлайн-навчання;– потрібно опанувати прийоми для ефективного проведення занять;– до проведення занять необхідно ретельно готуватись;– інколи перешкоджають проведенню занять технічні несправності.
<p>ПЕРЕВАГИ ОНЛАЙН-НАВЧАННЯ ДЛЯ ЗДОБУВАЧА ОСВІТИ:</p> <ul style="list-style-type: none">– є можливість обирати час та місце проведення занять;– легкість у контактуванні із викладачем через пошту, чат, дистанційні платформи;– сором'язливі здобувачі освіти на онлайн-заняттях отримують можливість легше навчатись;– таке навчання дає можливість здобувачам освіти із обмеженими можливостями навчатись з дому.	<p>НЕДОЛІКИ ОНЛАЙН-НАВЧАННЯ ДЛЯ ЗДОБУВАЧА ОСВІТИ:</p> <ul style="list-style-type: none">– брак контакту між здобувачем освіти та викладачем;– здобувачу освіти потрібно значно більше часу витратити на те, щоб освоїти матеріали навчальної програми;– через зовнішні фактори виникає неможливість чітко зосередитись на навчанні.

Рис.1. Переваги та недоліки онлайн-навчання для викладачів та студентів [2]

В умовах війни не кожен студент має змогу навчатися дистанційно (може перебувати в небезпеці, не мати засобів для навчання, тощо). Серед учасників опитування [3] приблизно 90% тих, хто знаходяться на неокупованих

територіях, мають змогу навчатися дистанційно, але 95% учнів, що знаходяться на територіях, захоплених окупантами (навіть ті, що періодично мають доступ до інтернету і зв'язок із викладачем) стверджують, що у них немає можливості дистанційно навчатися.

Водночас навіть у тих студентів, хто може навчатися, під час дистанційного навчання також виникають проблеми – не всі викладачі зможуть здійснювати дистанційне навчання в синхронному режимі – проводити онлайн лекції, а здобувачі освіти мають проблеми з підключенням до Інтернету під час приєднання до занять, своєчасністю виконання та надсилання онлайн завдань педагогам [4].

На думку автора [5], Міністерство освіти і науки України не було готове до здійснення навчання в умовах війни. На підтримку висловленої тези було використано аргумент, що у Стратегії розвитку державного підприємства «Український державний центр міжнародної освіти» на 2021-2025 роки у розділі «Управління ризиками» [6] навіть не згадано про можливе воєнне вторгнення (серед ризиків названо високу толерантність до корупції, продовження карантину, економічна криза в країні та брак якісних змін у нормативно-правовій базі). Таким чином, досвід дистанційного навчання під час пандемії став рятівним.

Звичайно, онлайн-навчання в умовах воєнного стану не може проходити так, як це відбувалося під час карантину. Попри розвиток військових дій в Україні, логіка та потенціал організації навчального процесу залишилися незмінними, що дозволило запустити процеси перетворення, що узагальнено відображаються в наступних основних тенденціях адаптації онлайн-навчання у закладах вищої освіти до умов війни:

1) перевага форм асинхронної взаємодії між викладачами та студентами (основне використання асинхронних платформ Moodle, Google Classroom тощо);

2) запис відео, можливість створення презентації викладачів і групування здобувачів освіти у онлайн-класи, що дозволить їм переглядати лекції чи семінари у відео форматі;

3) належні терміни проведення занять;

4) негайне переривання синхронної взаємодії зі студентами у випадку повітряної тривоги (попередньо студенти мають бути проінструктовані щодо алгоритму їхніх дій на випадок тривоги); завдання та матеріали, що не були опрацьовані у зв'язку з перериванням на час тривоги, виконуються асинхронно;

5) використання творчих завдань, заохочення освітньої та соціальної ініціативи тощо.

Для дистанційного навчання студентів, які знаходяться вдома, або є внутрішньо або зовнішньо переміщеними особами, але мають доступ до стабільного якісного Інтернету та технічних засобів, переважаючою формою взаємодії є синхронне навчання, з використанням он-лайн зустрічей, відео- та аудіоконференцій, консультування в режимі реального часу.

Але для студентів, які знаходяться у населених пунктах, близьких до зони ведення бойових дій, на окупованих територіях, або в місцях тимчасово притулку переміщених осіб, дана форма взаємодії здебільшого є

неприйнятною. Треба враховувати такі перешкоди, як тимчасова або постійна відсутність електрики, підключення до Інтернету, мобільного зв'язку, доступу до комп'ютерної техніки. Інтернет-з'єднання часто може бути нестабільним, обмеженим по об'ємам трафіку. У таких ситуаціях асинхронна форма навчання є не просто домінуючою, але і єдиною можливою.

