

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
ТЕРНОПІЛЬСЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ІВАНА ПУЛЮЯ**

**МАТЕРІАЛИ**

**X НАУКОВО-ТЕХНІЧНОЇ КОНФЕРЕНЦІЇ**

**«ІНФОРМАЦІЙНІ МОДЕЛІ,  
СИСТЕМИ ТА ТЕХНОЛОГІЇ»**



**7–8 грудня 2022 року**

**ТЕРНОПІЛЬ  
2022**

## ПРОГРАМНИЙ КОМІТЕТ

**Голова:** Сергій Лупенко– докт. техн. наук, професор.

**Співголови:** Павло Марущак– докт. техн. наук, професор, проректор з наукової роботи.  
Ігор Баран– канд. техн. наук, доцент, декан факультету ФІС.

**Науковий секретар:** Галина Семенишин– старший викладач.

**Члени:** докт. фіз.-мат. наук, професор Василь Кривень; докт. техн. наук, професор Ярослав Литвиненко; докт. техн. наук, професор Микола Карпінський; докт. фіз.-мат. наук, професор Михайло Петрик; канд. техн. наук, доцент Галина Осухівська; канд. пед. наук, доцент Жанна Баб'як; канд. техн. наук, доцент Наталія Загородна.

## ОРГАНІЗАЦІЙНИЙ КОМІТЕТ

**Голова:** Юрій Скоренький– канд. фіз.-мат. наук, доцент, завідувач кафедри фізики

**Члени:** канд. техн. наук, доцент Вячеслав Никитюк; канд. техн. наук, доцент Дмитро Михалик; канд. техн. наук, асистент Марія Стадник; асистент Наталія Шаблій; ст. викладач Ліліана Джиджора.

Матеріали X науково-технічної конфіції «Інформаційні моделі, системи та технології»  
М34 Тернопільського національного технічного університету імені Івана Пулюя,  
(Тернопіль, 7–8 грудня 2022 р.). – Тернопіль : Тернопільський національний технічний  
університет імені Івана Пулюя, 2022. –162 с.

**Адреса оргкомітету:** ТНТУ ім. І. Пулюя, м. Тернопіль, вул. Руська, 56, 46001,  
тел. (0352) 52-41-33, факс (0352) 254983.

E-mail: confis2022@gmail.com

Редагування, оформлення та верстка: Галина Семенишин

## СЕКЦІЇ КОНФЕРЕНЦІЇ, ЯКІ ПРЕДСТВЛЕНІ В ЗБІРНИКУ

- Математичне моделювання;
- Інформаційні системи та технології;
- Комп'ютерні системи та мережі;
- Програмна інженерія та моделювання складних розподілених систем;
- Новітні фізико-технічні та освітні технології.

В збірнику надруковано тези доповідей IX науково-технічної конференції «Інформаційні моделі, системи та технології» (Тернопіль, 7–8 грудня 2022 р.) за такими науковими напрямками: математичне моделювання; інформаційні системи та технології; комп'ютерні системи та мережі; програмна інженерія та моделювання складних розподілених систем; новітні фізико-технічні та освітні технології.

Розрахований на науковців, викладачів та студентів вузів.

**За зміст тез та дотримання норм академічної доброчесності відповідальність несе автор.**

## СЕКЦІЯ 1. МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ

УДК 004.4

**А. Кашосі, О. Кишкевич, Наталія Загородна**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

### УПРАВЛІННЯ ЯКІСТЮ ДАНИХ В ПРОЦЕСІ ЕТЛ В УМОВАХ РЕСУРСНИХ ОБМЕЖЕНЬ

UDC 004.4

**A. Kashosi, O. Kyshkevych, N. Zagorodna**

### DATA QUALITY MANAGEMENT IN ETL PROCESS UNDER RESOURCE CONSTRAINTS

It is impossible to picture today's firms' marketing success without Big Data. Data quality (DQ) evaluation in Big Data operations provides unique evaluation challenges. Several research [1, 2] have attempted to provide a more exact description of this term, which represents information assets with high volume, velocity, and variability that require particular technology and analytical approaches to convert into value.

A data integration phase is required inside the BI system for a variety of reasons, including heterogeneous formats, data formats that are difficult to comprehend or ambiguous, legacy systems that employ antiquated databases, and the structure of the data source that varies over time. All these data source factors make DQ unreliable. [3]

Dakrory et al. conducted research [4] to automate testing procedures that validate Data Quality Dimensions (DQD) such as completeness, consistency, uniqueness, validity, timeliness, and accuracy. The goal of their study was to track DQ in ETL to ensure the correctness of ETL methods and assess whether they need to be changed to solve data problems.

Moreover, the study in [4] leads to a more realistic representation and classification of DQ tests in ETL methods.

- **Completeness:** ensures that all relevant data is stored, which includes checking the number of records, duplicates, integrity constraints, and data boundaries.
- **Consistency:** ensures that all values are consistent across all datasets. This includes field mapping, integrity constraints, aggregation of metrics, and hierarchy-level integrity checks.
- **Uniqueness:** ensures that there are no duplicates in the stored data.
- **Validity:** Ensures that the data conforms to the syntax (format, type, range) of its description, and its implementation includes integrity constraint checking, field data type checking, and field length checking.
- **Timeliness:** ensures that all data is stored within the specified period. Data access and freshness are the tests that are applied.
- **Correctness:** ensures that all data accurately describes the «real object or event being described» Field-to-field comparison, data boundaries, and integrity constraints are all difficult to implement.

The implementation of DQ in an ETL process dealing with Big Data faces a problem related to the nature of the data to be validated, which presents characteristics such as «volume», «velocity», and «variety», to name a few, as Nagham and Laden argued in their research [5] that the number of Vs in Big Data varies by sector.

Volume is a particular challenge when developing tests to validate the quality of the dataset going through the ETL process. As a result, we have explored a sampling strategy that allows us to obtain a representative DQ result without having to process each data item.

Sampling has been successfully applied in areas such as network traffic measurement [7], where it has been shown to improve memory allocation and analysis performance.

To this end, we analyze the type of system that uses Big Data in this study and examine processes in batch rather than in stream. This is significant because it determines the type of sampling method to be used. We used stratified random sampling in this study.

Stratified random sampling divides the sampling units of the population into homogeneous groups called strata, and then draws a simple random sample within each stratum.

Strata are determined by information other than the attributes being assessed that is known or assumed to vary with the attributes of interest. Stratifying the population into homogeneous groups of sampling units minimizes sampling error; estimates obtained within strata are more accurate than simple random samples drawn from the same population. [6]

In this study, we consider the model of a data management system in the Python programming language. One of the many microservices manages the data integration process with various Customer Relationship Management (CRM) data sources, namely ZIP, TXT, CLS, XLS and XLSX files. The process of extracting, transforming, and loading is done sequentially, followed by the final transformation processes, i.e., sampling and performing the DQD tests and generating the DQ assessment reports.

Algorithm:

1. Divide the entire population into non-overlapping strata
2. Select a simple random sample from within each stratum

$L$  = number of strata

$N_i$  = number of sample units within stratum  $i$

$N$  = number of sample units in the population

The population mean ( $\mu$ ) is estimated with:

$$\hat{\mu} = \frac{1}{N} (N_1 \hat{\mu}_1 + N_2 \hat{\mu}_2 + \dots + N_L \hat{\mu}_L) = \frac{1}{N} \sum_{i=1}^L N_i \hat{\mu}_i$$

where  $N_i$  represents the total number of sample units in stratum  $i$ ,  $L$  represents the number of strata, and  $N$  represents the total number of sample units in the overall population.

Variance of the estimate is just the weighted average of estimated variances of the mean from a sequence of random samples selected from stratum  $i$  through  $L$ , although it appears to be a little more complicated:

$$\hat{\text{var}}(\hat{\mu}) = \frac{1}{N^2} \left[ N_1^2 \left( \frac{N_1 - n_1}{N_1} \right) \left( \frac{s_1^2}{n_1} \right) + \dots + N_L^2 \left( \frac{N_L - n_L}{N_L} \right) \left( \frac{s_L^2}{n_L} \right) \right] = \frac{1}{N^2} \sum_{i=1}^L N_i^2 \left( \frac{N_i - n_i}{N_i} \right) \left( \frac{s_i^2}{n_i} \right)$$

And  $s_i^2$  is an estimate of the overall population variance from each stratum  $i$  through  $L$ .

Similarly, estimating the proportion of the population with a particular trait ( $p$ ) using stratified random sampling involves combining estimates from multiple simple random samples, each generated within a stratum. The population proportion is estimated with the sample proportion:

$$\hat{p} = N_1 \hat{p}_1 + N_2 \hat{p}_2 + \dots + N_L \hat{p}_L = \sum_{i=1}^L N_i \hat{p}_i$$

Variance of the estimate  $\hat{p}$  is:

$$\hat{\text{var}}(\hat{p}) = \frac{1}{N^2} \sum_{i=1}^L N_i^2 \hat{\text{var}}(\hat{p}_i) = \frac{1}{N^2} \sum_{i=1}^L N_i^2 \left( \frac{N_i - n_i}{N_i} \right) \left( \frac{\hat{p}_i (1 - \hat{p}_i)}{n_i - 1} \right)$$

Standard error of  $\hat{p}$  is the square root of  $\hat{\text{var}}(\hat{p})$ .

Using stratified random sampling necessitates deciding how to distribute a set amount of work among the many strata. There are several methods for assigning sampling effort. Data on variability within each stratum, as well as the relative cost of acquiring and testing a sample unit, aid in increasing overall sampling efficiency [6].

### References

1. De Mauro, A., Greco, M., & Grimaldi, M. (2015). What is big data? A consensual definition and a review of key research topics. B AIP Conference Proceedings. INTERNATIONAL CONFERENCE ON INTEGRATED INFORMATION (IC-ININFO 2014): Proceedings of the 4th International Conference on Integrated Information. AIP Publishing LLC. URL: <https://doi.org/10.1063/1.4907823>.
2. Udofia, E., Buduka, S., Akpabio, J., Egwu, S., Udofia, E., & Olagunju, D. (2020). Digital Transformation: After the Big Data, What Next? B Day 1 Tue, August 11, 2020. SPE Nigeria Annual International Conference and Exhibition. SPE. URL: <https://doi.org/10.2118/203614-ms>.
3. Souibgui, M., Atigui, F., Zammali, S., Cherfi, S., & Yahia, S. B. (2019). Data quality in ETL process: A preliminary study. B Procedia Computer Science (Вип. 159, с. 676–687). Elsevier BV. URL: <https://doi.org/10.1016/j.procs.2019.09.223>.
4. B., S., M., T., & A., A. (2015). Automated ETL Testing on the Data Quality of a Data Warehouse. B International Journal of Computer Applications (Вип. 131, Issue 16, с. 9–16). Foundation of Computer Science. URL: <https://doi.org/10.5120/ijca2015907590>.
5. Saeed, N., & Husamaldin, L. (2021). Big Data Characteristics (V's) in Industry. B Iraqi Journal of Industrial Research (Вип. 8, Issue 1, с. 1–9). Corporation of Research and Industrial Development. URL: <https://doi.org/10.53523/ijoirvol8i1id52>.
6. Stratified Random Sampling. (n.d.). [cals.arizona.edu](https://cals.arizona.edu). Retrieved December 1, 2022. URL: <https://cals.arizona.edu/>.
7. Cormode, G., & Duffield, N. (2014). Sampling for big data. B Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. KDD '14: The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM. URL: <https://doi.org/10.1145/2623330.2630811>.

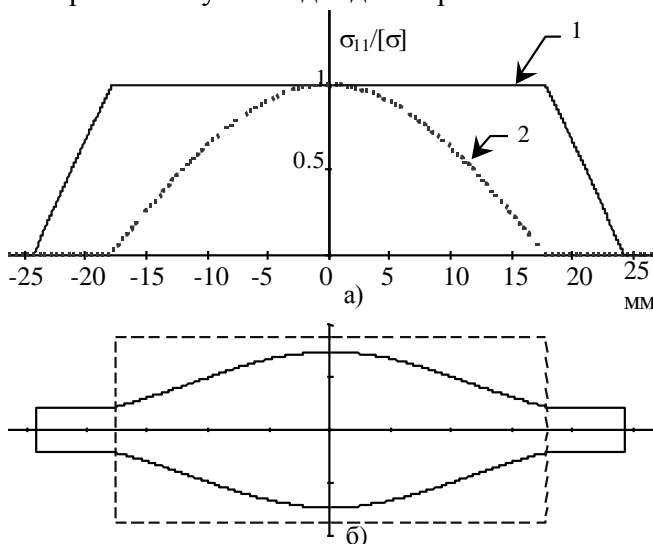
**ОПТИМІЗАЦІЯ ФОРМИ П'ЄЗОЕЛЕКТРИЧНОГО ТРАНСФОРМАТОРА**

**SHAPE OPTIMIZATION OF PIEZOELECTRIC TRANSFORMER**

Розглядається задача оптимізації форми плоского п'єзотрансформатора струму (ПТ) з поляризацією за товщиною пластини, що має товщину  $h$ , а її середня площина співпадає з площиною  $xOy$ . Матеріал має густину  $\rho$ . Припустимо, що бічні поверхні п'єзотрансформатора вільні від електродів, а верхня і нижня поверхні покриті системою електродів, зазор між якими наближається до 0. Для зменшення втрат енергії п'єзотрансформатор звичайно закріплюють так, щоб його поверхні не передавали зусилля на закріплення. Така умова приводить до граничної умови:  $\sigma_{ij}n_j = 0$ , де  $n_j$  - вектор зовнішньої нормалі. У випадку одномірних коливань з коловою частотою  $\omega$  по довжині (координаті  $x$ ) при змінній ширині  $b(x)$  п'єзотрансформатора та симетрії ПТ відносно осі  $Ox$  рівняння, що описують його, можуть бути записані у вигляді:

$$\frac{d}{dx}(b(x)\sigma_{11}) + \rho\omega^2 b u_1 = 0; \quad \frac{d}{dx} u_1 = s_{11}\sigma_{11} + \frac{d_{31}}{h \cdot b(x)} \int_{-b/2}^{b/2} \varphi(x, y) dy;$$

де  $\varphi(x, y)$  – різниця потенціалів між верхнім і нижнім електродами ПТ. П'єзотрансформатор найчастіше працює у режимі, близькому до резонансу, тому вважаємо, що розподіл напружень у трансформаторі такий самий, як при власній формі коливань. Тоді система спроститься і може бути переписана у вигляді одного рівняння:



**Рисинок 1.** Розподіл усереднених по ширині напружень у матеріалі п'єзотрансформатора у долях від  $[\sigma]$  (а), пропонується форма п'єзотрансформатора (б)

$$\frac{d}{dx} \left( \frac{1}{b(x)} \frac{d}{dx} (b(x)\sigma_{11}) \right) + \rho\omega^2 s_{11}\sigma_{11} = 0$$

Задаючись розподілом напружень  $\sigma_{11} = [\sigma]\sigma_0$  отримаємо необхідну форму бічної поверхні, котра реалізує його при заданій частоті  $\omega$ .

$$b(x) = \frac{B}{\sigma_0} \exp \left( - \int_0^x \left( \frac{\rho\omega^2 s_{11} \int_0^x \sigma_0 dx}{\sigma_0} \right) + \int_0^x \frac{A}{\sigma_0} dx \right).$$

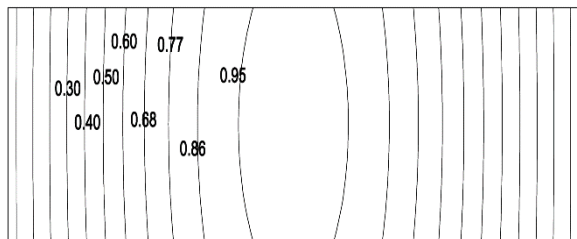
У випадку оптимальної форми ПТ  $\sigma_{11}$  наближено рівні  $[\sigma]$ , тому:

$$b(x) = B \exp \left( - \frac{\rho\omega^2 s_{11} x^2}{2} + A \cdot x \right).$$

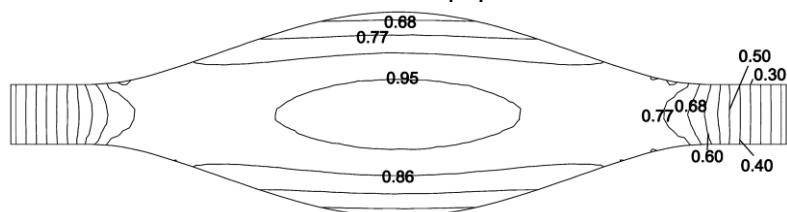
Невідома стала  $A$  лише призводить до переміщення п'єзо-трансформатора по осі  $x$ , а при  $A = 0$  невідома стала  $B$  визначається необхідною потужністю ПТ і

дорівнює максимальній ширині п'єзотрансформатора. Однак, ця оптимальна форма п'єзопластини має мати нескінчену довжину. Але, так як ширина у кінцевих частинах мала, то суттєвого впливу ці частини не мають. Тому, якщо примусово задати що при координаті  $x$

більшій за модулем ніж деяке значення  $l_e$  ширина  $b(x)$  є сталою і рівною  $b(l_e)$ , а також задати, що



**Рисунок 2.** Розподіл  $\sigma_{11}$  по ширині п'єзотрансформатора «звичайної» форми



**Рисунок 3.** Розподіл  $\sigma_{11}$  по ширині п'єзотрансформатора форми, що пропонується

довжина такої ділянки є достатньою, щоб напруження у матеріалі спали до нуля – отримана форма ПТ буде близькою до оптимальної і тим ближче до оптимальної, чим більше  $l_e$ . Оберемо довжину  $l_e$  рівною довжині ПТ, класичної форми, котрий має таку саме резонансну частоту.

Розподіл усереднених за шириною напружень для п'єзотрансформатора із матеріалу PZT-4 із робочою частотою 47.3 кГц зображений на рис. 1 а. Ескіз відповідної форми п'єзотрансформатора

представлений на рис. 1 б, причому прийнято, що  $l_e = \frac{1}{4 \cdot f \sqrt{\rho \cdot s_{11}}}$ . На рисунку пунктирними

лініями зображено п'єзотрансформатор «класичної» форми поперечно-поперечного типу із таким самим значенням електричних параметрів у околі частоти резонансу. Із графіка видно, що напруження у матеріалі пропонованого п'єзотрансформатора близькі до максимально допустимих практично по всьому матеріалі ПТ, крім прикінцевих областей, вклад котрих у загальну роботу матеріалу не є значним.

Для підтвердження розвинутих положень було проведено двовимірне моделювання п'єзотрансформатора пропонованої форми за допомогою програми FreeFem++. Для моделювання використовувався функціонал, що описує власну форму коливань п'єзотрансформатора у наступній формі:

$$J(u, v) = \int_S \left( \frac{\partial}{\partial x} v_1 \right) \cdot \left( c_{11} \cdot \frac{\partial}{\partial x} u_1 + c_{12} \cdot \frac{\partial}{\partial y} u_2 \right) + \left( \frac{\partial}{\partial y} v_2 \right) \cdot \left( c_{12} \cdot \frac{\partial}{\partial x} u_1 + c_{11} \cdot \frac{\partial}{\partial y} u_2 \right) + \left( \frac{\partial}{\partial x} v_2 + \frac{\partial}{\partial y} v_1 \right) \cdot (c_{66}) \cdot \left( \frac{\partial}{\partial x} u_2 + \frac{\partial}{\partial y} u_1 \right) + \rho \omega^2 \lambda (u_1 \cdot v_1 + u_2 \cdot v_2) ds,$$

де  $v$  – пробні переміщення.

Дискретизація області проводилась за допомогою методу скінчених елементів із використанням базисних функцій другого порядку, будувалась відповідні глобальні матриці і розв'язувалась стандартна задача відносно власної форми коливань. Потім, на основі розрахованих форм власних переміщень, отримувались значення напружень у матеріалі, котрі відразу нормувались так, щоб максимальні напруження були не більші за  $[\sigma]$ . Для порівняння, паралельно проводилось моделювання п'єзотрансформатора звичайної форми. Розподіл найвищих напружень у матеріалі ( $\sigma_{11}$ ) поданий на рисунках 2 та 3.

Із порівняння рисунків видно, що напруження у п'єзотрансформаторі, який пропонується, розподілені значно рівномірніше і досить точно описуються одновимірним наближенням. Більш рівномірний розподіл напружень дозволяє отримати вищий коефіцієнт використання матеріалу, покращити техніко-економічні показники п'єзотрансформаторів та знизити їх матеріаломісткість.

**БАГАТОВИМІРНІ УМОВНІ ЛІНІЙНІ ВИПАДКОВІ ПРОЦЕСИ****MULTIVARIATE CONDITIONAL LINEAR RANDOM PROCESSES**

Об'єктом дослідження, результати якого представлено у доповіді є процес математичного моделювання багатовимірного випадкового сигналу, який за структурою свого породження являє собою суму великого числа випадкових імпульсів, що виникають у випадкові моменти часу. Прикладами стохастичних сигналів такого типу можуть бути, зокрема, електроенцефалографічні та кардіографічні сигнали, фотоплетизмограми [1], процеси ресурсоспоживання (електро-, газо-, водоспоживання) [2], радіолокаційні сигнали, вібрації підшипників електричних машин [3, 4] та ін.

Поширеною у прикладних задачах математичною моделлю (зокрема, і у багатовимірному випадку) такого типу сигналів є лінійний випадковий процес, що допускає зображення сигналу у вигляді суми великого числа стохастично незалежних випадкових імпульсів, які виникають у пуассонівські моменти часу [3, 4]. Якщо ж імпульси є стохастично залежними (або моменти часу їх появи не є пуассонівськими), то математичною моделлю буде умовний лінійний випадковий процес [1, 2]. Означення та аналіз ймовірнісних властивостей таких процесів для багатовимірного випадку, на сьогодні, не проведено.

У доповіді наведено означення багатовимірного умовного лінійного випадкового процесу, кожна компонента якого зображена у вигляді стохастичного інтеграла від випадкового ядра за процесом із незалежними приростами. Отримано вирази для характеристичної функції та моментних функцій означеного процесу. Використовуваний підхід полягав у застосуванні математичного апарату умовних характеристичних функцій, а також відомого зображення безмежно подільної характеристичної функції лінійного випадкового процесу, як функціоналу від процесу з незалежними приростами.

Завдяки отриманим результатам забезпечується можливість проведення теоретичного аналізу ймовірнісних властивостей багатоканальних стохастичних сигналів, математичною моделлю яких є багатовимірний умовний лінійний випадковий процес, зокрема здійснення обґрунтування їх властивостей стаціонарності чи стохастичної періодичності, які для такої моделі є наслідком відповідних особливостей ядра та породжуючого процесу з незалежними приростами.

**Література**

1. M. Fryz, B. Mlynko, O. Mul, N. Zagorodna, Conditional Linear Periodical Random Process as a Mathematical Model of Photoplethysmographic Signal, *Scientific Journal of Riga Technical University*. Vol. 45. 2010. P. 82–86.
2. M. Fryz, L. Scherbak, Statistical analysis of random coefficient periodic autoregression and its application for short-term electricity consumption forecasting, *Technical Electrodynamics*. No. 2. 2019. P. 38–47.
3. М. Е. Фриз, Л. Н. Щербак, Эргодические свойства линейных процессов в задачах математического моделирования и статистического анализа случайных сигналов, *Электронное моделирование*. Том 32. Вып. 1. 2010. С. 3–14.
4. M. Fryz, Mixing property and ergodicity of linear random processes, *Proceedings of the IEEE International Workshop on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications*, Rende (Cosenza), Italy, 2009, p. 343–346.



## СЕКЦІЯ 2. ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ, КІБЕРБЕЗПЕКА

УДК 004.056

**А. Анпілогов**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

### ДОДАТКОВІ ЗАСОБИ ЗАХИСТУ БАЗИ МЕТАДАНИХ РЕЄСТРУ ІНФОРМАЦІЙНИХ РЕСУРСІВ

UDC 004.056

**A. Anpilohov**

### ADDITIONAL MEANS OF PROTECTION OF THE METADATA BASE OF THE REGISTER OF INFORMATION RESOURCES

Метадані – це дані про дані. Іншими словами, це інформація, яка використовується для опису даних, які містяться в щось на зразок веб-сторінки, документа або файлу. Інший спосіб думати про метадані - це як коротке пояснення або підсумок того, що є даними. Простий приклад метаданих для документа може включати збір інформації, такої як автор, розмір файлу, дата створення документа та ключові слова для опису документа.

На момент сьогоднішнього використання баз даних є однією з характерних рис більшості сучасних інформаційних систем. В основі суті поняття «база даних» є те, без чого не можлива і навколо чого будуються інформаційні системи будь-яких підприємств. Можливо саме тому практиці використання баз даних та теорії їх створення приділяється досить велика увага протягом усього періоду функціонування інформаційних систем.

Протягом досить довгого періоду часу основним типом баз даних вважалися реляційні, які сьогодні вже сприймаються як класичні. Проте стрімкий розвиток інформаційних систем поставив завдання перед сучасними базами даних, вирішення яких потребує більш широких можливостей, ніж можуть запропонувати реляційні бази даних. Окрім класичних завдань, на сучасні бази даних покладаються завдання забезпечення багатомашинної обробки та зберігання великого об'єму інформації, інтеграції з мережею Інтернет, оперативного аналізу даних, розмежування доступу користувачів до інформації що зберігається, захисту інформації під час передачі її по мережі. Для забезпечення комфортної роботи з базами даних сьогодні існують різноманітні системи управління базами даних (СУБД).

СУБД по суті являє собою комплекс програм, що дозволяють створити базу даних (БД) і проводити маніпуляції над даними (оновлювати, вставляти, видаляти і вибирати). Система забезпечує безпеку, надійність зберігання і цілісність даних, а також надає кошти для адміністрування БД. Однак, наскільки всім добре відомо, «Жодна система не є ідеальною» і будь-яка СУБД не є виключенням.

#### Література

1. Що таке метадані? URL: <https://uk.go-travels.com/85100-metadata-definition-and-examples-1019177-9528130>.
2. Работа с метаданными. URL: <https://habr.com/ru/post/93119/>.
3. Метадані URL: <https://uk.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%B0%D0%B4%D0%B0%D0%BD%D1%96>.

УДК 004.031.6

**О. Багрій**

(Тернопільський національний технічний університет імені Івана Пулюя)

## **ОСОБЛИВОСТІ РЕАЛІЗАЦІЇ ЗБОРУ ТА ОПРАЦЮВАННЯ ДАНИХ ДЛЯ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ НА ОСНОВІ КЛАВАТУРНОГО ПОЧЕРКУ**

UDC 004.031.6

**O. Bagriy**

### **FEATURES OF IMPLEMENTATION OF DATA COLLECTION AND PROCESSING FOR KEYBOARD-BASED USER AUTHENTICATION**

Оскільки динамічна автентифікація користувачів за клавіатурним почерком може широко застосовуватися і на web-сайтах, і в самостійних локальних додатках, у відомих рішеннях, розглядається як здійснюваний web-браузером web-збір даних, що характеризують динаміку роботи користувачів з клавіатурою, і локальний збір даних засобами операційної системи комп'ютера.

Технології, що використовуються для web-збирання даних, що характеризують динаміку роботи користувача з клавіатурою, можна поділити на дві категорії залежно від використання плагінів – незалежних програмних модулів, динамічно підключаються до web-браузера і призначені для розширення його функціональних можливостей. До програмних засобів, що не використовують плагіни при web-зборі, відносяться вбудовувані в web-сторінки JavaScript-програми, а також розширення веб-браузера. До програмних засобів, робота яких ґрунтується на використанні у системі спеціальних плагінів, відносяться такі широко відомі технології як Flash та Java-аплети. Взаємодія web-браузера з плагіном здійснюється через API-інтерфейс. Серед найпопулярніших API-інтерфейсів виділяють NPAPI (Netscape Plugin Application Programming Interface), PPAPI (Pepper Plugin Application Programming Interface) та ActiveX. Проте, поступово Internet Explorer відмовляється від використання NPAPI та ActiveX, а інтерфейс PPAPI підтримується лише web-браузерами Google Chrome та Opera, що свідчить у тому, що немає універсального рішення. Відмітимо, що найчастіше використовується для web-збору та перспективними сьогодні є технологія JavaScript.

JavaScript-програми. Вбудовувані на web-сторінки JavaScript-програми виконуються на стороні клієнта і тим самим можуть взаємодіяти із зовнішніми ресурсами, збираючи інформацію про їхнє використання без використання додаткового ПЗ.

Однак, варто мати на увазі, що у різних web-браузерах можуть використовуватись різні версії JavaScript-інтерпретатора, що необхідно враховувати під час розробки. Також, автори говорять про великі затримки при обробці подій натискання на клавіші клавіатури при використанні технологій web-збору, тривалість яких істотно залежить від ступеня завантаженості комп'ютера. Час, що минає від моменту натискання на клавішу до виклику обробника цієї події, може становити від десятків до сотень мілісекунд, що значно нижче швидкості обробки подій локальних збирачів даних засобами операційної системи. Також варто зазначити, що при використанні JavaScript-технологій в більшості web-браузерів немає можливості визначити, ліві чи праві функціональні клавіші (Shift, Ctrl, Alt і т.д.) були натиснуті, що є серйозним недоліком, оскільки дана інформація може суттєво допомогти автентифікувати користувача.

Розширення для веб-браузера. Розширення для web-браузера є програмами, що розширюють його функціональні можливості. На відміну від розглянутих вище JavaScript програм, що вбудовуються в коди web-сторінок, розширення для web-браузера надають можливість збору даних, що характеризують динаміку роботи користувача з клавіатурою, під час перегляду будь-яких web-сторінок, а не тільки тих, у коди яких заздалегідь вшиті JavaScript дані. Це обґрунтовується здатністю розширень модифікувати код веб-сторінок, що переглядаються. В

цьому полягає їхня головна відмінність від плагінів. Однак, варто мати на увазі, що різні браузері надають різні програмні інтерфейси для написання розширень (у тому числі і вимагають реалізації різними мовами програмування), що значно ускладнює створення універсальних рішень.

Flash – це мультимедійна платформа компанії Adobe Systems, призначена для створення інтерактивних web-додатків з багатою векторною, растровою, тривимірною комп'ютерною графікою та мультимедіа, що працюють як усередині, так і поза веб-браузером. Для створення програм використовується власне розроблена мова ActionScript. Adobe Flash використовується у веб-браузерах Opera та Google Chrome (за допомогою інтерфейсу PPAPI), а також у веб-браузері Firefox (за допомогою інтерфейсу NPAPI). Варто відзначити, що використання Flash сильно уповільнює роботу браузера, внаслідок чого попит на цей продукт поступово знижується.

Зауважимо, що при використанні технології Flash зібрати дані динаміки роботи користувача з клавіатурою вийде тільки всередині Flash-об'єкта.

### **Література**

1. N. Poolsappasit, R. Dewri, I. Ray Dynamic security risk management using Bayesian attack graphs. *IEEE Transactions on Dependable and Security Computing*. 2012. Vol. 9. No.1. P. 61–74.
2. T. Toth, C. Kruegel Evaluating the impact of automated intrusion response mechanisms. *Proceedings of the 18th Annual Computer Security Applications Conference (ACSAC)*. 2020. P. 301–310.
3. Gibellini, E.; Righetti, C. Unsupervised Learning for Detection of Leakage from the HFC Network. In *Proceedings of the ITU Kaleidoscope: Machine Learning for a 5G Future (ITU K)*, Santa Fe, Argentina, 26–28 November 2018; pp. 1–8. [CrossRef].

## АНАЛІЗ ЗАСОБІВ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ НА ОСНОВІ КЛАВАТУРНОГО ПОЧЕРКУ

## ANALYSIS OF KEYBOARD-BASED USER AUTHENTICATION MEANS

На сьогоднішній день кількість рішень щодо аутентифікації користувачів на основі динаміки їхньої роботи з клавіатурою персонального комп'ютера постійно збільшується. І якщо раніше дані системи обмежувалися лише аналізом введення пари логін/пароль (розглядалася виключно статична аутентифікація), то зараз активно розвиваються системи, здатні аналізувати поведінку користувача за комп'ютером безперервно.

BehavioWeb (рисунок 1). Одним з найбільш відомих комерційних рішень у галузі безперервної фонові аутентифікації користувачів за клавіатурним почерком є продукт BehavioWeb компанії BehavioSec. Для аналізу поведінки користувача у ньому використовуються ритм та швидкість набору тексту, а також сила натискання на клавіші.

Програмне забезпечення вбудовується у веб-сайт або додаток. Для цього використовуються JavaScript-бібліотека, що поставляється, а також J2EE-модуль, що вбудовується в веб-сервер для здійснення процедури аутентифікації. Розробники звертають увагу на те, що їхнє рішення аналізує зміну характеристик введення користувача з часом і періодично оновлює модель користувача. Проте, алгоритми, що використовуються для цього, не називаються.

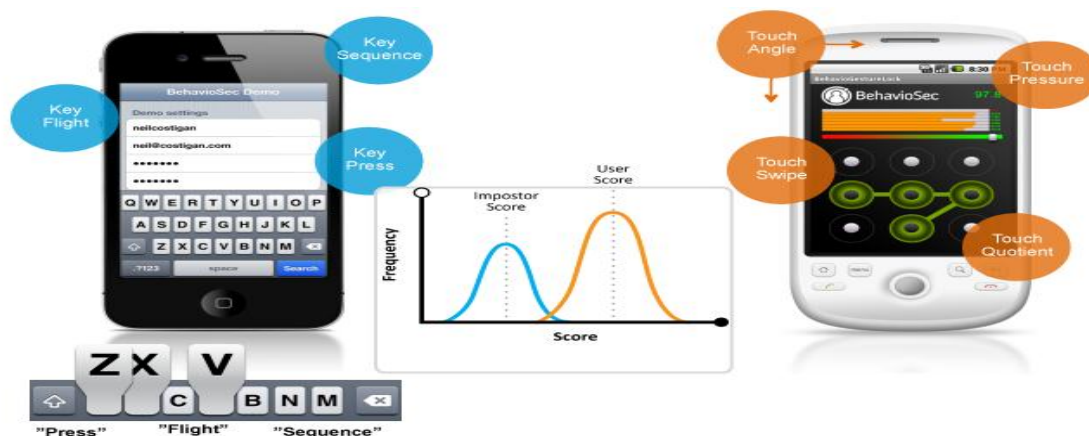
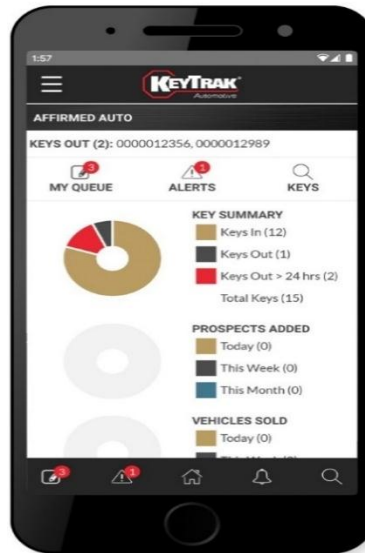


Рисунок 1. Система BehavioWeb

KeyTrac (рисунок 2). Не менш популярним рішенням є продукт KeyTrac [4, 5], що дозволяє здійснювати у фоновому режимі як аутентифікацію, так ідентифікацію користувачів комп'ютера, ґрунтуючись на динаміці їх клавіатурного введення. Дані користувачів (тривалість натискання на клавіші клавіатури, а також тривалості перескоку між клавішами) записуються за допомогою компонента KeyTrac Recorder і відправляються на сервер компанії, де відбувається їх порівняння із побудованою раніше моделлю. При цьому побудова моделі користувача здатна здійснюватися на будь-якому довільному тексті, а не тільки при багаторазовому введенні тих самих фраз. Для передачі даних використовується наданий KeyTrac API. Далі сервер повертає свій вердикт у вигляді булевої величини true/false – чи

відповідають надіслані тестові дані розглянутому легітимного профілю чи ні. Для вбудовування цього рішення на веб-сайт також пропонується використовувати JavaScript-бібліотеку, що надається. Розробники системи стверджують, що їх рішення нечутливе до зміни мови, що використовується, а також до зміни обладнання, що використовується. Використовувані для цього алгоритми, а також методи побудови моделі та їх подальшої класифікації не називаються.