Студенти спеціальності 121 «Інженерія програмного забезпечення» крім засвоєння теоретичних знань повинні також формувати практичні навички розробки та створення програмного забезпечення, що передбачає роботу з середовищами програмування, компіляторами, емуляторами, графічними редакторами, аналітичними платформами. Важливою є не лише колективна робота і обговорення матеріалу, але й індивідуальна експертиза та консультування викладачем у процесі створення та налагодження навчальних програм. Таким чином, враховуючи всі особливості та потреби здобувачів спеціальності 121 «Інженерія програмного забезпечення», для забезпечення доступу до освітніх послуг студентам всіх категорій пропонується дотримуватись наступних процедур та рекомендацій:

1. Програмні засоби та платформи, що використовуються для здійснення дистанційного навчання, повинні обов'язково мати адаптивний інтерфейс або версії клієнтських застосунків, призначені для роботи на мобільних пристроях;

2. За можливістю, потрібно дублювати відеоконтент (відеозаписи лекцій або додаткові матеріали) текстовими матеріалами, для цього також можна використовувати функцію транскрибування;

3. Всі текстові документи мають бути доступні для перегляду без завантаження на пристрій студента;

4. Забезпечення підтримки спрощеного режиму перегляду сторінок дистанційних курсів;

5. Використання онлайн-компіляторів та інтерактивних середовищ програмування, які дозволяють виконувати лабораторні та практичні роботи без наявності на комп'ютері студента встановленого програмного забезпечення або з використанням мобільного пристрою;

6. Гнучкий режим консультацій та можливість індивідуального консультування, у тому числі засобами платформ дистанційної освіти.

Виконання даних вказівок дасть можливість долучати до навчального процесу тих студентів, що не мають доступу до ноутбуку або персонального комп'ютера, низьку швидкість Інтернет-з'єднання, обмежений об'єм Інтернет-трафіку, та інших умов для навчання.

Наукова новизна. Поточне дослідження розширює горизонти для подальшого ретельного аналізу впровадження дистанційного навчання для вирішення проблеми неналежного доступу до високоякісної освіти в областях, де відбуваються активні воєнні дії.

Висновки. Дистанційне навчання є не лише вимушеним кроком у складних умовах, але й має довгострокові впливи на подальше функціонування системи вищої освіти в Україні. На думку деяких експертів, повернення до повністю очного навчання без часткового застосування змішаних

(дистанційних) технологій вже навряд чи відбудеться [7]. Тому актуальним напрямом для подальших досліджень є формування принципів та вимог до контролю ефективності навчання при синхронній та асинхронній формах взаємодії для здобувачів вищої освіти спеціальності 121 «Інженерія програмного забезпечення».

ПЕРЕЛІК ПОСИЛАНЬ

1. Синхронне й асинхронне дистанційне навчання [Електронний ресурс] Режим доступу: <https://osvita.ua/school/method/78950/>
2. Рудницька О.Г. (2022). Перспективи онлайн-навчання в умовах війни // О.Г. Рудницька, П.В. Кузик, В.Й. Дзямко / *Наука і техніка сьогодні*, №7 (7), 196-204. DOI: [https://doi.org/10.52058/2786-6025-2022-7\(7\)-196-204](https://doi.org/10.52058/2786-6025-2022-7(7)-196-204)
3. Васильєва, Д. (2022). Стан дистанційного навчання математики під час війни в Україні. *Український Педагогічний журнал*, (2), 38–47. <https://doi.org/10.32405/2411-1317-2022-2-38-47>
4. Освітній омбудсмен України «2022/2023 навчальний рік у складних умовах.» [Електронний ресурс] Режим доступу: <https://eo.gov.ua/2022-2023-navchalnyy-rik-u-skladnykh-umovakh/2022/10/21/>
5. Крамаренко І.С. (2022). Адаптація дистанційного навчання у вищій школі до умов воєнного стану // І.С. Крамаренко, Т.Л. Корнішева, І.О. Сілютіна. / *Перспективи та інновації науки*, No 4(9), 192-193. DOI: [https://doi.org/10.52058/2786-4952-2022-4\(9\)-192-205](https://doi.org/10.52058/2786-4952-2022-4(9)-192-205)
6. Стратегія розвитку державного підприємства «Український державний центр міжнародної освіти» на 2021-2025 роки [Електронний ресурс] Режим доступу: <https://studyinukraine.gov.ua/strategiya/>
7. Ніколаєв Є. Що зміниться у вищій школі завдяки онлайн-освіті? [Електронний ресурс] Режим доступу: <https://zn.ua/ukr/EDUCATION/shcho-zminitsja-u-vishchij-shkoli-zavdjaki-onlajn-osviti.html>

УДК 378.633.018.43:004.58

В.В. Спирінцев¹, А.Л. Ширін¹, П.О. Щербина¹, Д.Ю. Садовий¹

¹Національний технічний університет «Дніпровська політехніка», Дніпро, Україна

ООНОВЛЕННЯ ІСНУЮЧИХ ВЕБ-РЕСУРСІВ ЗАКЛАДІВ ВИЩОЇ ОСВІТИ ЯК ВДОСКОНАЛЕННЯ НАДАННЯ ОСВІТНІХ ПОСЛУГ

Анотація. В роботі розглянуто можливості оновлення веб-ресурсів вищих навчальних закладів (ВНЗ) завдяки застосування сучасних веб-технологій. Для підвищення користувальницької юзабільності пропонується використання таких інструментів, як: JavaScript-бібліотека React, платформа Node.js, мова таблиць стилів Sass та система керування базами даних MongoDB. Описано переваги цих інструментів перед програмним забезпеченням, яке використовують заклади освіти зараз.

Ключові слова: веб-додаток, сайт, веб-ресурс, React, Node.js, Sass.

Вступ. Сьогодні ми не можемо уявити своє життя без використання всесвітньої павутини World Wide Web. Кожного дня у мережу потрапляє близько 5-6 мільярдів запитів. В залежності від певних факторів обробки запитів пошуку

клієнта, браузер, в якості відповіді, надає відповідний список веб-ресурсів. Для цього обчислюється релевантність (ступінь відповідності введеному запиту) цих ресурсів. Звісно сортування не відбувається випадково, для цього, за допомогою пошукових роботів, враховується безліч внутрішніх та зовнішніх параметрів кожного сайту, які характеризуються поняттям пошукова оптимізація сайту або SEO-оптимізація (від англ. Search Engine Optimization) [1]. Наприклад щільність ключових слів, індекс цитування, швидкість завантаження сайту та поведінкові фактори (зовнішні та внутрішні) впливають на позицію, яку займає сайт, у списках результатів пошукової системи браузера.

Основний зміст роботи. Як не дивно, студенти теж користуються різноманітними інтернет технологіями, у тому числі для пошуку інформації, користуючись різними освітніми веб-ресурсами. Якщо взяти до уваги навчання, то одним з найголовніших ресурсів для студента є сайт його ВНЗ. Нажаль, далеко не усі сучасні учбові портали можуть надати відповідну користувальницьку юзабільність. Більшість з них були побудовані близько 20 років тому на платформі CMS-системи 1С-Bitrix використовуючи мову PHP. Відсутність адаптивності під мобільні пристрої та невідповідність структури UX/UI-факторам (UX – функціонал інтерфейсу, UI – його зовнішній вигляд) роблять їх користувальницько неорієнтованими. Впевнений, що основною причиною продовження користування даними технологіями є звичність використання.

У даній роботі розглядається можливість розробки веб-додатків замість існуючих веб-сайтів ВНЗ з використанням наступних технологій:

- React (front end) – відповідає за клієнтську частину;
- Node.js (back end) – відповідає за серверну частину;
- Sass – препроцесор для зручної стилізації додатку;
- MongoDB – в якості бази даних.

Звертаю вашу увагу на те, що рекомендується розробити саме «веб-додаток», а не «сайт» чи «веб-ресурс». Основною причиною цього є більша зручність користування. Якщо дуже просто пояснювати, то веб-сайт це сукупність статичних веб-сторінок зв'язаних між собою гіперпосиланнями одна на одну, при переході по яким відбувається повне оновлення веб-сторінки і повторна її загрузка. Веб-додаток, у свою чергу, припускає динамічну взаємодію без перевантаження, тим самим у цьому варіанті буде краща інтерактивність, а також оновлення даних у реальному часі.

Нижче наведено переваги використання обраних технологій для розробки програмного забезпечення:

Особливості React:

- Virtual DOM – це концепція програмування, де ідеальне або «віртуальне» представлення інтерфейсу користувача зберігається в пам'яті та синхронізується з «реальним» DOM бібліотекою, такою як ReactDOM.
- Алгоритм реканселяції. React надає декларативний API, який дозволяє не перейматися тим, що саме змінюється при кожному оновленні. [2]
- Модульна будова. Надає можливість збирати проект із компонентів як із конструктора. Компонентам можна змінювати стан, завдяки чому готовий

сайт не має виглядати шаблонним. Плюс такий підхід дозволяє адаптувати елементи під абсолютно будь-який проект і цілі.

- Препроцесор JSX. Формально є сумішю JavaScript та Html, тим самим полегшує та прискорює запис шаблонів, а також виконує оптимізацію при компіляції у JS, що надає йому високу швидкість. [3]

- Велике комьюніті. Означає те, що бібліотека надалі розвиватиметься великими темпами.

Особливості Node.js:

- Швидка обробка запитів та ефективна подієва модель.
- Написаний на C++, використовує «движок» Chrome V8 для компіляції функцій в машинний код, і виконує цю роботу з вражаючою швидкістю.