**Рисунок 2.** Система KeyTrac

### **Література**

1. Peltier, T. R. Information security risk analysis, Third Edition. CRC Press, 2020. 456 p.
2. Olsson, T. Assessing security risk to a network using a statistical model of attacker community competence. Proceedings of the 11th international conference on Information and Communications Security. 2019. P. 308–324.
3. Peltier, T. R. Information security risk analysis, Third Edition. CRC Press, 2020. 456 p

УДК 004:519.2

**Т. Базан**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АНАЛІЗ ВХІДНИХ ДАНИХ СИСТЕМИ ПРОГНОЗУВАННЯ ФІНАНСОВОЇ РЕНТАБЕЛЬНОСТІ ПІДПРИЄМСТВА**

UDC 004:519.2

**T. Bazan**

## **ANALYSIS OF INPUT DATA OF THE SYSTEM FOR FORECASTING THE FINANCIAL PROFITABILITY OF THE ENTERPRISE**

Для виконання дослідження використовуються дані із сайту Kaggle [1] – відкритого джерела різноманітних конкурсних завдань у сфері аналізу даних. Дана платформа дозволяє ділитися своїми рішеннями, обговорювати процес аналізу та змагатися у точності побудованих моделей. Використані дані «Financial Distress Prediction: Bankruptcy Prediction». Контекст даних – це набір даних (НД), призначений для прогнозування фінансової кризи для вибірки компаній. Фрагмент стовпців даних представлений на рисунку.

	Company	Time	Financial Distress	x1	x2	x3	x4
0	1	1	0.010636	1.2810	0.022934	0.87454	1.21640
1	1	2	-0.455970	1.2700	0.006454	0.82067	1.00490
2	1	3	-0.325390	1.0529	-0.059379	0.92242	0.72926
3	1	4	-0.566570	1.1131	-0.015229	0.85888	0.80974
4	2	1	1.357300	1.0623	0.107020	0.81460	0.83593
...	...	...	...	...	...	...	...

Перший стовпець: Є ідентифікатором компанії. Другий стовпець: Показує різні періоди часу, до яких належать дані. Довжина часового ряду варіюється від 1 до 14 кожної компанії. Третій стовпець: Цільова змінна позначається як «Фінансова криза», якщо вона більша за -0,5, то компанію слід вважати рентабельною (0). В іншому випадку він був би розцінений як фінансово неблагополучний (1). Від четвертого до останнього стовпця: характеристики, позначені від x1 до x83, є деякі фінансові та нефінансові характеристики відібраних компаній. Ці характеристики відносяться до попереднього періоду часу, який слід використовувати для прогнозування того, буде компанія відчувати фінансові труднощі чи ні. Класифікація. Особливість x80 є категоріальною змінною. Наприклад, компанія 1 зазнає фінансових труднощів у момент 4, але компанія 2 все ще знаходиться в хорошому стані в момент 14.

Цей НД не є збалансованим (є 136 компаній із фінансовими проблемами проти 286 здорових, тобто 136 спостережень за звітний рік є фінансовими проблемами, тоді як 3546 спостережень за звітний рік є рентабельними). Слід зазначити, що 30% цього набору даних слід випадковим чином призначити як набір тестових даних, так що 70% тих, що залишилися, використовуються для вибору функцій і вибору моделі.

До даних також додаються примітки. Стовпець перший – ці дані можна розглядати як проблему класифікації. Стовпець другий – дані також можна розглядати як проблему регресії, а потім результат буде перетворено на класифікацію. Стовпець третій – ці дані можна розглядати як багатовимірну класифікацію часових рядів.

Основними проблемами даних для дослідження є такі пункти: які особливості найбільше свідчать про фінансову скруту; які типи моделей машинного навчання найкраще працюють із цим набором даних?

### **Література**

1. Kaggle: Your Machine Learning and Data Science Community. URL: [^https://www.kaggle.com/](https://www.kaggle.com/) (дата звертання: 01.12.22).

УДК 004.9

**К. Бєлоусов, Т. Маєвський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)  
(Технічний коледж Тернопільського національного технічного університету імені Івана Пулюя)

## **РОЛЬ ТА ЗНАЧЕННЯ ВЕЛИКИХ ДАНИХ В СУЧАНИХ НАУКОВИХ ДОСЛІДЖЕННЯХ**

UDC 004.9

**К. Bielousov, T. Maievskyi**

## **THE BIG DATA ROLE AND SIGNIFICANCE IN MODERN SCIENTIFIC RESEARCH**

Інноваційний інформаційно-технологічний концепт «Великі дані» трансформує сучасні наукові дослідження. Вони об'єднують великі за обсягом масиви та колекції даних зібрані під час лабораторних експериментів, наукових досліджень, опрацювання різнотипів персональних та медичних записів, в тому числі накопичені з використанням Інтернету речей (IoT-пристроїв). Зокрема, в медичній галузі, експоненційно зростає швидкість отримання інформації геноміки та епігеноміки, транскриптоміки та протеоміки, метаболоміки та фармакогеноміки [1]. Це дає перспективні можливості проведення досліджень та, як наслідок, вагомих досягнень для персоналізованої медицини, прокладаючи при цьому шлях до покращення якості життя та надання медичних послуг.

«Великі дані» (англ. Big Data) характеризуються величезними обсягами забраних, накопичених та опрацьованих даних, створених у приватному та державному секторах людської діяльності. Вони спрямовані на заохочення використання інноваційних інформаційних технологій для аналізу великої кількості доступних наборів даних, видобування та отримання нових знань та інформації. «Великі дані» можуть набувати будь-яку форму складних даних великого обсягу.

Колекції «великих даних», при формуванні обчислювальних систем з метою їх аналітичного опрацювання, зазвичай характеризуються п'ятьма характеристиками: обсяг розмірів загального набору даних, швидкість з якою обчислювальні конвеєри можуть завантажувати та опрацьовувати набори даних, достовірність наборів даних у відображенні справжньої природи стану систем, різноманітність елементів наборів даних один відносно одного в колекції, мінливість – якісна характеристика даних, що представляє постійні зміни в даних та спричиняє відповідний вплив на продуктивність обчислювальної системи [2]. В комплексі всі зазначені якості сприяють системній цінності «великих даних» при застосуванні в будь-якій галузі наукових досліджень. Їх необхідно ретельно враховувати при формуванні процесів роботи з «великими даними».

Тому доцільно критично розглянути та проаналізувати основні використані аналітичні обчислювальні методи, алгоритми та отримані з їх допомогою результати, які сприяли нещодавнім науковим досягненням, отриманим з використанням великих даних. Водночас варто проаналізувати тенденції та перспективні напрямки проведення наукових досліджень в галузі великих за обсягом даних.

### **Література**

1. Khan, Ibrahim Haleem, and Mohd Javaid. «Big data applications in medical field: A literature review. «Journal of Industrial Integration and Management» 6.01 (2021): 53–69.
2. Cremin, Conor John, Sabyasachi Dash, and Xiaofeng Huang. «Big data: Historic advances and emerging trends in biomedical research.» Current Research in Biotechnology (2022).

УДК 004.9

**К. Бєлоусов, Т. Маєвський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)  
(Технічний коледж Тернопільського національного технічного університету імені Івана Пулюя, Україна)

## **ВЕЛИКІ ДАНІ ТА АНАЛІТИЧНЕ ОПРАЦЮВАННЯ В НАУКОВИХ ДОСЛІДЖЕННЯХ**

UDC 004.9

**К. Bielousov, T. Maievskyi**

## **BIG DATA AND ANALYTICAL PROCESSING IN SCIENTIFIC RESEARCH**

На даний час спостерігається безпрецедентне зростання обсягів колекцій даних накопичених в результаті наукових досліджень. Установи та організації використовують різнотипові аналітичні інструменти, сформовані на основі методів штучного інтелекту (англ. Artificial intelligence, AI) та методів машинного навчання (англ. Machine Learning, ML) для отримання знань та інформації на основі даних, щоб зменшити витрати, збільшити потоки доходів, розробити персоналізовані сервіси та послуги. Невпинно зростає перелік, які використовують «великі дані» (англ. Big Data, BD) для вдосконалення та оптимізації своїх технологічних та бізнес-процесів, виявлення ринкових споживчих тенденцій.

За оцінками фахівців прогнозується, що капіталізація світового ринку «великих даних» до 2025 року перевищить 70 мільярдів доларів США. Водночас очікується, що Сполучені Штати домінуватимуть в цій галузі, сприяючи відбору та аналітичному опрацюванню понад дев'яносто відсотків колективних «великих даних» Північної Америки до 2025 року. Використання «великих даних» в наукових дослідженнях швидко зростає. На даний час обсяги дослідницьких даних, що генеруються за один день, за оцінками фахівців, співмірна з обсягами, які раніше генерувалися за десятиліття. Нещодавні звіти приватних компаній в галузі аналітичного опрацювання даних, в тому числі Всесвітнього економічного форуму, свідчать, що наразі людство оперує сорока чотирма зетабайтами даних. В перспективі ці обсяги зростатимуть до 463 екзабайтів накопичуваних даних щодня в усьому світі [1]. Великі технологічні гіганти, зокрема Google, Facebook, Amazon і Microsoft, зберігають приблизно 1200 петабайт даних завдяки доступу до цифрових технологій по всьому світу. У сукупності статистичні дані свідчать, що розвиток цифровізації, доступність Інтернету та поширення Інтернету речей (IoT) сприяють зростаючому перевантаженню даними.

Складна природа наборів сучасних дослідницьких наборів та колекцій даних формує складну задачу видобування та отримання значущих висновків та знань з «великих даних». Накопичення наборів даних з більш різноманітними колекціями «великих даних» дозволяє ефективніше ідентифікувати загальні закономірності, притаманні загальній масі наборів даних, за рахунок незначних характеристик, які можуть висвітлити важливі області [2]. Тому аналітичне опрацювання «великих даних» повинно використовувати надійні методи для ідентифікації надійних ознак.

### **Література**

1. How Much Data Is Created Every Day? [27 Staggering Stats], 2022. URL: [https://earthweb.com/how-much-data-is-created-every-day/?gclid=Cj0KCQIAkMGcBhCSARIsAIW6d0B0cDctoIEGRbWuT3s7MdUY-ppN87LCz9C9Z\\_RKrkGKvUejqzSVDWEaArPS EALw\\_wcB](https://earthweb.com/how-much-data-is-created-every-day/?gclid=Cj0KCQIAkMGcBhCSARIsAIW6d0B0cDctoIEGRbWuT3s7MdUY-ppN87LCz9C9Z_RKrkGKvUejqzSVDWEaArPS EALw_wcB).
2. Boehm, Kevin M., et al. «Harnessing multimodal data integration to advance precision oncology.» *Nature Reviews Cancer* 22.2 (2022): 114–126.



УДК 004.6

**А. Блавицький, С. Мацюк, С. Криськова**

(Тернопільський національний технічний університет імені Івана Пулюя)

## **ОЦІНКА РОЗВИТКУ БЕЗПЕКИ ОПЛАТИ ПЛАТІЖНИМИ КАРТКАМИ**

UDC 004.6

**A. Blavitskyi, S. Matsiuk, S. Kryskova**

## **ASSESSMENT OF THE SECURITY DEVELOPMENT OF PAYMENT CARDS**

Навіть сьогодні в операціях з платіжними картками все ще використовуються застарілі технології. Щоб вирішити цю ситуацію, важливо проаналізувати та оцінити розвиток безпеки в методах електронних платежів, щоб зрозуміти, як захист даних, що використовуються в платіжних транзакціях, покращився з появою нових технологій.

Аналіз структурних елементів платіжної картки виявив десять елементів. Кожен із цих елементів, крім двох, реалізує функцію захисту платіжної картки. Першим винятком є магнітна смуга, яка є застарілою та незахищеною технологією для каналу наявної картки, яка все ще використовується, щоб зробити платіжну картку зворотною сумісністю із застарілими POI.

Другим винятком є CVV2, який є механізмом безпеки для каналу відсутності картки, який лише підтверджує володіння платіжною карткою.

Вивчення життєвого циклу платежу карткою виявило щонайменше п'ять суб'єктів, які відіграють певну роль у платіжному процесі. Для процесу оплати карткою доступні два платіжні канали: канал наявності картки та канал відсутності картки. Ці канали вказують на присутність або відсутність власника картки на об'єктах продавця, коли ініціюється оплата карткою.

Було розглянуто три технології EMVCo: чіп EMV, EMV 3DS 2.0 і токенизація платежів EMV. Кожна з цих технологій реалізує кілька функцій безпеки, включаючи автентифікацію даних, методи перевірки власника картки, потоки автентифікації, а також методи перевірки та ідентифікації. Ці функції безпеки є важливими, оскільки їх наявність, відсутність або поєднання впливає на загальну безпеку технології.

Дослідження стосувалося чотирьох стандартів PCI SSC: PCI DSS, PCI PTS, PCI PA-DSS і PCI P2PE. Головною метою цих стандартів є захист конфіденційних даних власника картки.

В майбутньому важливо продовжити детальне вивчення нових механізмів безпеки, які впроваджуються для інноваційних та альтернативних методів електронних платежів. Крім того, постійні дослідження та інновації в механізмах автентифікації покращать безпеку платежів карткою для обох платіжних каналів. Зазначимо, що постійна еволюція злочинних атак на електронні платежі з метою отримання незаконної вигоди повинна супроводжуватися та протистояти розробка нових технологій та заходів безпеки.

### **Література**

1. «Payment Card Industry 3-D Secure (PIC 3DS) – Security Requirements and Assessment Procedures for EMV 3-D Secure SDK,» PCI Security Standard Council. [Online]. URL: <https://bit.ly/2xPLm5R> (Accessed: Nov. 10, 2022).

УДК 004.4

**А. Буковська**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ПАРАЛЕЛЬНЕ ТА РОЗПОДІЛЕНЕ ГЕНЕРУВАННЯ POWERSSET З ВИКОРИСТАННЯМ ПЛАТФОРМИ ОБРОБКИ ВЕЛИКИХ ДАНИХ**

UDC 004.6

**A. Bukovska**

### **PARALLEL AND DISTRIBUTED POWERSSET GENERATION USING A BIG DATA PLATFORM**

Новий напрямок досліджень – Великі дані. Прикладами гілки великих даних є структуровані і неструктуровані дані, медіа або випадкові процеси, оскільки вони практично не можуть бути оброблені традиційним способом. На зміну традиційним монолітним системам приходять нові асинхронні та паралельні рішення. Ці нові рішення забезпечують можливість роботи з великими даними [1].

Інформаційна технологія Big Data – це сукупність методів і засобів обробки різних типів структурованих (бази даних) і неструктурованих (текст, потік) динамічних великих обсягів даних для їх аналізу та використання для підтримки прийняття рішень.

Кластеризація є одним із способів зменшити часову складність обробки великих даних. Слід враховувати два варіанти масштабування, тобто горизонтальне та вертикальне масштабування. Горизонтальне масштабування ділить набір даних і розподіляє дані між кількома серверами або кластерами.

Основними технологіями обробки великих даних є: NoSQL; MapReduce; Apache Hadoop; Apache Spark.

Проблему збільшення обсягу інформації неможливо вирішити за допомогою класичних реляційних архітектур. Однією з проблем класичної реляційної бази даних є проблема роботи з масивними даними та проектами з високим навантаженням. Перший цільовий підхід полягає в тому, щоб розширити базу даних, якщо SQL досить гнучкий, а не переміщувати її, де б вона не виконувала свої завдання. Також реляційний підхід не підтримує обидва типи масштабування (вертикальне та горизонтальне).

Існують класичні підходи та парадигми розвитку засобів обробки даних. Однією з них є парадигма MapReduce [2]. Ця модель розподіленої обробки даних запропонована Google для обробки значного обсягу даних на обчислювальних кластерах. MapReduce забезпечує організацію даних у вигляді списків, які проходять 3 етапи обробки: етап карти (map stage), етап перемішування (shuffle stage), зменшення сцени (reduce stage).