- Асинхронне оброблення запитів. У контексті серверної частини синхронна обробка передбачає, що код виконується послідовно. Таким чином, кожен новий запит блокує потік інших, а інші команди почнуть виконуватися лише після того, як буде виконано попередню. У свою чергу, асинхронний метод, який використовується в Node.js, максимально використовує однопоточну обробку, що скорочує час відгуку в кілька разів.

- Подієва модель. При використанні однієї мови як на стороні клієнта, так і в бекенді синхронізація відбувається максимально швидко, що особливо корисно для додатків реального часу, заснованих на подіях.

- Ідеальний вибір для мікросервісної архітектури. Цей підхід передбачає розробку одного додатка як набору невеликих сервісів, кожен з яких працює із власними процесами та взаємодіє з легковажними механізмами, часто з API ресурсів HTTP.

- Багата екосистема. NPM – менеджер пакетів Node.js за умовчанням – також є основною платформою для інструментів JavaScript з відкритим вихідним кодом, які відіграють важливу роль у розвитку цієї мови програмування. З урахуванням того, що на даний момент у реєстрі npm доступно близько мільйона бібліотек і щотижня публікується понад 10 000 нових, екосистема Node.js досить багата. За даними статистики, 97% сучасних веб-додатків складаються з модулів npm. [4]

Особливості Sass:

- Sass – це мова таблиць стилів, яка компілюється в CSS.
- Дозволяє використовувати змінні, вкладені правила, функції тощо, з повністю CSS-сумісним синтаксисом.

- Покращує зручність написання стилів замість звичайного CSS.
- Допомагає добре організувати великі таблиці стилів і дозволяє легко ділитися дизайном всередині та між проектами. [5]

Особливості MongoDB:

- Відсутність схеми. MongoDB – це документоорієнтована база даних, в якій одна колекція містить різні документи. Кількість полів, вміст та розмір документів можуть відрізнятися.

- Розширені можливості запитів. MongoDB підтримує динамічні запити документів, що практично не поступається в ефективності SQL.

- Відома структура об'єкта. Легкість масштабування. Не потрібне перетворення об'єктів програми на об'єкти бази даних.
- Використання внутрішньої пам'яті для зберігання віконного робочого набору, забезпечуючи прискорений доступ до даних.

Одним із найважливіших особливостей розробки є використання Атомного підходу. Це методологія, яка пропонує структуру компонентів із різною ієрархією складності та залежностями.

Назва «Атомний підхід» походить від ідеї поділу компонентів на атоми, молекули, організми, шаблони та сторінки, як показано на рис. 1 [6].



Рис. 1. Атомний підхід до розробки веб-додатка

За допомогою такого підходу можна уникнути низки проблем: брак правил стилізації компонентів; значна кількість правил для розробників; невідповідність між компонентами та поглядом на них; код, що повторюється; специфічність компонентів кожної зі сторінок. Тим самим розробка стане зручнішою, простішою і зрозумілішою для подальшої масштабізації.

Висновок. Нові технології веб-програмування дозволяють покращити швидкодію та зручність використання наукових та освітніх порталів, як зі сторони користувачів (абітурієнтів, студентів, викладачів, науковців) так і адміністрації. При стрімкому розвитку надання освітніх послуг в дистанційному on-line форматі, впровадження запропонованих рекомендацій дозволить сформувати ефективний процес навчання та управління комунікаційною політикою в закладах вищої освіти.

ПЕРЕЛІК ПОСИЛАНЬ:

1. Пошукова оптимізація сайту: матеріали Wikipedia [Електронний ресурс]. URL: https://en.wikipedia.org/wiki/Search_engine_optimization (дата звернення: 07.11.2022).
2. Reconciliation: документація React [Електронний ресурс]. URL: <https://reactjs.org/docs/reconciliation.html> (дата звернення: 07.11.2022).
3. React JS - JSX, WebForMySelf, 2018 [Електронний ресурс]. URL: <https://webformyself.com/react-js-jsx/> (дата звернення: 07.11.2022).
4. Introduction to Node.js: документація Node.js [Електронний ресурс]. URL: <https://nodejs.dev/en/learn/> (дата звернення: 07.11.2022).
5. [Sass: Documentation](https://sass-lang.com/documentation/) [Електронний ресурс]. URL: <https://sass-lang.com/documentation/> (дата звернення: 07.11.2022).
6. Matt Rae, Atomic Design Principles & Methodology 101, 2020 [Електронний ресурс]. URL: <https://xd.adobe.com/ideas/process/ui-design/atomic-design-principles-methodology-101/> (дата звернення: 07.11.2022).

ВІДОМОСТІ ПРО АВТОРІВ

Алексєєв Михайло Олександрович – д.т.н., професор, завідувач кафедри програмного забезпечення комп'ютерних систем, НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Алексєєв Олексій Михайлович – к.т.н., доцент, доцент кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Антоненко Єлизавета Олексіївна – магістр групи 124м-21-1 кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Бердник Михайло Геннадійович – д.т.н., доцент, професор кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Бойко Олег Олександрович – к.т.н., старший викладач кафедри кіберфізичних та інформаційно-вимірювальних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Бородай Валерій Анатолійович – к.т.н., доцент, доцент кафедри електропривода НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Випанасенко Станіслав Іванович – д.т.н., професор, професор кафедри електроенергетики НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Воронко Юлія Миколаївна – магістр групи 124м-21-1 кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Воскобойник Євген Костянтинович – асистент кафедри кіберфізичних та інформаційно-вимірювальних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Гаврилко Дмитро Іванович – здобувач ступеня доктора філософії кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Галушко Олег Михайлович – к.т.н., доцент, доцент кафедри безпеки інформації та телекомунікацій НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Ґец Томас – д.ф.-м.н., професор, заступник декана з наукової роботи та роботи з молоддю, Університет Кобленц-Ландау, м. Кобленц, Німеччина.

Гнатушенко Володимир Володимирович – д.т.н., професор, завідувач кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Голінько Олександр Васильович – асистент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Гришак Дмитро Дмитрович – к.т.н., докторант кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Гуліна Ірина Григорівна – к.т.н., доцент, доцент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Демиденко Михайло Андрійович – к.т.н., доцент, доцент кафедри економіки та економічної кібернетики НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Дік Микита Павлович – магістр групи 126м-21-1 кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Денисюк Софія Максимівна – студентка групи 121-21-2 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Дрешпак Наталія Станіславівна – к.т.н., доцент, доцент кафедри електротехніки НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Жабко Олександр Сергійович – магістрант кафедри автоматики та телекомунікацій ДВНЗ «Донецький національний технічний університет», м. Луцьк, Україна.

Желдак Тімур Анатолійович – к.т.н., доцент, завідувач кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Зберовський Олександр Владиславович – д.т.н., професор, завідувач кафедри екології та охорони навколишнього середовища Дніпровського державного технічного університету, м. Кам'янське, Україна.

Зима Віталій Леонідович – студент кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Іванченко Олег Васильович – д.т.н., доцент, професор кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Іванов Денис Валерійович – асистент кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Кабак Леонід Віталійович – к.т.н., доцент, доцент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Каштан Віта Юріївна – к.т.н., доцент, доцент кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Качан Ігор Сергійович – магістр групи 126м-21-1 кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Коваленко Олександр Сергійович – магістр групи 121м-22-1 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Коврайська Валерія Олександрівна – магістр групи 124м-19-1 кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Кожевников Антон Вечеславович – к.т.н., доцент, доцент кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Корнієнко Валерій Іванович – д.т.н., професор, завідувач кафедри безпеки інформації та телекомунікацій НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Коротенко Григорій Михайлович – д.т.н., професор, професор кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Коробко Ольга Валеріївна – асистент кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Коряшкіна Лариса Сергіївна – к.т.н., доцент, доцент кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Кострицька Світлана Іванівна – професор, завідувач кафедри іноземних мов НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Кошеленко Євгеній Валерійович – к.т.н., доцент, доцент кафедри електропривода НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Кремньов Володимир Володимирович – здобувач ступеня доктора філософії кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Крутько Владислав Сергійович – магістр групи 126м-21-1 кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Куваєв Володимир Миколайович – д.т.н., професор, професор кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Кунденко Павло Русланович – магістр групи 122м-21-1 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Кучер Артем Русланович – студент кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Лактіонов Іван Сергійович – д.т.н., доцент, професор кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Ларикова Марія Володимирівна – магістр групи 122м-18-3 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Лось Єлизавета Ігорівна – студентка групи 123-21-1 кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Ляшенко Андрій Євгенович – здобувач ступеня доктора філософії кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Малієнко Андрій Вікторович – к.т.н., доцент, доцент кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Манін Богдан Максимович – студент групи 123-21-2 кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Мартиненко Андрій Анатолійович – старший викладач кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Мацюк Сергій Михайлович – к.т.н., доцент, доцент кафедри безпеки інформації та телекомунікацій НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Мельніков Дмитро Владиславович – студент групи 121-21-1 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Мещеряков Леонид Іванович – д.т.н., професор, професор кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Мороз Борис Іванович – д.т.н., професор, професор кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Мороз Дмитро Максимович – здобувач ступеня доктора філософії кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