Алгоритм Powerset є одним із алгоритмів, які використовуються для вилучення ознак у інтелектуальному аналізі даних [3], а також для генерування та вибору унікальних функцій [4].

Формування наборів потужностей за допомогою обробки MapReduce:

1. Partitioning  $S$  into subsets  $S[m]$ .
2. for  $i \in \{0, \dots, m-1\}$ .
3. if  $(0 < S_m < m) S[i] = d/d$  is number of elements in subset.
4. END if.
5. END for.
6. Mapping  $S[m]$  to Mapper Machines.
7. Mapper (Input  $S[i]$ , output  $P(S_i)$ ).
8.  $E[S] =$  all possible combination of basic element for  $S[m]$ .
9. END mapper.
10. Reducers receive sub-powerset.
11. Reducer (Input  $E[S_j]$ , output  $P(S)$ ).

12. Combine results from all tasks to find powerset using Union operation.

13. END Reducer.

Алгоритм генерує набір потужностей шляхом поділу заданого набору на підмножини та відображення підмножин серед різних картографів у кластер. Машини картографів обчислюють набір потужностей для кожної підмножини. Потім редуктор об'єднує піднабори ступенів, використовуючи операцію об'єднання вмісту набору ( $u$ ), щоб створити остаточний набір ступенів набору  $S$ . Програма powerset (рис. 1) застосовується до кожної платформи.

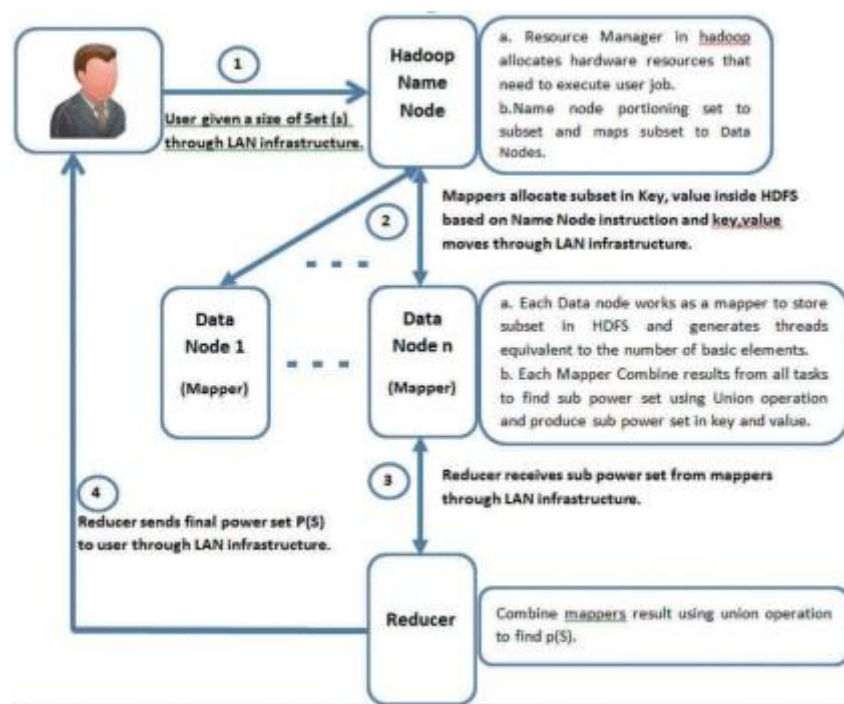


Рисунок 1. Створення Powerset за допомогою MapReduce

Це проста програма, яка створює набір даних розміру, заданого користувачем. Потім він розділяє набір на підмножини, які обробляються в паралельних і розподілених середовищах.

### Література

1. Janssen, M., van der Voort, H., & Wahyudi, A. (2017). «Factors influencing big data decision-making quality». *Journal of Business Research*, 70: 338–345.
2. De Mauro, A., Greco, M., & Grimaldi, M. (2016). «A formal definition of Big Data based on its essential features». *Library Review*, 65 (3): 122–135.
3. IEEE International Conference on Cluster Computing (CLUSTER), 433–42, Taipei, Taiwan, IEEE. Esfandiari, M., R. Babavalian, et al. 2014.
4. Spolaôr, E.A.Cherman, and et al. 2013. ReliefF for multi-label feature selection. *Brazilian Conference on Intelligent Systems*, 6–11.

## ВИКОРИСТАННЯ СТАКУ ELK ДЛЯ ДОСЛІДЖЕННЯ ПОДІЙ

## USING ELK STACK TO RESEARCH OF EVENTS

Logstash – це засіб для збору, фільтрації та структуризації log-файлів (подій). Це безплатний та open source додаток, створений на базі Apache Lucene. Додаток Logstash входить до стаку ELK - Elasticsearch, Logstash і Kibana, де Elasticsearch – пошукова і аналітична система, Logstash – серверний конвеєр для обробки даних, який може отримувати дані одночасно з декількох джерел, переробляти їх та відправляти на сервер, в нашому випадку – це Elasticsearch. Kibana – засіб для візуального подання інформації, який дозволяє створювати різні діаграми та графіки з інформації, яка знаходиться в Elasticsearch.

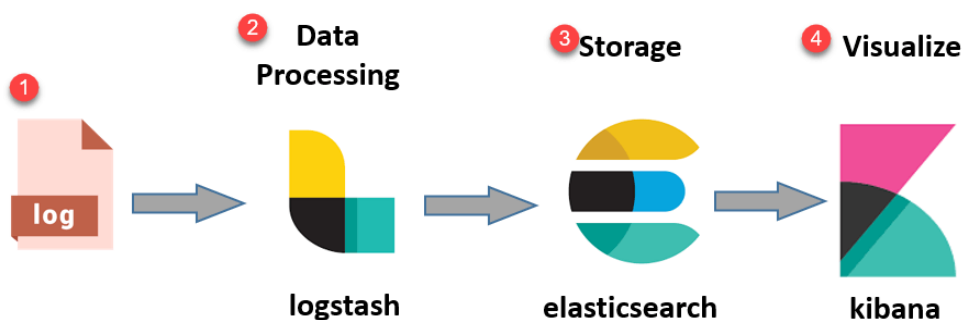


Рисунок 1. Схема роботи стаку ELK

Logstash використовує спеціальні вирази, які називаються грок патерни (grok-patterns) для розбору log-файлів. Grok – це фільтр всередині Logstash, який використовується для структуризації неструктурованих даних. Він знаходиться поверх регулярного виразу (regex), і використовує текстові шаблони для зіставлення рядків у файлах журналів. Logstash постачається з більш ніж 100 вбудованих шаблонів, які можна використати для загальних системних журналів apache, linux, harpoxu, aws тощо. Також є можливість написання свого власного патерну, за яким буде розбиратись лог. Для цього можна використати спеціалізовані сервіси для написання патерну, одним з найкращих є грок дебагер «herokuapp», в якому можна знайти приклади для різних частин логів. Синтаксис шаблону патерну виглядає наступним чином – % {SYNTAX:SEMANTIC}.

Після написання патерну, який буде проходитись по всіх рядках log-файлу, потрібно відправити структуровані файли на Elasticsearch. Після цього отримані дані можна буде візуалізувати за допомогою багатьох вбудованих засобів та візуалізацій, до яких входять різноманітні графіки і таблиці.

### Література

1. What is the ELK Stack? URL: <https://www.elastic.co/what-is/elk-stack>.
2. The complete guide to the elk stack. URL: <https://logz.io/learn/complete-guide-elk-stack/>.
3. ELK Stack Tutorial: What is Kibana, Logstash & Elasticsearch? URL: <https://www.guru99.com/elk-stack-tutorial.html>.

УДК 004.062

**В. Василенко, Н. Стадник**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ЛОГУВАННЯ – ЩО ЦЕ І В ЧОМУ ЙОГО КОРИСТЬ**

UDC 004.062

**V. Vasylenko, N. Stadnyk**

## **LOGGING – WHAT IS IT AND WHAT IS ITS BENEFIT**

Якщо в роботі сервера, комп'ютера або програмного забезпечення виникла невідома помилка, насамперед дивляться логи. Лог – це текстовий файл з інформацією про дії програмного забезпечення або користувачів, який зберігається на комп'ютері або сервері. Це хронологія подій та їх джерел, помилок та причин, з яких вони сталися. Читати та аналізувати логи можна за допомогою спеціального ПЗ.

Логуванням називають запис логів. Воно дозволяє відповісти на питання, що відбувалося, коли і за яких обставин. Без логів складно зрозуміти, через що з'являється помилка, якщо вона виникає періодично і лише за певних умов. Записується інформація не тільки про помилки, але і про причини їх виникнення.

Існують різні рівні та різні подробиці логування. Якщо помилка унікальна, використовують максимально докладні логи; якщо це не потрібно, збирають лише ключову інформацію. Для роботи злогами та пошуком інформації у великих текстових даних використовують спеціалізовані інструменти. Для зручної роботи злогами їх ділять на типи. Це допомагає швидше знаходити потрібні та вибирати правильні інструменти для роботи з ними. Наприклад, виділяють:

- системні логи, тобто ті, які пов'язані із системними подіями;
- серверні логи, що реєструють звернення до сервера і помилки, що виникли при цьому;
- поштові логи, що стосуються вхідних/вихідних листів та відслідковують помилки, через які листи не були доставлені;
- логи аутентифікації.

Лог файли можуть знадобитися у багатьох ситуаціях під час роботи зі сайтами, ПК або сервером. Але зверніть увагу, що логи не зберігаються вічно, тому якщо виникла потреба перевірити їх, слід це робити своєчасно. Наприклад, часто хостинг-провайдери зберігають логи до 14 днів, а далі вони видаляються та записуються нові.

Отже, якщо сайт зламали більше кількох тижнів тому, встановити причину по логах не можливо, при умові, що логи вже видалені.

### **Література**

1. Usage logging. URL: <https://www.ibm.com/docs/en/csfdcd/7.1?topic=ult-usage-logging/>
2. Logging. URL: [https://en.wikipedia.org/wiki/Logging\\_\(software\)](https://en.wikipedia.org/wiki/Logging_(software)).
3. An introduction to logging for programmers. URL: <https://www.freecodecamp.org/news/you-should-have-better-logging-now-fbab2f667fac/>.
4. The Log: What every software engineer should know about real-time data's unifying abstraction. URL: <https://engineering.linkedin.com/distributed-systems/log-what-every-software-engineer-should-know-about-real-time-datas-unifying>.

УДК 004

**Р. Волошин**

(Тернопільський національний технічний університет імені Івана Пулюя)

## **АУДИТ БЕЗПЕКИ AMAZON SELLING PARTNER API**

UDC 004

**R. Voloshyn**

### **AMAZON SELLING PARTNER API CYBERSECURITY AUDIT**

Selling Partner API (SP-API) – це API на основі технології REST, яка допомагає партнерам з продажу Amazon програмно отримувати доступ до своїх даних про замовлення, відправлення, платежі та багато іншого. Програми, що використовують SP-API, можуть підвищити ефективність продажів, зменшити потреби в робочій силі та покращити час відповіді клієнтам, допомагаючи партнерам з продажу розвивати свій бізнес. Щоб мати змогу створити програму з використанням SP-API розробники повинні пройти реєстрацію з вказанням усіх деталей щодо бізнесу та рівня компанії, на яку вони працюють та аудит безпеки.

Аудит безпеки для Amazon Selling Partner API включає аудит щодо:

- захисту мережі (наявність засобів керування захистом мережі, включаючи мережеві брандмауери та списки контролю доступу до мережі, щоб заборонити доступ до несанкціонованих IP-адрес; сегментація мережі, програмне забезпечення для захисту від вірусів і шкідливих програм на пристроях кінцевих користувачів; публічний доступ лише для схвалених користувачів),
- керування доступом до даних та ресурсів (офіційний процес реєстрації доступу користувачів, щоб призначити права доступу для всіх типів користувачів і послуг, забезпечивши присвоєння унікального ідентифікатора кожній особі, алгоритм “блокування” користувача за спричинені несанкціоновані дії),
- шифрування під час передавання інформації (необхідно шифрувати всю інформацію, що передається, за допомогою безпечних протоколів, таких як TLS 1.2+, SFTP і SSH-2),
- шифрування REST запитів (за допомогою принаймні AES-128 або RSA з розміром ключа 2048-біт або вище),
- зберігання інформації щодо покупців на серверах без шифрування цієї інформації,
- наявність плану реагування на ризики та менеджмент ризиків, систем логування доступів та змін.

Даний аудит містить не лише ці пункти, але і значно більший перелік. У випадку, якщо веб-додаток розміщується на серверах Amazon, то для автоматизації використовують Amazon SP-API Guard. У випадку розміщення програми на інших серверах необхідно використовувати цілий стек програм для аналізу вимог та їх забезпечення, що в кінцевому випадку свідчить про значні витрати.

#### **Література**

1. Data Protection Policy. URL: <https://sellercentral.amazon.com/mws/static/policy> (2022).
2. Acceptable Use Policy. URL: <https://sellercentral.amazon.com/mws/static/policy?documentType=AUP> (2022).

**ПОРІВНЯННЯ МЕТОДІВ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ****COMPARISON OF TIME SERIES FORECASTING METHODS**

Прогнозування часових рядів, має велике значення у плануванні. Особливу роль воно відіграє в бізнесі та фінансах: прогнозування щоденного курсу валют, цін акції компаній, місячних/квартальних продажів товару тощо. Також прогнозування часових рядів використовується у метеорології (наприклад температури повітря, кількості опадів), у енергетиці (наприклад прогнозування споживання електроенергії, води, газу).

Методи прогнозування значною мірою залежать від доступних даних. Так, якщо немає доступних даних, то використовуються якісні методи, що надають суто припущення на основі розроблених підходів. В іншому випадку, використовуються кількісні методи [1]. Надалі розглядатимуться саме кількісні методи прогнозування.

У кількісному прогнозуванні можуть використовуватися різноманітні методи, що можуть розроблятися в межах окремих сфер застосування. Кожен з цих методів має певні властивості, точність та вартість. Найпростіші методи використовують лише дані про змінну, яку потрібно прогнозувати. Для цього вони розділяють часові ряди на компоненти: тренд, сезонність і залишки або «шум». Кожен з цих компонентів відображає певний шаблон поведінки ряду. Методи прогнозування, які найчастіше використовуються, розробляються на основі регресивних моделей, моделей експоненційного згладжування, моделей ARIMA (Autoregressive Integrated Moving Average, авторегресивне інтегроване ковзне середнє).

Регресивні моделі прогнозування спираються опис зв'язку між декількома рядами. Застосування регресивної моделі дозволяє прогнозувати числові значення часового ряду на основі даних, які на нього впливають. Ця модель надає можливість виражати у вигляді певної функції залежність цільової змінної від інших змінних [2].

У моделях експоненційного згладжування для обчислення прогнозів використовуються зважені середні значення минулих даних у часовому ряді. Варто зазначити, що ваги експоненційно зменшуються для давніших даних. Тобто, для раніших даних значення ваг буде більшим [1, 3].

Моделі ARIMA, як і моделі експоненційного згладжування, є найбільш популярними моделями прогнозування, які застосовуються до нестационарних часових рядів. ARIMA спирається на моделі авторегресії (AR, Autoregression) та ковзного середнього (MA, Moving Average). Для усунення нестационарності ця модель використовує інтегровані ряди [3].

У даній роботі розглядаються основні методи прогнозування часових рядів, концепції, на яких вони базуються. Проводиться порівняння розглянутих методів залежно від особливостей їх використання та залежно від властивостей часових рядів, до яких вони застосовуються.

**Література**

1. Hyndman R.J., Athanasopoulos G. Forecasting: Principles and Practice. 3rd ed. Melbourne, Australia: OTexts, 2021. 442 p.
2. Palma W. Time series analysis. Wiley, 2016. 579 p.
3. Pathak P.P. Time Series Forecasting – A Complete Guide. Medium. URL: <https://medium.com/analytics-vidhya/time-series-forecasting-a-complete-guide-d963142da33f>.

УДК 004.056

**М. Гаврилов**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ПОВТОРНА ІДЕНТИФІКАЦІЯ ЛЮДЕЙ ЗА ФОТО ТА ВІДЕО ЗАСОБАМИ COMPUTER VISION**

UDC 004.056

**М. Havrylov**

### **RE-IDENTIFICATION OF PEOPLE FROM PHOTOS AND VIDEOS BY MEANS OF COMPUTER VISION**

Повторна ідентифікація особи (Re-ID) широко вивчалася як проблема пошуку конкретної особи через камери, що не перекриваються. Мета Re-ID полягає в тому, щоб визначити, чи з'являлася конкретна особа в іншому місці в інший час, зафіксоване іншою камерою. Особа може бути представлена зображенням, відеорядом і навіть текстовим описом. У зв'язку з гострою потребою громадської безпеки та збільшенням кількості камер спостереження в університетах, тематичних парках, на вулицях і тому подібне, повторна ідентифікація особи є обов'язковою у розробці інтелектуальних систем відеоспостереження. Re-ID напрямком виник не так давно, але враховуючи його дослідницький вплив та практичну важливість, він продовжує стрімко розвиватись.