М'якенький Арсеній Вячеславович – здобувач ступеня доктора філософії кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Назаренко Костянтин Дмитрович – студент групи 113-22-1 кафедри прикладної математики НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Немченко Максим Ігорович – магістр групи 122м-21-2 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Нестерова Ольга Юріївна – к.т.н., доцент, доцент кафедри електропривода НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Нікулін Сергій Леонідович – д.г.н., професор, професор кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Нікольська Олена Ігорівна – магістр групи 125м-20з-1 кафедри безпеки інформації та телекомунікацій НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Нугуманов Максим Олександрович – студент групи 121-20-2 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Одновол Микола Миколайович – доцент кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Олевський Віктор Ісаакович – д.т.н., професор, професор кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Петрига Марина Вадимівна – бакалавр кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Попова Єлизавета Сергіївна – магістр групи 122м-21-1 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Приходько Віра Володимирівна – к.т.н., доцент, доцент кафедри вищої математики НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Приходченко Сергій Дмитрович – к.т.н., доцент, доцент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Прудченко Анна Юріївна – студентка групи 122-21-1 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Потемпа Михал – магістр-інженер, керівник фірми «GeoBit», Хшанув, Польща.

Потемпа Мачей – студент Сілезький університет, факультет науки і техніки, напрям Комп'ютерні науки, м. Краків, Польща.

Потемпа Войчех – магістр, Технічний університет «Краківська гірничо-металургійна академія ім. Станіслава Сташиця» (AGH), м. Краків, Польща.

Реута Олександр Васильович – к.т.н., доцент, доцент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Рогоза Михайло Валентинович – к.т.н., професор, декан електротехнічного факультету НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Родна Катерина Станиславівна – асистент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Садовий Данило Юрійович – студент групи 122-21-1 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Сарібекян Андрій Арменович – студент групи 122-21-1 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Сергєєва Катерина Леонідівна – к.т.н., доцент, доцент кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Сироткіна Олена Ігорівна – к.т.н, доцент, School of Computer Science University of Windsor, Windsor, Ontario, Kanada.

Скірко Дмитро Юрійович – магістр групи 124м-19-1 кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Слесарєв Володимир Вікторович – д.т.н., професор, професор кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Смигунов Ілля Володимирович – магістр групи 121м-21-1 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Соколова Наталя Олегівна – к.т.н, доцент, доцент кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Соловей Юлія Сергіївна – магістр групи 124м-19-1 кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Спірінцев В'ячеслав Васильович – к.т.н., доцент, доцент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Статива Богдан Олександрович – магістр групи 125м-20-2 кафедри безпеки інформації та телекомунікацій НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Стешенко Анна Андріївна – магістр групи 122м-21-1 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Стефанскі Яцек – д-р електроінженерії, професор, декан факультету електроніки, телекомунікації та інформатики, Польща.

Субач Денис Вячеславович – магістр групи 126м-21-1 кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Ткаченко Сергій Миколайович – к.т.н., доцент, доцент кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Ткаченко Ілля Андрійович – магістр групи 126м-21-1 кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Удовик Ірина Михайлівна – к.т.н., декан факультету інформаційних технологій, професор кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Удовик Олександр Васильєвич – здобувач ступеня доктора філософії кафедри електротехніка та електромеханіка НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Уланова Наталія Петрівна – к.т.н., доцент, доцент кафедри вищої математики НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Федоров Сергій Іванович – старший викладач кафедри електропривода НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Футтерлинг В. – інженер-програміст Haas Schleifmaschinen GmbH, м. Троссінген, Німеччина.

Хабарлак Костянтин Сергійович – аспірант кафедри системного аналізу і управління НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Харь Альона Тарасівна – асистент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Чарнул Павел – д-р. хабілітований, професор, зам. декана із співробітництва та промоції, факультет електроніки, телекомунікації та інформатики, Польща.

Чумичов Денис Дмитрович – асистент кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Чуріканова Олена Юріївна – д.е.н., доцент, завідувач кафедри економіки та економічної кібернетики НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Швачич Генадій Григорович – д.т.н., професор, професор кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Шевцова Ольга Сергіївна – асистент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Шевченко Владислав Олексійович – магістр групи 126м-21-1 кафедри інформаційних технологій та комп'ютерної інженерії НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Щербина Павло Олександрович – асистент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Ширін Артем Леонідович – к.т.н., доцент, доцент кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Шолойко Софія Андріївна – студентка групи 122-21-2 кафедри програмного забезпечення комп'ютерних систем НТУ «Дніпровська політехніка», м. Дніпро, Україна.

Шчешневскій Патрик – магістр, Технічний університет «Краківська гірничо-металургійна академія ім. Станіслава Сташиця» (AGH), м. Краків, Польща.

ЗМІСТ

Розділ 1 МІЖНАРОДНЕ СПІВРОБІТНИЦТВО У СФЕРІ ОСВІТИ, НАУКИ І ВИРОБНИЦТВА.....	3
1. Кабак Л.В., Кострицька С.І., Ларикова М.В., Потемпа В. Хорс as «universal soldier» on the big data «battlefield».....	3
2. Алексєєв М.О., Алексєєв О.М., Кремньов В.В., Шчешневскій П. Контроль параметрів об'єктів керування з використанням перебудованих матричних спектральних операторів.....	7
3. Удовик І.М., Мещеряков Л.І., Попова Є.С., Сироткіна Е.І. Дослідження ефективності фреймворку Flutter при розробці високодинамічних мобільних додатків.....	10
4. Антоненко Є.О., Коряшкіна Л.С., Гец Т. Математичне моделювання епідемії з урахуванням мобільності.....	14
5. Спирінцев В.В., Ширін А.Л., Сарібекян А.А., Потемпа Мачей Використання React як фреймворку для розробки веб-утиліт.....	16
6. Футтерлинг В., Родна К.С., Ларикова М.В. ImGui. Dear ImGui: library for apps in C++.....	20
Розділ 2 ЗАСТОСУВАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ У СФЕРІ ОСВІТИ, НАУКИ І УПРАВЛІННЯ ВИРОБНИЦТВОМ.....	24
7. Гнатушенко В.В., Грищак Д.Д., Шевцова О.С. Розпізнавання зелених насаджень із застосуванням геоінформаційних технологій.....	24
8. Субач Д.В., Ткаченко І.А., Коротенко Г.М., Гнатушенко В.В. Розробка інформаційної технології створення та використання компонентів NFT маркетплейса.....	27
9. Олевський В.І., Крутько В.С. Розробка підсистеми візуальної аутентифікації користувача на основі використання бібліотек OpenCV та Face Recognition.....	30
10. Манін Б.М., Лось Є.І., Каштан В.Ю., Гнатушенко В.В. Розробка системи керування комбінованого велотренажера для реабілітації військових службовців.....	35
11. Мещеряков Л.І., Ширін А.Л., Галушко О.М., Слесарєв В.В., Шолойко С.А. Побудова правильної топологічної сітки об'єктів та модифікатор Retopology в Autodesk 3D Studio MAX.....	38
12. Малієнко А.В. Аналіз сучасної платформи інформаційних потоків та методів управління проектами в умовах розвитку циклічної економіки....	43
13. Рогоза М.В., Бородай В.А., Нестерова О.Ю., Кошеленко Є.В., Федоров С.І. Швидкісне обслуговування тягових акумуляторів сучасних автотранспортних засобів	46
14. Чумичов Д.Д. Огляд технології інтернету речей та його ключових компонентів.....	48
15. Ткаченко С.М. Обґрунтування технічного рішення для систем цифрової	

термометрії зерна на основі контролера Arduino.....	52
16. Малієнко А.В., Воронко Ю.М. Аналіз та оптимізація системи інтернет торгівлі в умовах сучасного розвитку ринку ювелірних виробів.....	62
Розділ 3 ПРОГРАМНІ ЗАСОБИ УПРАВЛІННЯ, ЗБОРУ, ОБРОБКИ І ПЕРЕДАЧІ ІНФОРМАЦІЇ.....	67
17. Нікулін С.Л., Каштан В.Ю., Сергєєва К.Л., Коробко О.В., Дік М.П. Вплив попередньої обробки супутникових знімків різними методами на виділення лінеаментів природного походження.....	67
18. Сергєєва К.Л., Каштан В.Ю., Коробко О.В., Іванов Д.В., Качан І.С. Аналіз супутникових даних для оцінки островів тепла урбанізованих територій.....	71
19. Спірінцев В.В., Ширін А.Л., Гуліна І.Г., Прудченко А.Ю. Дослідження можливостей створення дизайну інтерфейсів користувачів веб-сайтів у середовищі Figma.....	76
20. Петрига М.В., Соколова Н.О. Розробка веб-орієнтованої інформаційної системи тайм менеджменту з використанням стеку технологій MERN.....	79
21. Мещеряков Л.І., Випанасенко С.І., Дрешпак Н.С., Кунденко П.Р. Дослідження ефективності фреймворку React Native при розробці мобільних додатків.....	83
22. Сергєєва К.Л., Нікулін С.Л., Каштан В.Ю., Коробко О.В., Шевченко В.О. Дослідження особливостей виділення границь яскравості природних об'єктів на різномасштабних космознімках.....	90
23. Мороз Б.І., Швачич Г.Г., Мороз Д.М., Смигунов І.В. Розробка та дослідження ефективності впровадження програмного забезпечення для управління персоналом з використанням HRM системи на базі автоматизованих ключових HR-процесів.....	95
24. Мещеряков Л.І., Зберовський О.В., Ширін А.Л., Денисюк С.М. Потенціал використання об'єктів типу Loft у середовищі Autodesk 3D Studio MAX.....	101
25. Приходченко С.Д., Нугуманов М.О., Іванченко О.В., Голінько О.В. Особливості застосування алгоритмів типу «розділяй і володій» в галузі Data Science.....	105
26. Бердник М.Г., Коваленко О.С. Програмне забезпечення для моделювання оптимального раціону харчування.....	113
27. Алексєєв М.О., Алексєєв О.М., Ляшенко А.Є. Контроль параметрів роторних об'єктів з використанням частотно-хвильового спектрального аналізу.....	115
28. Желдак Т.А., Коврайська В.О. Розробки інформаційної системи контролю за поведінням з відходами.....	118
29. Спірінцев В.В., Ширін А.Л., Мартиненко А.А., Шолойко С.А. Дослідження фреймворку Vue.js.....	123
30. Желдак Т.А., Скірко Д.Ю. Бізнес-аналіз вимог до інформаційної архітектури фітнес додатку та його проектування.....	127