Повторна ідентифікація особи є складним завданням через наявність різних точок зору, різну роздільну здатність зображення, зміну освітленості, невимушені пози, оклюзії, тощо. Ранні дослідницькі зусилля зосереджені в основному на ручній конструкції елементів зі структурами тіла або дистанційне метричне навчання. З розвитком глибокого навчання Re-ID досяг надихаючої продуктивності за широко використовуваними тестами. Проте все ще існує великий розрив між дослідницькими сценаріями та практичним застосуванням. Це спонукає багатьох осіб, які зацікавлені в даному напрямку шукати методи удосконалення Re-ID.[1]

Під час повторної ідентифікації особи порівнюють досліджувану особу з набором осіб із існуючої бази картинок для створення ранжованого списку відповідно до їх подібності, зазвичай припускаючи, що правильний збіг призначається одному з вищих рангів, в ідеалі рангу 1. Оскільки люди з галереї часто знімаються з пари камер, які не накладаються одна на одну, у різний час, візуальні варіації зовнішнього вигляду можуть бути значними. Повторна ідентифікація шляхом візуального зіставлення за своєю суттю є складною. Сучасні методи виконують це завдання здебільшого шляхом зіставлення просторових характеристик зовнішнього вигляду (наприклад, гістограми градієнта кольору та інтенсивності) за допомогою пари однознімкових зображень людини. Однак функції зовнішнього вигляду одного кадру внутрішньо обмежені через притаманну візуальну неоднозначність, спричинену подібністю одягу людей у громадських місцях, а зовнішній вигляд змінюється через зміну освітленості в поперечному ракурсі, різницю в точках огляду, захарашений фон і оклюзії. Бажано досліджувати просторово-часову інформацію з послідовностей зображень людей для повторної ідентифікації в громадських місцях [2].

В подальших дослідженнях планується покращити наявне рішення Re-ID під конкретно встановлені задачі цієї роботи, а саме для повторного виявлення конкретних людей на інших відео або фото матеріалах.

#### **Література**

1. Ye, Mang, et al. «Deep learning for person re-identification: A survey and outlook.» IEEE transactions on pattern analysis and machine intelligence 44.6 (2021): 2872–2893.
2. Wang et al. (2014) Wang, T., Gong, S., Zhu, X., and Wang, S. (2014). Person re-identification by video ranking. In ECCV.



## РОЛЬ CRM-СИСТЕМИ У СУЧАСНИХ БІЗНЕС-ПРОЦЕСАХ

О. Holyns'ka, I. Mudryk

### ROLE OF CRM SYSTEM IN MODERN BUSINESS PROCESSES

У сучасну інформаційну епоху, де все взаємопов'язано, усі традиційні речі зазнають інновацій, викликаних технологічним прогресом. Завдяки передовій мережевій технології режим роботи комерційного ринку також дещо змінився за останні десять років, і підприємства також почали використовувати різні системи програмного забезпечення для підтримки власної діяльності та розвитку. Немає сумніву, що клієнти є джерелом корпоративних прибутків, і те, як ефективно керувати клієнтами, є ключем до успіху компанії. У сучасному середовищі жорсткої конкуренції пристосування до потреб клієнтів, надання клієнтам права вибору продуктів і надання їм можливості отримати те, що вони дійсно хочуть, є ключем до конкурентного успіху будь-якої компанії.

Система управління взаємовідносинами з клієнтами є важливим елементом основної програми підприємства. У простому розумінні CRM – це центр обробки даних клієнтів, який може обробляти різні деталі всіх клієнтів компанії в CRM, включаючи відстеження та керування кожним потенційним клієнтом, який взаємодіє із корпоративним брендом, веб-сайтом або продуктом, а також може аналізувати його сліди та переваги. Зворотній зв'язок ринку протягом багатьох років довів, що це дуже допомагає розширювати бізнес підприємств, тому сучасні підприємства інвестували багато ресурсів у розвиток клієнтів, щоб швидше та ефективніше розширювати свою територію.

Роль CRM полягає в тому, щоб допомогти підприємствам скрутити відповідні відділи в єдину мотузку. Коли підприємство вступає в стадію швидкого зростання, деякі клієнти можуть бути маргіналізованими. Як і інші корпоративні прикладні системи, CRM-система також дуже допомагає автоматизувати роботу підприємств, прискорити процес виконання різноманітних завдань.

Таким чином, важливість CRM-систем у сучасному бізнес-середовищі добре відома як розробникам, так і потенційним та реальним клієнтам. Така система збирає, обробляє та керує всіма даними клієнтів, які є основою кожного бізнесу. CRM аналізує активних і потенційних клієнтів, і результати аналізу дозволяють підприємствам налагоджувати міцні ділові відносини з клієнтами, тим самим збільшуючи швидкість транзакцій. Що ще важливіше, CRM може допомогти компаніям знайти нових клієнтів і розвивати новий бізнес.

#### Література

1. Glova B., Mudryk I. Application of Deep Learning in Neuromarketing Studies of the Effects of Unconscious Reactions on Consumer Behavior. 2020 IEEE Third International Conference on Data Stream Mining & Processing (DSMP): Conference, Lviv, 21–25 August 2020. P. 337–340.
2. Kumar, V., & Reinartz, W. (2018). Customer Relationship Management. Springer Texts in Business and Economics. Doi:10.1007/978-3-662-55381-7

УДК 004.62

**В. Грицюк, М. Стадник**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **КЛАСТЕРИЗАЦІЯ СПАМ-ДОМЕНІВ МЕТОДАМИ МАШИННОГО НАВЧАННЯ**

UDC 004.62

**V. Hrytsiuk, M. Stadnyk**

## **SPAM DOMAINS CLUSTERIZATION BY USING MACHINE LEARNING METHODS**

Covid-2019 спричинив значний поштовх для ще більшого розвитку електронної комерції. Кількість людей, що замовляла одяг, побутові товари чи товари першої необхідності через інтернет тільки зростала. Водночас зростає і попит на різноманітні мобільні додатки та веб-інтерфейси для успішного ведення бізнесу та представлення бренду в мережі. Після завершення пандемії тенденція замовляти товари із супермаркетів чи з локальних брендів зберіглася. Відповідно власник торгового маркетплейсу чи брендованого магазину на Shopify чи іншій платформі повинен задуматись над безпекою транзакції та над стабільністю веб-додатку. Звичайно існують системи захисту банківських транзакцій, багатофакторна аутентифікація, Google reCaptcha v2 чи v3, проте зловмисники вигадують нові способи зробити електронний ресурс недоступним. Задача виявлення зловмисника або ресурсу, що здійснює DDoS атаку, ідентифікувати його як спам-домен є актуальною [1].

Спам-доменом називають домен, який попав у спам-список. Ідентифікація спам домена є попереднім етапом, перед тим як спам домен попадає у чорний список. При надсиланні листа чи при доступі до ресурсу веб додатку сервер розпізнає запит від вказаної IP адреси, перевіряє наявність її у спам списку і лише тоді здатен виконати запит. Звичайно, якщо IP адрес не вказаний у спам списку, сервер буде багатократно здійснювати свою роботу таким чином відбувається навантаження на систему та врешті – відмова у доступі. Формування актуального спам-списку сприяє вчасному виявленню зловмисних дій.

Для актуалізації спам-списку, що включає як новостворені домени, так і раніше сформовані використовують кілька технік. Найпростішою методикою є виявлення спам-домена на основі його дій, тобто пост-фактум, при цьому зловмисник уже домігся своєї цілі. Іншою технікою є збір параметрів про IP адресу та з використанням отриманої інформації відбувається класифікація IP адреси з використанням методів машинного навчання чи штучного інтелекту. Для прикладу сервіс Alexa таким чином рангує домени, а з метою захисту користувачів stop-forumspam.org висвітлює «токсичні» домени [2].

Однією проблемою при виконанні класифікації чи кластеризації є поява значної кількості нових доменів щоденно. Зважаючи на таку тенденцію необхідно виконувати агрегацію новостворених доменних імен і в зазначеному періоді здійснювати кластеризацію. В роботі було використано метод k-найближчого сусіда, дерево рішень, алгоритми на основі графів. За результатами оцінок метод k-найближчих сусідів є найбільш оптимальним для поставленої задачі.

### **Література**

1. Chio, C.; Freeman, D. Machine Learning and Security, O'Reilly Media, 2018, 125–180.
2. Webb, S., Caverlee, J. «Characterizing Web Spam Using Content and HTTP Session Analysis». The 4th Conference on Email and AntiSpam. Aug. 2007. Mountain View, CA.

УДК 004.416.2

**Н. Зарічний, Є. Тиш**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АВТОМАТИЗАЦІЯ ТЕСТУВАННЯ МОБІЛЬНИХ ДОДАТКІВ ЗА ТЕХНОЛОГІЄЮ AGILE**

UDC 004.416.2

**N. Zarichnyi, Ye. Tysh**

## **AUTOMATION OF MOBILE APPLICATION TESTING USING AGILE TECHNOLOGY**

В даний час технологія Agile набирає все більшої популярності у сфері розробки програмного забезпечення, у тому числі - для мобільних пристроїв.

Для автоматизації додатків, що проектуються із застосуванням гнучких методологій, найчастіше використовується методика Scripting. Він підходить для автоматизації тестування API, але є неоптимальним рішенням для підготовки автотестів GUI. В результаті, тестування графічного інтерфейсу проводиться вручну. Протягом коротких ітерацій на Agile проекті можливо провести тільки базові перевірки GUI. Такий підхід допустимий для веб- та десктоп-додатків, які розраховані на роботу з обмеженою кількістю браузерів та платформ. На відміну від них, мобільні програми розробляються під різні мобільні платформи, версії операційних систем і конфігурації пристроїв. Через те, що тестування обмежується базовими перевітками, багато дефектів GUI потрапляють у фінальну версію програми та виявляються кінцевими користувачами. Для мобільних програм така ситуація може призвести до отримання негативних відгуків від користувачів і, як наслідок, до комерційного провалу.

Agile – підхід до тестування має на увазі наступні зміни в роботі QA-команди:

- тестування перестає бути ізольованою фазою у створенні ПЗ та активно застосовується на всіх стадіях життєвого циклу продукту – починаючи з планування. Таким чином, QA -фахівці можуть своєчасно виявляти найпростіші у виправленні помилки (неточності, неоднозначні деталі та інші проблеми документації), скласти уявлення про програмний продукт задовго до написання коду та виявити його потенційно слабкі місця;

- обсяг тестової документації скорочується до мінімуму: на зміну докладним тест-кейсам приходять більш високорівневі та універсальні тест-плани та чек-листи. Формат чек-листа дозволяє не розписувати перевірки досконально, за рахунок цього документацію простіше підтримувати в актуальному стані, а у роботі тестувальника збільшується частка дослідницького тестування;

- на всіх стадіях розробки підтримується зворотний зв'язок між фахівцями з тестування та рештою членів команди (розробники, бізнес-аналітики, дизайнери, проектний менеджер). Завдяки цьому у кожного з учасників проекту формується повніше і всебічне бачення продукту;

- швидка віддача від тестування - знайдені баги підлягають оперативному виправленню, що дозволяє підтримувати «чистоту коду» та уникнути накопичення застарілого та складно підтримуваного коду;

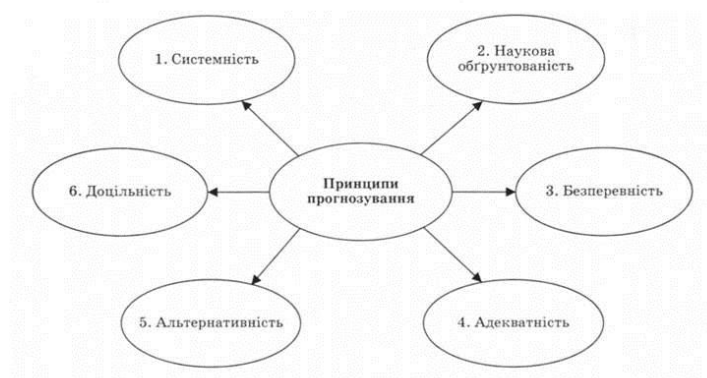
- тестування – невід'ємна частина критерію готовності: ступінь готовності ПЗ визначається з урахуванням кількості, пріоритету та серйозності виявлених проблем. Наприклад, критерієм готовності МП до випуску може бути відсутність у продукті дефектів з пріоритетом вище «незначного» (Minor) або «середнього» (Normal).

Таким чином, актуальність дослідження обумовлена необхідністю розробки методики, яка дозволить автоматизувати тестування API та графічного інтерфейсу мобільного додатка.

## ВИЗНАЧЕННЯ ПОГОДНИХ УМОВ У TELEGRAM

## DETERMINATION OF WEATHER CONDITIONS IN TELEGRAM

Прогнозування – це процес цілком якого є можливість передбачення певного стану предмета чи явища у майбутньому на основі здійснення аналізу сучасного і минулого. На рисунку 1 зображено основні принципи прогнозування [1].



**Рисунок 1.** Принципи прогнозування

Прогноз – аргументоване передбачення, яке надає інформацію про розвиток певних явищ і процесів у майбутньому. Типи прогнозів: пошуковий, нормативний, цільовий, плановий, програмний і проектний [2].

Погода – фактор який є елементом нашого повсякденного життя, який може кардинально змінювати наші плани і розвиток подій у майбутньому. Для прикладу, погана погода може вплинути на затримку дорожнього транспорту чи авіарейсів. Тому точність прогнозування і її результатів має вагомий вплив для людства. Проте, не можливо зробити ідеально точний прогноз [3]. У прогнозах виникають помилки і як наслідок – прогнози можуть бути не повними та не точними. Через постійних технічний та науковий розвиток, як і розуміння атмосфери та технічної галузі, прогнози постійно стають кращими і точнішими. Зараз точність прогнозу на наступні 4 дні дорівнює прогнозу на день, який здійснювали 30 років тому, а розвиток технічного забезпечення, надає нам можливість поглянути клімат який буде через кілька років у майбутньому [3].

На даний час фактично у кожного мешканця України є смартфони з різними додатками та месенджерами, наприклад Telegram чи Viber. За допомогою них люди комунікують між собою і можуть отримувати різну інформацію. Щоб не сталось жодних погодних сюрпризів перед тим як іти кудись люди переглядають прогноз погоди. Але перегляд інформації займає певний час, якого може не бути.

### Література

1. Методи прогнозування. URL: [https://pidru4niki.com/12920522/menedzhment/metodi\\_prognozuvannya/](https://pidru4niki.com/12920522/menedzhment/metodi_prognozuvannya/)
1. Прогноз. URL: <https://buklib.net/books/27070/>.
2. Як роблять прогнози погоди і чому вони не збуваються. URL: <https://www.bbc.com/ukrainian/features-51545290>.

УДК 004.042

**О. Кравчук**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **РОЗРОБКА ТЕЛЕГРАМ БОТІВ НА PYTHON**

UDC 004.042

**O. Kravchuk**

## **DEVELOPMENT OF TELEGRAM BOTS IN PYTHON**

За останні роки боти у телеграм набирають велику популярність так як автоматизувати більшість рутинних процесів і полегшити життя користувачів. Вони можуть здійснювати розсилку новин, допомагати здійснювати покупки у магазинах, керувати певними банківськими послугами, викликати таксі чи здійснювати розсилки з прогнозом погоди [1].

На сьогоднішній день у можна виділити лиш кілька способів для розробки ботів у телеграмі. В першу чергу слід виділити найпопулярнішу з можливих – BotAPI, яка забезпечує свою роботу через HTTP запити, що робить можливим отримання доступу до взаємодії із серверами telegram [2]. Під час вибору даного методу, користувач змушений власноруч створювати усі моделі взаємодії і не може користуватися уже готовими рішеннями для полегшення роботи і пришвидшення її виконання. При розборі такого способу створення боту стосовно якості, то результат буде малоефективним так як він буде без параметрів зручності і супроводу [2].

Найефективнішим і найкращим методом для розробки бота є використання додаткових бібліотек які мають можливість працювати із BotAPI. Python-telegram-bot – одна із найкращих бібліотек що надає змогу працювати з ботами у телеграмі[2]. Переваги бібліотеки:

- загальнодоступність;
- відкритий код;
- підтримка модифікацій;
- проста у користуванні.

Дана бібліотека також є простою і надійною у користуванні. Завдяки можливості роботи із об'єктами дана бібліотека є зручною у користуванні. Через об'єктно орієнтовану архітектуру і проявляється зручність використання даної бібліотеки, такий проект буде зручно модифікувати і доробляти у майбутньому для покращення його роботи, додання нових функцій і збільшення продуктивності. Для забезпечення надійності у боті використовується авторизація за допомогою токенів.[3].

Через свою простоту і доступність, фактично кожен користувач Telegram, має можливість спробувати реалізувати свою власну ідею і просувати її, кожного разу вносячи нові зміни і покращенні, а можливість використання додаткових API дозволяє підключати різні додаткові сервіси, покращувати дизайн/

### **Література**

1. Telegram Bot API. URL: <https://core.telegram.org/bots/api>.
2. Боти: вступ для розробників. URL: <https://core.telegram.org/bots>.
1. Документація Telegram Bot. URL: <https://python-telegram-bot.readthedocs.io/en/stable/index.html>.

УДК 004.9

**Н. Лісовий, А. Ставицька, А. Гіжовський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)  
(Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)

(Технічний коледж Тернопільського національного технічного університету імені Івана Пулюя, Україна)

## **АНАЛІТИЧНЕ ОПРАЦЮВАННЯ ВЕЛИКИХ ЗА ОБСЯГОМ ДАНИХ**

UDC 004.9

**N. Lisovyi, A. Stavytska, A. Hizhovskiy**

## **LARGE DATA VOLUMES ANALYTICAL PROCESSING**

В продовж останнього періоду часу експоненційно зростають обсяги цифрових даних, що створені засобами різнотипових IoT-пристроїв та інформаційно-технологічних платформ. Це зростання відбувається завдяки недавньому розвитку інформаційних та комунікаційних технологій та їх активному впровадженню у практично всі сфери людської діяльності. Зокрема, збільшенням кількості інтелектуальних пристроїв, що генерують дані з інтегрованими дачачами та виконавчими механізмами, які підключені засобами глобальних загальнодоступних хмарних сервісів, збільшенням кількості користувачів Інтернету, активним запровадженням інформаційних технологій з елементами віртуальної та доповненої реальності, розвитком мобільного зв'язку 5G, популяризацією соціальних мереж, збільшення транзакцій електронної комерції тощо. Станом на грудень 2020 року обсяг щоденних цифрових даних у всьому світі становив 59 зетабайт. За прогнозами фахівців очікується, що в 2024 році він досягне 149 зетабайт [1], оскільки людство переходить в майбутнє, яке буде більше керуватися даними.

Інтелектуальні технології аналітичного опрацювання «Великих даних» активно використовуються при розбудові «розумних міст» з інтегрованими інформаційними системами керування дорожнім рухом, які автоматично оптимізують транспортні потоки, системами моніторингу характеристик навколишнього середовища, які оновлюють дані щодо його забруднення в режимі реального часу та прогнозують зміну якості повітря та води, раціоналізованим та автоматизованим збиранням сміття та побутових відходів, інтелектуальними системами паркування транспортних засобів у густонаселених містах [2]. Вибухове зростання обсягів накопичуваних даних сформувало обширний перелік задач, пов'язаних з оперативним збиранням даних, ефективним їх зберіганням, пошуком, обробкою та поданням через зростання характеристик обсягу, різноманітності та швидкості даних. Процеси видобування знань або виявлення корисних шаблонів у великих за обсягом наборах та колекціях даних на даний час потребують величезних обчислювальних ресурсів, розвиненої та розгалуженої інформаційно-технологічної інфраструктури та кваліфікованих фахівців в галузі аналітики даних та є актуальним напрямком сучасних наукових досліджень.

### **Література**

1. Holst, A. «Amount of information globally 2010–2024.» Statista. URL: <https://www.statista.com/statistics/871513/worldwide-datacreated/#:~:text=The%20total%20amount%20of%20data,ever%2Dgrowing%20global%20data%20sphere>. Erişim Tarihi 27 (2020): 2020.
2. Khan, Shahbaz. «Barriers of big data analytics for smart cities development: a context of emerging economies.» International Journal of Management Science and Engineering Management 17.2 (2022): 123–131.

УДК 004.67

**Н. Шаблій, П. Марценюк**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **СИСТЕМИ МОНІТОРИНГУ СТАНУ ДОВКІЛЛЯ**

УДК 004.67

**N. Shabliy, P. Martseniuk**

## **ENVIRONMENTAL STATE MONITORING SYSTEMS**

Стан довкілля значною мірою залежить від компетентності фахівців у галузі екологічної безпеки, які відповідають за прийняття управлінських рішень щодо зменшення негативного впливу на природне навколишнє середовище. Спеціалізовані системи управління екологічною безпекою за даними моніторингу необхідні для: виявлення і прогнозування прихованих тенденцій і закономірностей розвитку екологічних процесів, ідентифікації раніше невідомих взаємозв'язків між екологічними параметрами і факторами впливу, розробки оптимізаційних рекомендацій у галузі екологічної безпеки, візуалізації результатів аналізу, підготовки попередніх звітів і проектів допустимих рішень з відповідним оцінюванням тощо. Об'єктом спостереження моніторингу можуть бути як окремі точки й зони, так і глобальні природні системи; різні види та рівні моніторингу відрізняються залежно від предмета спостереження, ступеня антропогенного порушення об'єктів контролю, просторово-часових параметрів досліджень, методів досліджень, джерел і факторів антропогенного впливу тощо. Система екологічного моніторингу повинна накопичувати, систематизувати та аналізувати інформацію про стан довкілля, про причини спостережуваних і ймовірних змін стану, про допустимість змін і навантажень на середовище, про існуючі резерви біосфери. Слід зазначити, що у відношенні екологічної інформації необхідно враховувати також вимоги та специфікації міжнародних зобов'язань. Поняття навколишнього середовища можна визначити як біологічні, біофізичні і хімічні чинники і ресурси, які оточують людину в процесі його життя. Необхідно відмітити однак, що існуючі статистичні дані відносяться до захворюваності і смертності, тобто, вони засновані на наявності хвороб.

Для інформаційної системи – основним елементом може бути запропонований „показник” або «індикатор». Індикатори можуть бути в тій або іншій мірі узагальнені в залежності від необхідності спрощення, яка, в свою чергу, залежить від цільової групи, для якої вони призначені. Рухаючись вгору по піраміді узагальнення інформація стискується з початкових даних до більш комплексних і укрупнених індексів – індикаторів. Індекси складаються на основі початкових даних із застосуванням простих функцій, таких як: складання, множення або перетворення початкових даних в укрупнений індекс. Крім того, необхідно підкреслити, що індикатори не зводяться тільки до автоматичного об'єднання даних нижчого рівня, але, і, як говорилося вище, вони можуть розраховуватись із застосуванням простих функцій на основі даних нижчого рівня.

### **Література**

1. A.O. Zaporozhets, V.S. Eremenko, R.V. Serhienko, and S.A. Ivanov, «Development of an intelligent system for diagnosing the technical condition of the heat power equipment», XIII International Scientific and Technical Conference «Computer Sciences and Information Technologies» (CSIT 2018), Lviv, p. 48–51, 2018. URL: <https://doi.org/10.1109/STC-CSIT.2018.8526742>. Accessed on: December 30, 2018.
2. Дмитренко І.А. Екологічне право України.- 2-е вид., переробл. та допов. К.: Юрінком Інтер, 2001. 352 с.

УДК 004.056

**Р. Маслій**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **СИСТЕМА БЕЗПЕКИ ДЛЯ ІОТ З ВИКОРИСТАННЯМ SIEM ТЕХНОЛОГІЙ**

UDC 004.056

**R. Maslii**

### **SECURITY SYSTEM FOR IOT USING SIEM TECHNOLOGIES**

Вже зараз технологія IoT (Internet of Things) застосовується у широкому побутовому колі та сферах бізнесу, починаючи від розумних будинків, закінчуючи пристроями в космічній промисловості. Кожного разу, коли в систему додається можливість підключення нових пристроїв, збільшуються ризики. На сьогодні немає повністю прийнятної архітектури безпеки IoT систем. Виробники часто жертвують заходами безпеки заради того, щоб як найшвидше вийти на ринок, що може призвести до серйозних проблем у майбутньому для кожного окремого користувача.[1]

SIEM (Security Information and Event Management) є ключовим компонентом корпоративної інфраструктури. Термін SIEM комбінує в собі два управління: керування мережею та керування безпекою. SEM (управління подіями безпеки) здійснює аналіз журналів і кореляції подій (часто в режимі реального часу) для протидії загрозам безпеки та інцидентам. А SIM (управління інформацією про безпеку) – збір та керування журналами та звітність для внутрішніх аудитів або дотримання вимог.

Завдяки програмному забезпеченню SIEM можливо використовувати утиліти, які допомагають оцінити вразливості згідно стандартів. Перш ніж користувач обере той чи інший інструмент для роботи, він повинен розуміти основні принципи роботи моніторингу, наприклад, інструмент повинен відокремлювати нешкідливі невдалі спроби входу від цільових атак. Адже ключовими моментами є:

1. аналіз даних у реальному часі та автоматичне оповіщення користувача;
2. ведення журналу подій;
3. інтелектуальне виявлення загроз на основі архівних даних.

Всі ці дані повинні бути доступні для пошуку та фільтрації, щоб користувачі могли легко і швидко приймати рішення стосовно подальшої роботи. Графіки та лічильники наочно представляють те, що відбувається в системі, саме тому останнім часом більш популярною стає візуалізація даних.

Більшість хмарних постачальників, які надають рішення для IoT (MS Azure, Amazon Web Services, IBM Watson IoT і т.д.), надають надійний набір API та зовнішніх сховищ даних, що можуть бути легко інтегровані в кращі в своєму роді SIEM-рішення. [2] Тобто на етапі проектування в IoT систему можна легко інтегрувати існуючу архітектуру SIEM, що зрештою покращить і надасть додаткову цінність рішенням.

#### **Література**

1. Безпека IoT починається з ідентифікації. URL: [https://iot-ssl.com.ua/iot\\_secure.html](https://iot-ssl.com.ua/iot_secure.html).
2. IoT and SIEM Integration. URL: <https://medium.com/@dtembe/iot-and-siem-integration-pt-1-6645a012bdc>.



## ЖИТТЄВИЙ ЦИКЛ ПЛАТЕЖУ

## PAYMENT LIFE CYCLE

Щоб отримати більш повне розуміння здійснення платежів, важливо розглянути життєвий цикл платіжного процесу та суб'єкти, які задіяні (див. рис. 1).



Рисунок 1. Життєвий цикл платіжного процесу

Нижче перераховано задіяні суб'єкти:

- Власник картки: Власник картки – особа, якій банк-емітент видає платіжну картку; іншими словами, це власник платіжної картки.
- Продавець: продавець – це будь-яка організація, яка прийняла платіжну картку як форму оплати своїх товарів або послуг.
- Постачальник платіжних рішень (PSP): також відомий як платіжна мережа, PSP є організацією, яка відповідає за зв'язок продавця з різними банками-еквайєрами та мережами карток.
- Банк-емітент: Банк-емітент – це фінансова установа, яка випускає платіжні картки та пропонує інші послуги своїм споживачам.
- Банк-еквайєр: банк-еквайєр – це фінансова установа, яка на підтримку торговця опрацьовує платежі, здійснені за допомогою платіжних карток.

Важливо визначити елементи, які використовуються для обробки платіжних операцій:

- ICC: пластикова картка з вбудованою схемою, яка використовується для контролю доступу до ресурсу чи послуги.
- POI: Апаратний компонент, що дозволяє здійснювати покупки платіжними картками.
- POS: Місце, де клієнт ініціалізує платіж картою.

Іншим терміном, який часто використовується, є PIN-код, який є ідентифікаційним номером, призначеним емітентом власнику картки та який використовується для автентифікації власника картки перед транзакцією в каналі CP.

УДК 004.056

**М. Мокрицький, Ю. Скоренький, к.ф.-м.н.**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ДОСЛІДЖЕННЯ ВРАЗЛИВОСТЕЙ НЕЙРОІНТЕРФЕЙСІВ**

UDC 004.056

**М. Mokrytskyi, Yu. Skorenkyu**

### **STUDY OF BRAIN-COMPUTER INTERFACES VULNERABILITY**

**Ключові слова:** інформаційна безпека, нейроінтерфейс, вразливості.

**Key words:** information security, brain-computer interface, vulnerability.

Розвиток мікроелектроніки та інформаційних технологій забезпечив умови для створення інтерфейсів для безпосередньої взаємодії між нервовою системою людини та комп'ютерними системами [1, 2]. Питання безпеки застосування нейроінтерфейсів донедавна не досліджувалися через їх малу поширність та специфіку застосування. На сьогодні, поява відносно недорогих моделей китайського виробництва та відсутність стандартизації роблять актуальними питання безпеки конфіденційної інформації, витік якої може трапитися при використанні нейроінтерфейсів.

Принцип дії нейроінтерфейсів пов'язаний з генеруванням сигналів у мозку. Згенеровані дані відображають намір користувача керування зовнішнім пристроєм. Електромагнітні хвилі, утворені електричними сигналами у мозку, реєструються електродами за допомогою різноманітних технологій, таких як електроенцефалографія або функціональна магнітно-резонансна томографія. Неопрацьовані аналогові сигнали піддаються аналого-цифровому перетворенню, щоб забезпечити подальшу обробку даних. Однією з головних цілей цього етапу є максимізація відношення сигнал/шум, щоб виміряти вихідний сигнал в якомога точнішій формі. Обробка цифрових даних необхідна для декодування запланованої дії користувача. Після цього різні моделі (наприклад, класифікатори, предиктори, регресори) або системи на основі правил визначають заплановану дію. Програми можуть надсилати необов'язковий зворотний зв'язок користувачеві, щоб генерувати сигнали мозку та, отже, нові ітерації циклу. На кожному з етапів генерується інформація, яка відображає індивідуальні особливості користувача та є конфіденційною. Програмні компоненти нейроінтерфейсів можуть мати вразливості та зазнавати атак зловмисників.

В даній роботі представлено аналіз особливостей нейроінтерфейсів та відповідних вразливостей, які можуть суттєво вплинути на функціонування цих систем.

#### **Література**

1. Bernal S.L., Celdrán A.H., Pérez G.M., Barros M.T., Balasubramaniam S. Security in Brain-Computer Interfaces: State-of-the-Art, Opportunities, and Future Challenges. *ACM Comput. Surv.* Vol. 54. P. 1–35.
2. Butsiy R., Lupenko S. Comparative analysis of neurointerface technologies for the problem of their reasonable choice in human-machine information systems. *Scientific Journal of the Ternopil National Technical University.* 2020. No. 4 (100). P. 135–148. URL: [https://doi.org/10.33108/visnyk\\_tntu2020.04](https://doi.org/10.33108/visnyk_tntu2020.04).

**АНАЛІТИКА ОПТИМІЗАЦІЇ ЧАТ-БОТА****H. Mushynska, L. Dmytrotsa****CHAT BOT OPTIMIZATION ANALYTICS**

Однією з найбільш очевидних тенденцій для чат-ботів у 2022 році є те, що окрім обслуговування клієнтів і збору даних, чат-боти використовуються в сферах, таких як маркетинг, кадри та операційна діяльність.

Дослідження мікросервісів чат-боту у додатку Telegram є однією з важливих частин аналітики для покращення оптимізації субдомених моделей. До них належать: мікросервіси авторизації користувачів, логіки чат-бота, розсилок, взаємодії з Telegram Bot API, повідомлень та файлів.

Аналітика чат-бота допомагає зрозуміти його продуктивність. Це включає такий показник, як кількість успішних взаємодій або потенційних клієнтів, створених чат-ботом. Вивчення аналітики чат-бота слід починати із загальної кількості ініційованих сеансів.

Коефіцієнт залучення користувачів – це кількість людей, які приєдналися до розмови (або виконали певну дію, наприклад отримали код знижки чат-бота), поділена на кількість сеансів чат-бота. Будь-яка форма взаємодії з користувачем може вважатися залученням.

$$\text{Коефіцієнт залучення користувачів} = \frac{\text{Кількість взаємодій з чат ботом}}{\text{Кількість запущених сесій}}$$

Передача чат-бота – це відсоток користувачів, яким чат-бот не міг допомогти і був змушений відправити до людей-агентів.

$$\text{Резервна ставка} = \frac{\text{Кількість запитів, що не обробляються ботом}}{\text{Загальна кількість взаємодій з ботом}} * 100$$

$$\text{Швидкість стримування} = \frac{\text{Кількість запитів, що успішно обробляються ботом}}{\text{Загальна кількість взаємодій з ботом}} * 100$$

$$\text{Резервна ставка} + \text{Швидкість стримування} = 100\%$$

Швидкість стримування показує, скільком людям чат-бот зумів допомогти самостійно, не загострюючи ситуацію та не передаючи її людям. Найкращий спосіб відстежувати поведінку користувачів на сайті – інтегрувати чат-бота з Google Analytics, щоб з'ясувати, чи існує кореляція між взаємодією з чат-ботами служби підтримки клієнтів, показниками відмов та іншими показниками.

Найбільше уваги потребує мікросервіси логіки чат-бота, розсилок, файлів та повідомлень. Щоб отримати максимальну віддачу від свого чат-бота, важливо вимірювати його ефективність за допомогою кількісних даних, та оптимізувати його роботу, як тільки виявлено певні недоліки за наведеними показниками.

**Література**

1. Sehl K. Engagement Rate Calculator + Guide for 2023. 2022. URL: <https://blog.hootsuite.com/calculate-engagement-rate/>.

## **РОЗВІДКА ВІДКРИТИХ ДЖЕРЕЛ ІНФОРМАЦІЇ ДЛЯ ВИЯВЛЕННЯ ЗАГРОЗ БЕЗПЕКИ БІЗНЕСУ**

**К. Nykolyn**

### **OPEN SOURCE INTELLIGENCE FOR IDENTIFYING BUSINESS SECURITY THREATS**

Світові підприємства працюють в епоху цифрової трансформації. Це дає безліч переваг для компаній: допомагає покращити клієнтський досвід, продуктивність й управління ресурсами. Але разом із цими перевагами, більш широке впровадження технологій також означає збільшення можливості компрометації даних.

Open-source intelligence (OSINT) – розвідка на основі аналізу відкритих джерел інформації) – одна з форм процесу організації та управління збором розвідувальних даних (Intelligence Collection Management), що включає їх пошук і відбір із публічних загальнодоступних джерел, добування та аналіз інформації, формування розвідувального документу для прийняття відповідного рішення [1]. Процес OSINT складається зі збору, обробки, аналізу даних та формування звіту після їх виявлення. У світі кібербезпеки OSINT найчастіше використовується на ранніх стадіях тестування на проникнення, при цьому ця інформація також доступна і суб'єктам загроз; етап розвідки забезпечує базу для пошуку вразливостей для експлуатації з технічної точки зору. Кожна організація має модифіковану структуру OSINT відповідно до своєї мети, оскільки вимоги до OSINT відрізняються від однієї організації до іншої. Проте OSINT є лише однією з усталених дисциплін збору розвідувальної інформації. Стандартного переліку дисциплін збору розвідувальної інформації не існує, однак у розвідувальному співтоваристві США існує консенсус щодо існування п'яти основних дисциплін: HUMINT, SIGINT, IMINT/GEOINT, MASINT [2], і звичайно OSINT. Варто зазначити, що деякі з цих дисциплін (або їх піддисциплін) використовуються виключно державними установами, особливо у сфері воєнної та спеціальної розвідки.