Розділ 4 МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ ТА ОПТИМІЗАЦІЯ СИСТЕМ ОБРОБКИ ІНФОРМАЦІЇ ДЛЯ ВИРІШЕННЯ ЗАВДАНЬ ОСВІТИ, НАУКИ І УПРАВЛІННЯ ВИРОБНИЦТВОМ.....	133
31. Приходько В.В., Уланова Н.П., Назаренко К.Д. Математичне моделювання геомеханічних процесів з використанням метода граничних елементів.....	133
32. Мещеряков Л.І., Одновол Н.Н., Удовик О.В., Немченко М.І. Розробка додатку для аналітики користувачьких даних та побудови скорингових моделей.....	137
33. Каштан В.Ю., Зима В.Л. Методика картографування пожеж на цифрових знімках з використанням машинного навчання.....	143
34. Алексєєв М.О., Алексєєв О.М., Гаврилко Д.І. Класифікація сигналів об'єктів керування з використанням перебудованих матричних спектральних операторів.....	149
35. Лактіонов І.С., Жабко О.С. Аналіз функціональних можливостей пакетів прикладних програм моделювання бездротових сенсорних мереж.....	152
36. Демиденко М.А., Чуріканова О.Ю. Модель оптимізації маркетингових компаній в пошукових системах.....	158
37. Каштан В.Ю., Сергєєва К.Л., Коробко О.В., Іванов Д.В., Кучер А.Р. Методика автоматизованого регулювання дорожнього руху на основі нейронної мережі.....	164
38. Хабарлак К.С. Навчання адаптивної нейронної мережі для сегментації зображень.....	169
39. Куваєв В.М., Бойко О.О., Воскобойник Є.К. Відтворення синтезованої моделі системи керування на базі промислового програмного забезпечення.....	171
Розділ 5 ІНФОРМАЦІЙНА БЕЗПЕКА.....	174
40. Корнієнко В.І., Статива Б.О. Вдосконалення методу захисту хмарних даних із контролем ліній зв'язку інформаційно-комунікаційних систем...	174
41. Мещеряков Л.І., Кожевников А.В., Мацюк С.М., Стешенко А.А. Аналіз безпечності та швидкості сучасних алгоритмів шифрування.....	178
42. Корнієнко В.І., Нікольська О.І. Розробка організаційно-технічних заходів захисту інформації при використанні цифрової АТС EWSD.....	182
43. Желдак Т.А., Соловей Ю.С. Бізнес-аналіз додатка для ідентифікації кіберзлочинів у банківській сфері.....	188
Розділ 6 ПРОБЛЕМИ ДИСТАНЦІЙНОЇ ОСВІТИ.....	193
44. Алексєєв М.О., М'якенький А.В., Алексєєв О.М. Модель оцінки когнітивних навичок студента.....	193
45. Харь А.Т., Куваєв В.Н., Реута О.В., Мельніков Д.В. Дистанційна освіта під час війни.....	197
46. Спірінцев В.В., Ширін А.Л., Щербина П.О., Садовий Д.Ю. Оновлення існуючих веб-ресурсів закладів вищої освіти як вдосконалення надання освітніх послуг.....	201
ВІДОМОСТІ ПРО АВТОРІВ.....	205

Наукове видання

**ПРОБЛЕМИ ВИКОРИСТАННЯ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ У
СФЕРІ ОСВІТИ, НАУКИ ТА ПРОМИСЛОВОСТІ**

XVII Міжнародна конференція

Збірник наукових праць
№7
(Українською мовою)

Відповідальний за випуск Л.І. Мещеряков

Видано в редакції авторів публікацій

Електронний ресурс

Підготовлено і видано
в НТУ «Дніпровська політехніка».
Свідоцтво про внесення до Державного реєстру ДК №1842 від 11.06.2004 р.
49005, м.Дніпро, просп. Д. Яворницького, 19