Приватні компанії головним чином розглядають застосування сучасних інструментів OSINT як ефективний спосіб ідентифікації зовнішньої інформації та виявлення внутрішніх активів, що перебувають у відкритому доступі.

У подальших дослідженнях розглядатиметься саме розвідка на основі відкритих джерел – OSINT, адже застосування даного методу може допомагати організаціям зменшити бізнес-ризик і фінансові втрати шляхом збору даних у видимій частині Інтернету й у даркнеті, моніторити у реальному часі інформацію щодо можливих атак, ідентифікувати певні внутрішні загрози компанії.

#### **Література**

1. Електронна енциклопедія Wikipedia. Англomовна версія. URL: [http://en.wikipedia.org/wiki/Open-source\\_intelligence](http://en.wikipedia.org/wiki/Open-source_intelligence)
2. Intelligence Studies: Types of Intelligence Collection. URL: <https://usnwc.libguides.com/c.php?g=494120&p=3381426>.

УДК 004.622

**Т. Патральський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ТРАНСФОРМАЦІЯ ДАНИХ У НАСТРОЮВАНІ ІНФОРМАЦІЙНІ ЗВІТИ ТА ІНФОРМАЦІЙНІ ПАНЕЛІ LOOKER STUDIO**

UDC 004.622

**T. Patralskyi**

### **DATA TRANSFORMATION INTO CUSTOMIZABLE INFORMATION REPORTS AND INFORMATION PANELS LOOKER STUDIO**

Для швидкого розуміння ситуацій в тій чи іншій сфері багато людей використовують прості системи для показу даних щоб швидко відреагувати та запровадити дії для врегулювання ситуації. Google об'єднав усі свої інструменти бізнес-аналітики під брендом Looker, тепер Looker стане обличчям усіх існуючих програм аналізу даних Google Cloud [1].

До особливостей Looker Studio належить наявність можливості показувати свою історію даних за допомогою діаграм, включаючи лінійні, стовпчасті та кругові діаграми, географічні карти, площинні та бульбашкові діаграми, таблиці даних із розбивкою на сторінки, зведені таблиці тощо [2]. Можна додавати посилання та зображення, які можна натиснути, щоб створити каталоги продуктів, бібліотеки відео та інший вміст із гіперпосиланнями.

Looker Suite містить інтегровані служби хмарної інфраструктури. Сьогодні Looker запускається в Cloud Preview. Нова версія включатиме інтегровані базові служби хмарної інфраструктури, включаючи ключові служби безпеки та управління. Google також запускає попередню версію інтеграції Looker з Google Таблицями. З часом планується також підтримувати такі інструменти візуалізації, як Salesforce Tableau [1].

Looker Studio дає можливість легко з'єднатися з будь-яким джерелом даних. Є можливість інтегруватися з багатьма рекламними платформами. До основних належать: Google, Facebook, Twitter, TikTok, Bing, Yahoo та інші. Для підключення просто потрібно мати доступ до аккаунта рекламного засобу. Є також багато SQL сервісів з яких можна вибрати оразу потрібні таблиці або написати SQL запит.

У доповіді розглянуто основні етапи трансформації відкритих даних у інформаційні панелі та звіти Looker Studio, наведено приклади практичної реалізації

#### **Література**

1. Frederick B. Google Cloud Rebrands Data Studio As «Looker Studio». 2022. URL: <https://www.searchenginejournal.com/google-cloud-rebrands-data-studio-as-looker-studio/467599/#close>.
2. Welcome to Looker Studio! 2022. URL: <https://support.google.com/looker-studio/answer/6283323?hl=en>.

УДК 338:358.5

**Ю. Петришин**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **СИСТЕМИ МЕНЕДЖМЕНТУ, МОДЕЛЬ ISO 27001**

UDC 338:358.5

**Yu. Petryshyn**

## **MANAGEMENT SYSTEMS, ISO 27001 MODEL**

Стандарт ISO/IEC 27001 визначає вимоги до побудови системи управління інформаційною безпекою (СУІБ); включає кілька аспектів безпеки: фізичну, логічну та організаційну. Входить до сімейства стандартів, починаючи від впровадження засобів контролю техніки безпеки та управління ризиками (ISO, 2020). У деяких аспектах логіка впровадження стандартів сімейства ISO 27000 також вимагає посилань на стандарти ISO 31000, що стосується (загального) управління ризиками, і стандарт ISO22301, що стосується безперервності бізнесу. Проаналізувавши сертифікацію ISO 27001 його здатність перетворюватися на організаційний, управлінський та операційний інструмент, повертаючи справжнє уявлення про IT-безпеку.

СУІБ – це систематичний підхід до управління конфіденційною інформацією компанії, щоб тримати її у безпеці. Система застосовує процеси управління персоналом, процесів і технологічних систем, і може бути впроваджено на підприємствах будь-якого розміру (ISO, 2020). Для цього є кілька причин прийняти СУІБ наприклад:

- а) Стратегічні, для відповідності уряду зі стратегічних причин, пов'язаних з управлінням корпоративною інформацією;
- б) причини, корисні для відносин з клієнтами та постачальниками;
- в) контроль використання корпоративних IT-ресурсів;
- г) організаційна ефективність управління.

Процес впровадження системи менеджменту на основі стандарту ISO 27001 є суворим і жорстким. Впроваджена система потребує складання організаційно-експлуатаційної документації, що розглядає процедури контролю та аудиту, аспекти лідерства, презентаційну систему управління, процедуру управління менеджментом, системних процедур з підтримка управління ризиками.

З точки зору чистого корпоративного управління IT, сертифікація має переваги у можливості дотримання строгих систем роботи, які дозволяють організоване управління IT-ризиком та передбачає періодичний аудит відповідності стандарту.

Сертифікація ISO 27001 наразі не дуже поширена серед компаній:

- а) процес сертифікації вимагає важливих передумов щодо обладнання організація IT-структури та процедур управління інформаційними потоками;
- б) процес сертифікації вимагає постійних зусиль IT-структури компанії та адаптація до процедур і правил, які не завжди присутні;
- в) поновлення сертифікації кожних три роки із щорічним наглядом, вимагає постійної зобов'язання, що не всі організації можуть собі дозволити.

Тому стандарт ISO 27001 не має високої дифузії через вимоги та зобов'язання щодо його впровадження ця сертифікація представляє повну систему для забезпечення інформаційної безпеки. Він охоплює технічні, організаційні та навчальні аспекти, налагодження ефективної робочої логіки та методології.

### **Література**

1. Calder, A. (2018). Implementing Information Security based on ISO 27001/ISO 27002. Van Haren Publishing.

## **ІНТЕРНЕТ РЕЧЕЙ, «ВЕЛИКІ ДАНІ» ТА АНАЛІТИКА. СТАН ТА ПЕРСПЕКТИВИ ДОСЛІДЖЕНЬ**

## **THE INTERNET OF THINGS, BIG DATA AND ANALYTICS. RESEARCH STATUS AND PROSPECTS**

Розвиток, доступність та запровадження інноваційних цифрових технологій практично у всіх сферах людської діяльності призводить до генерації безпрецедентно великих наборів та колекцій даних. Цифрові дані, сформовані з використанням різних цифрових платформ і пристроїв, у всьому світі зростають неймовірними темпами [1]. Експоненційне зростання кількості інтегрованих в фізичне середовище пристроїв з датчиками та виконавчими механізмами, підключеними через мережу Інтернет, спричиняє відповідне зростання обсягів даних, отриманих завдяки інформаційним технологіям на основі інтернету речей (IoT, англ. Internet of Things). Сумісна робота людей та машин на основі IoT підвищує їх операційну ефективність та загальну продуктивність людинно-машинних систем.

Аналітичне опрацювання даних з використанням IoT-пристроїв спрощує процеси прийняття рішень, підвищує їх ефективність та, як наслідок, підвищує якість життя [2]. Це призвело до чергового етапу інформаційно-технологічної еволюції та формування засад «розумного» світу. Інтернет речей дозволяє поєднати фізичний світ з Інтернетом, який дозволяє передавати критично-важливу інформацію швидше, ніж інформаційні системи, яка залежить від втручання людей. Дані, зібрані IoT-пристроями, характеризуються великими обсягами, мінливістю, швидкоплинністю. Інформаційні технології «Великих даних» (англ. Big Data) забезпечують їх швидке та ефективніше зберігання і аналітичне опрацювання. Аналітика великих даних (BDA, англ. Big Data Analytics) застосовує інструменти аналітичного опрацювання до великих за обсягом потоків та наборів даних, створених IoT-пристроями, щоб допомогти в прийнятті ефективних та оперативних рішень. Конвергенція IoT, BDA і хмарних інформаційно-технологічних платформ відкриває обширний перелік напрямків наукових досліджень.

На даний час доступні хмарні програмно алгоритмічні засоби, які надають можливості для ефективного аналітичного опрацювання «Великих даних» в режимі реального часу. Зокрема, це інструменти для аналізу великих за обсягом даних. Почастки вони сформовані з використанням алгоритмів машинного навчання (ML, англ. Machine Learning). Однак обширний перелік загальнодоступних та безкоштовних інструментів BDA, хмарних інформаційно-технологічних платформ та інструментів інтелектуального аналізу даних зазвичай ускладнює вибір засобів для достатньо ефективного виконання поставлених завдань. Тому потребують детальнішого та системного дослідження концепція IoT, особливості застосування IoT-пристроїв, характеристики «Великих даних», зв'язок між IoT та BDA, хмарні інструменти, що використовуються для BDA та прикладні задачі аналітичного опрацювання «великих даних» для потреб IoT-систем.

### **Література**

1. Volume of data/information created, captured, copied, and consumed worldwide from 2010 to 2020, with forecasts from 2021 to 2025 (in zettabytes). URL: <https://www.statista.com/statistics/871513/worldwide-data-created/>.
3. Nti, Isaac Kofi, et al. «A mini-review of machine learning in big data analytics: Applications, challenges, and prospects.» *Big Data Mining and Analytics* 5.2 (2022): 81–97.

## **ІНСТРУМЕНТИ АНАЛІТИЧНОГО ОПРАЦЮВАННЯ «ВЕЛИКИХ ДАНИХ»**

### **TOOLS FOR BIG DATA ANALYTICAL PROCESSING**

На даний час щодня генеруються великі за обсягом набори та колекції даних практично у всіх сферах людської діяльності. Зокрема дані надходять від соціальних мереж, інженерії, виробництва, транспорту, комерції, галузі охорони здоров'я, біомолекулярних досліджень, фізіології тощо. «Великі дані» (BD, англ. Big Data) та інноваційні методи та підходи їх аналітичного опрацювання, зокрема Big Data Analytics (BDA), змінили спосіб функціонування установ, підприємств та організацій, сформувавши при цьому обширний перелік нових перспективних напрямків досліджень для фахівців та наукової спільноти [1]. Окрім виробничих підприємств і дослідницьких установ, урядові та неурядові організації на даний час регулярно генерують великі за обсягом унікальні набори та колекції даних. Тому видобування та отримання значущої інформації із доступних «Великих даних» стало життєво важливим для підприємств, установ та організацій стало критично актуальним у всьому світі.

Інструменти аналітичного опрацювання «Великих даних» (англ. Big Data) використовуються для обробки великих за обсягом, структурованих, неструктурованих і напівструктурованих даних з метою видобування знань, бізнес-прогнозування, підвищення ефективності процесів прийняття рішень, візуалізації шаблонів тощо.

Apache Hadoop є одним із найпопулярніших інструментів аналізу даних. Він поширюється з відкритим кодом. HDFS (розподілена файлова система Hadoop) – це високоефективний компонент зберігання даних, який використовується для зберігання різнотипових та різноманітних даних, зокрема тексту, xml або json файлів, аудіофайлів, зображень та відео. Зберігання відбувається завдяки поділу даних на частини та збереження в кластерах товарних серверів [2]. Заснований на Java, Apache Hadoop характеризується високою швидкістю, оскільки окремі завдання розділяються та виконуються одночасно на розподілених серверах. Оскільки дані зберігаються на множині розподілених серверів то резервне копіювання даних доступне, навіть при виході з ладу одного окремого сервера.

Apache Spark – це розподілена інформаційна система з відкритим кодом, яка обробляє дані з використанням апаратної оперативної пам'яті. Водночас швидкість обробки даних засобами Spark відчутно перевищує швидкість Hadoop [3]. Spark зручний для системних архітекторів та розробників програмного забезпечення, оскільки для створення програм можна використовувати різні мови програмування, зокрема java, python, R, scala тощо. На даний час обширний перелік організацій, зокрема Fingra, Yelp, Zillow, gumgum, використовували Spark. Тому він став практично одним із найпопулярніших фреймворків розподіленої обробки «Великих даних».

#### **Література**

1. Islam, AYM Atiquil, et al. «Performance-based evaluation of academic libraries in the big data era.» *Journal of Information Science* 47.4 (2021): 458–471.
2. Akhtar, Nikhat, Firoj Parwej, and Yusuf Perwej. «A perusal of big data classification and hadoop technology.» *International Transaction of Electrical and Computer Engineers System (ITECES), USA* 4.1 (2017): 26–38.
4. Cao, Jian, et al. «Personalized flight recommendations via paired choice modeling.» *2017 IEEE International Conference on Big Data (Big Data)*. IEEE, 2017.



УДК 004.4

**I. Ралік**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ЗАДАЧА РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОБЛІКУ РЕАЛІЗАЦІЇ ТОВАРІВ В ТОРГІВЛІ**

UDC 004.4

**I. Ralik**

## **THE TASK OF SOFTWARE DEVELOPMENT FOR THE GOODS SALE ACCOUNTING IN RETAIL**

Актуальність розробки програмного забезпечення для обліку реалізації товарів зумовлена тим, що на сьогоднішній день галузь торгівлі є однією із найбільш розвинених в економіці України. Кількість компаній, які займаються торгівлею товарами, збільшується з кожним роком. Разом із цим зростає потреба в обліку реалізації товарів в торгівлі у електронному форматі, оскільки це дає змогу автоматизувати та пришвидшити значну кількість процесів.

До відомих програмних рішень в даній предметній області належать: 1С, ISpro, А5, Master: Бухгалтерія, Dilovod. Дебет Плюс, HugeProfit. Загальною їх характеристикою є те, що вони пропонують широкий набір функціональних можливостей, які підійдуть в основному представникам середнього та великого бізнесу. Компанії такого масштабу мають можливість витратити більше коштів на закупівлю відповідного програмного забезпечення та зробити більше інвестицій в навчання персоналу.

На сьогоднішній день внаслідок сукупності декількох причин використання програмних продуктів сімейства 1С є обмеженим. Однак, фактично багато компаній продовжують нею користуватися. Тому є важливим створення якісного українського продукту аналогічного призначення.

Для розробки програмного рішення обліку реалізації товарів в торгівлі пропонується обрати мову програмування С# та СУБД Microsoft SQL Server. Дане поєднання технологій вже зарекомендувало себе з позитивної сторони чином при реалізації проектів різного призначення та масштабів. На мові С# буде розроблено клієнтське програмне забезпечення, тоді як SQL Server відповідатиме за організацію роботи із даними.

Проект пропонується реалізувати на основі клієнт-серверної архітектури. Основними перевагами даної архітектури є безпека, централізований доступ до даних та простота [1, 2]. В склад програмного забезпечення входять:

1) сервер, на якому будуть оброблятися запити до бази даних та проводитимуться обчислення.

2) клієнтське програмне забезпечення (тонкий клієнт), яке дозволить вносити та отримувати дані з сервера.

Пропоноване програмне забезпечення повинне забезпечити реалізацію наступних функцій: довідник клієнтів, довідник товарів, облік приходу товарів, облік реалізації товарів, повернення товарів постачальну, прибуткові касові ордера, видаткові касові ордера, поточні залишки товарів, звіт по реалізації товарів, замовлення товарів, борги постачальникам, борги покупців, рух грошових коштів тощо.

### **Література**

1. Nathan Muller. Client-server Architecture and Implementation. Auerbach Publication. 2020. P. 189–190.
2. J.D. Meier, David Hill, Alex Homer, Jason Taylor, Prashant Bansode, Lonnie Wall, Rob Boucher Jr., Akshay Bogawat. «Руководство Microsoft по проектированию архитектуры приложений». 2009.

## **ЯКІСТЬ УПРАВЛІННЯ ІНФОРМАЦІЙНОЮ БЕЗПЕКОЮ ОРГАНІЗАЦІЙ**

UDC 004.056

**Revnuik O.**

### **THE QUALITY OF INFORMATION SECURITY MANAGEMENT OF ORGANIZATIONS**

За останнє десятиліття організації прагнули стати більш ефективнішими та продуктивнішими шляхом впровадження інформаційних та комунікаційних технологій на свої рішення. Технології стають все більш поширеними, їхня внутрішня цінність стає нижчою порівняно з критично важливою інформацією, яку вони опрацьовують. Такий тісний зв'язок між технологіями та бізнес-функціями виявився двигуном для різкого збільшення інцидентів і збоїв у сфері інформаційної безпеки, що призводить до значних витрат компанії. Відповідно організації краще усвідомлюють ризики інформаційної безпеки та потребу вживати відповідних заходів. Однак з такою кількістю доступних варіантів безпеки багато організацій намагаються визначити найкращі способи протидії загрозам, з якими вони стикаються.

Прагнучи захистити конфіденційність, цілісність і доступність бізнес-процесів, галузь інформаційної безпеки демонструє різноманітні набори продуктів, послуг, процесів і політик, починаючи від складних математичних алгоритмів шифрування до управління людськими ресурсами та законодавством [1]. Оскільки власники компаній не впевнені в найкращому контролі безпеки для своїх продуктів – часто розгортають якомога більше засобів захисту, незважаючи на їхню якість чи ефективність.

Насамперед організації повинні підвищувати якість шляхом раціонального впровадження засобів контролю, необхідних для мінімізації дефектів і забезпечення безперервної функціональності бізнесу. Завдяки більш точному розумінню практики управління інформаційною безпекою можна зрозуміти, яким чином застосовувати ефективні стратегії для покращення якості та зниження ризику.

Багато перших програм інформаційної безпеки покладалися в значній мірі на технологічні інновації. Такий підхід був доцільним, оскільки багато активів, які потребували захисту, також були високоякісними технологічно. При правильному використанні ці методи значно зменшують вірогідність атаки на безпеку в веб-додатках.

Однак, якими б успішними та складними не були ці технології, лише технічні підходи не можуть вирішувати проблеми безпеки з тієї простої причини, що безпека інформації – це не просто технічна проблема. Це також соціальна та організаційна проблема [2]. Національний інститут стандартів і технологій США класифікує засоби контролю інформаційної безпеки на три групи:

- Технічний контроль – традиційно включає продукти та процеси (такі як брандмауери, антивірусне програмне забезпечення, виявлення вторгнення та методи шифрування), які фокусуються в основному на захист програмного забезпечення організації та інформації, що обробляється в системі.
- Операційний контроль – включає механізми та способи усунення недоліків роботи, які можуть використовувати різні загрози; контроль фізичного доступу, можливості резервного копіювання.
- Контроль управління – політика використання, навчання співробітників та планування безперервності роботи бізнесу спрямовані на нетехнічні сфери інформаційної безпеки.

Основною метою є визначення засобів контролю, які організації в середньому впроваджують комплексно порівняно з елементами контролю, які впроваджуються погано.

Віруси та шкідливий код є одними з більш очевидних ризиків стійкості засобів безпеки, тому більшість організацій старанно захищаються від загроз, які вони створюють. Часто, веб-сайти та програмне забезпечення оснащені резервним копіювання інформації та надійним управлінням системи, що є теж досить вагомим плюсом в якості управління інформаційною безпекою[3]. Також організації зосереджені щодо технічної документації, принаймні в області розробки та підтримки свого продукту.

Проте є і погані риси захисту інформації, які притаманні багатьом компаніям. Погано реалізоване навчання персоналу для запобігання атакам соціальної інженерії, особливо тих, хто має доступ або знання до систем, що містять конфіденційну інформацію. Такі люди є потенційними жертвами атак соціальної інженерії. Організації також повинні вміти виявляти інциденти та усувати їх загрози.

Також важливо виявляти загрози в режимі реального часу і у разі порушення безпеки – система повинна сповістити відповідних осіб для реагування. Нажаль це правило часто нехтується. Організації, які мають системи виявлення загроз веб-додатків без реального часу сповіщення – схожі на будинки, наповнені детекторами диму, які не мають механізмів сигналізації. [4]

Організації повинні усвідомити, що значна частина проблем інформаційної безпеки виходить далеко за межі технологій. Навчитися оцінювати роль менш технічних засобів контролю, таких як розробка політики безпеки – грає велику роль для мінімізації ризиків здійснення атак на систему.

Впровадження всіх доступних елементів керування та захисту не є ефективним використанням ресурсів, тому організації повинні інвестувати в безпеку лише до точки, коли гранична вигода дорівнює граничній вартості. Цей принцип стосується як компанії в цілому – так і кожного працівника окремо. Можна, наприклад, оптимізувати рентабельність інвестицій, запровадивши три засоби керування на середньому рівні, а не один комплексний рівень.[5]

Сучасні організації активно прагнуть контролювати інформаційну безпеку, але вони стараються досягнути певних цілей істотно різними шляхами. Загалом, багато організації керують безпекою дещо непослідовно та поверхово. Замість того, щоб використовувати прорахований чи раціональний підхід, вони наголошують на певних елементах керування, залишаючи інші, хоча й не менш важливі, погано захищеними.

Кожній компанії слід додатково дослідити переваги поєднання різних рівнів технічного, управлінського та операційного засобів керування для досягнення справжньої цілісної безпеки від різноманітних поточних і майбутніх ризиків.

## Література

1. O.S. Saydjari, «Multilevel Security: Reprise,» IEEE Security & Privacy. Vol. 2. No. 5. 2004. P. 64–67.
2. R.T. Mercuri, «Computer Security: Quality Rather than Quantity,» Comm. ACM. Vol. 45. No. 10. 2002. P.12–14.
3. D.W Straub and R.J. Welke, «Coping with Systems Risk: Security Planning Models for Management Decision Making,» MIS Quarterly. Vol. 22. No. 4. 1998. P. 441–470.
4. G. Dhillon and J. Backhouse, «Information Systems Security Management in the New Millennium,» Comm. ACM. Vol. 43. No. 7. 2000. P. 125–128.
5. G. Stoneburner, A. Goguen, and A. Feringa, «Risk Management Guide for Information Technology Systems,» Nat'l Inst. of Standards and Technology, US Dept of Commerce. 2002. URL: [http://csrc.nist.gov/publications/nist\\_pubs/800-30/sp800-30.pdf](http://csrc.nist.gov/publications/nist_pubs/800-30/sp800-30.pdf).

УДК 004.657

**А. Романець, Г. Козбур**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ПРОБЛЕМИ АУТЕНТИФІКАЦІЇ АКАУНТІВ У СОЦМЕРЕЖАХ**

UDC 004.657

**A. Romanets, G. Kozbur**

## **ACCOUNT AUTHENTICATION PROBLEMS IN SOCIAL NETWORKS**

Авторизація та реєстрація є невід'ємною частиною більшості цифрових продуктів, зокрема соціальних мереж. Проте створення нового облікового запису чи відновлення доступу не завжди є зручним для користувача. Дизайнери продуктів не завжди докладають належних зусиль для проектування систем реєстрації, тому більшість користувачів відчувають незручності при повторному вході в обліковий запис, особливо на етапі згадування паролю для входу. Іноді сторінка реєстрації завантажується довше ніж зазвичай, це свідчить про велику кількість перевірок та функцій для валідації полів. Для деяких користувачів незручно реєструватися за допомогою e-mail, тому вони надають перевагу реєстрації за допомогою інших сервісів, які підтягнуть усі дані користувача. Для вирішення цих проблем пропонується схема реєстрації, аутентифікації та відновлення доступу до акаунта за допомогою e-mail та інших сервісів (Google, Twitter і т.д.).

Поля, що заповнюються користувачем під час реєстрації, можуть бути довільними, проте обов'язковими є поля «e-mail» і «пароль». Перше використовується як універсальний ідентифікатор, друге побічно визначає тип реєстрації. При реєстрації через інші сервіси пароль не створюється, йде перевірка чи такий користувач існує, якщо так - відбувається перевірка, за яким сервісом користувач реєструвався раніше. Якщо сервіси збігаються, то відбувається авторизація користувача. Реєстрація за допомогою сервісів є дуже зручною, тому що власні дані користувача підтягуються автоматично. У випадку реєстрації за допомогою e-mail відразу надсилається повідомлення з підтвердженням на пошту. Для зручності користувача потрібно реалізувати вивід повідомлень про помилки валідації форм та відповіді сервера на дії користувача. Для валідації різних типів полів потрібно створити окрему функцію, яка буде застосовуватись до конкретних полів за потребою - це зменшить кількість коду, що позитивно відіб'ється на швидкості роботи сайту. Після авторизації користувача краще перенаправити його на попередню сторінку. Для цього потрібно запам'ятати URL, з якого був викликаний вхід і повернути користувача на цю сторінку після успішної аутентифікації. Під час входу користувача, якщо минув термін життя сесії, але в браузері збереглися куки інших сервісів, через які авторизовувався користувач, можна дати йому просту підказку - наприклад, підсвітити сервіс або вивести поруч із сервісом фото його профілю. При потребі відновлення користувачу на e-mail прийде посилання, за яким потрібно перейти та відновити доступ до акаунту. В цьому разі користувачу не потрібно знову входити до облікового запису. Сам факт доступу до пошти з можливістю наступної зміни пароля скасовує необхідність запровадження додаткових заходів безпеки у вигляді авторизації. Тому автентифікація має відбуватися автоматично, як тільки було перевірено ключ із посилання.

### **Література**

1. Adrian W. West. Practical PHP and MySQL Website Databases: A Simplified Approach. Apress, 2018. P. 61–74.
2. Awa Melvine. Complete user registration system using PHP and MySQL database. URL: <https://cutt.ly/B1HKUA2>.

УДК 004.657

**А. Романець, Г. Козбур**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **БЕЗПЕКА СОЦМЕРЕЖІ ПІД ЧАС АУТЕНТИФІКАЦІЇ КОРИСТУВАЧА**

UDC 004.657

**A. Romanets, G. Kozbur**

## **SOCIAL NETWORK SECURITY DURING USER AUTHENTICATION**

Авторизація та реєстрація є невід'ємною частиною для роботи будь-якої соцмережі. Проте на етапі авторизації або реєстрації вебсайт є дуже вразливим, оскільки недбросовісні користувачі можуть зашкодити базі даних методом надсилання певних запитів. Тому потрібно запровадити валідацію даних та перевіряти усі дані, що надсилаються на сервер з боку клієнта. Система має бути захищена від перебору паролів зловмисниками для доступу до акаунта. Особливу увагу потрібно приділити безпеці соцмережі, від цього може залежати стабільна робота усього вебсайту.

Потрібно реалізувати функцію валідації даних як на сервері, так і на клієнті, яка буде приймати як параметри самі дані та тип валідації (включаючи регулярні вирази). Викликати таку функцію можна за кілька рядків коду. Будь-які дані, що потрапляють на сервер із клієнта, повинні бути очищені, оскільки це може вплинути на життєздатність всієї соцмережі та безпеки даних користувачів. При реєстрації через e-mail проводиться перевірка на наявність пошти в базі даних. Якщо користувач з таким e-mail вже зареєстрований, відбувається перевірка на прив'язані сервіси. До того, як користувач не підтвердив свій e-mail, його обліковий запис є тимчасовим. Це дозволяє керувати правами доступу (наприклад, таким користувачам можна обмежити набір дозволених дій), автоматично очищати базу від неактивованих облікових записів. У базі даних тимчасовий обліковий запис можна відрізнити від звичайного простою булевою позначкою або зберігати в окремій таблиці. Для реалізації підтвердження e-mail можна використати вже готові рішення. Листи можуть надсилатися з багатьох сторінок продукту, тому потрібно створити єдину функцію для цього. Враховуючи, що на сервері реалізовано очищення БД від неактивних акаунтів, потрібно встановити термін дії підтверджуючого посилання, який буде меншим, ніж період активності тимчасових акаунтів. Це виключить помилки, коли користувач намагається підтвердити пошту, а його тимчасовий обліковий запис вже видалено. Дуже важливо, щоб посилання на зміну пароля не було «вічним». Тому під час перевірки посилання на сервері, безпосередньо, перевіряється його термін дії. Однак для убезпечення від зловмисників не слід показувати повідомлення про «застаріле» посилання, вказавши просто на загальну помилку «неправильне посилання». Іноді це може створити незручність для користувачів, проте підвищить рівень безпеки. Також не можна дозволяти користувачам кілька разів скидати пароль за одним посиланням. Тому після першого переходу з листа посилання потрібно деактивувати. Для захисту від перебору паролів потрібно реалізувати функцію перевірки ір-адреси користувача та обмежити кількість спроб вводу паролю.

### **Література**

1. Adrian W. West. Practical PHP and MySQL Website Databases: A Simplified Approach. Apress, 2018. P. 61—74.
2. Awa Melvine. Complete user registration system using PHP and MySQL database. URL: <https://cutt.ly/B1HKUA2>.
3. Ruslan. Створюємо реєстрацію на сайті з допомогою PHP і MySQL. URL: <https://cutt.ly/q1HJHUN>.

## **ІНФОРМАЦІЙНІ СИСТЕМИ В ТУРИЗМІ**

## **INFORMATION SYSTEMS IN TOURISM**

Під визначенням «інформаційна система» розуміють сукупність технічних, програмних та організаційних складників, що використовуються для збору, обробки, аналізу, зберігання та використання інформації для користувача з певною метою.

За ДСТУ 2392-94: Інформаційна система – комунікаційна система, що забезпечує збирання, пошук, оброблення та пересилання інформації [1].

Підсумовуючи вище згадані поняття, можемо зрозуміти, що інформаційні системи відіграють важливу роль у життєдіяльності кожної людини, адже нас щодня оточує велика кількість інформації і нам постійно доводиться з нею «працювати». Застосування інформаційних систем в галузі туризму є один з важливих завдань діяльності в даній галузі. З їх допомогою підприємство має можливість збільшити свої економічні і якісні показники.

Індустрія туризму є найбільшим користувачем телекомунікаційних та інформаційних технологій, рівнем комп'ютерної оснащеності займає високу сходинку в діловому бізнесі. Основною причиною є сама форма інформації, яка є необхідною для подорожей. На сучасному етапі розвитку туризму виокремлюють застосування CRM – систем в плані обробки інформації. CRM – це певна ідеологія бізнесу у питанні побудови відносин із клієнтом. Але утилітарно це – певне програмне забезпечення, що автоматизує процес збору інформації про клієнта для подальшого аналізу [2].

Досить яскравим прикладом застосування CRM - систем є організація робочого процесу в мережі туристичних агентств «Поїхали з нами» [3]. Дана компанія розробила свою власну систему, яка доступна лише в агентствах цієї мережі.

Серед основних складових даної системи варто виокремити наступні:

- Чітке планування робочого дня працівників – у систему можна додавати нових клієнтів та всю необхідну інформацію про них, а також є повне планування роботи з клієнтом від моменту отримання запиту до його повернення з подорожі.

- Отримання корисних новин – публікуються всі важливі новини про роботу мережі, успіхи колег, інформація від туроператорів, тощо.

- Підвищення кваліфікації працівників – періодично по мережі проводиться навчання по різних моментах роботи в туризмі і кожен працівник повинен його пройти.

- Клієнтська база – в системі зберігається кожен запит, який надходив в турагентство, і завдяки хорошій базі, компанії мають основу для хорошого розвитку.

Дана система вважається однією з кращих серед усіх наявних туристичних систем на ринку і є вагомим чинником, що постійно вдосконалює роботу своїх менеджерів.

**Висновки.** Інформаційні системи є важливою складовою в індустрії туризму для її розвитку та організованості робочих процесів. Завдяки CRM менеджери можуть виконувати різного плану робіт з чітким розумінням поставлених завдань, ефективному плануванню роботи на кожен день, а також отриманню різного роду інформації, яка є важливою для подальшої діяльності. В свою чергу ефективність роботи позначається на фінансовому стані підприємства.

### **Література**

1. ДСТУ 2392-94 Інформація та документація. Базові поняття. URL: <https://zakon.rada.gov.ua/rada/show/v0114774-17#Text>.
2. Ганущак-Єфіменко Л. М. CRM-система як ефективний інструмент розвитку готельного бізнесу в Україні. Вісник Київського національного університету технологій та дизайну. Серія: Економічні науки. № 4. Київ. 2017. С. 51–56.
3. Сайт мережі туристичних агентств «Поїхали з нами». URL: <https://www.pohalisnami.ua/ua>.

**ПОЄДНАННЯ ТЕХНОЛОГІЇ ДОПОВНЕНОЇ РЕАЛЬНОСТІ ТА  
ФАКТОГРАФІЧНОГО ПОШУКУ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ  
КОНСОЛІДАЦІЇ СОЦІО-КОМУНІКАЦІЙНИХ РЕСУРСІВ «РОЗУМНОГО  
МІСТА» В МУЗЕЙНІЙ ДІЯЛЬНОСТІ**

**COMBINATION OF AUGMENTED REALITY TECHNOLOGY AND  
FACTOGRAPHICAL SEARCH OF THE INFORMATION SYSTEM FOR THE  
CONSOLIDATION OF SOCIO-COMMUNICATION RESOURCES OF THE SMART  
CITY IN MUSEUM ACTIVITIES**

Підхід до фактографічного пошуку [2] в інформаційній системі для консолідації соціо-комунікаційних ресурсів «Розумного міста» [1] ефективно можна використати в музейній діяльності з застосуванням технології доповненої реальності.

Пропонована можливість передбачає надання актуальних описів музейних елементів, використовуючи оновлюване сховище консолідованої інформаційної системи. Для перегляду описів в приміщеннях музеїв слугуватимуть мультимедійні девайси (смартфон, шолом чи окуляри доповненої реальності), а в віртуальних відтвореннях музеїв – інформаційні блоки, що з'являтимуться при активації.

Оскільки експерти не володіють всеохоплюючою інформацією в тому обсязі, що здатне вміщувати і накопичувати сховище даних та не в змозі побудувати стільки ж зв'язків між фактами та висновків, що здатна аналітична система, то її застосування має сенс для пришвидшення актуалізації та системного оновлення існуючої інформації. Проте, вагомою перешкодою повністю автоматичного оновлення інформаційних елементів, базованих на документованих та історичних джерелах, є можливість суперечностей й ситуацій, що не піддаються чіткій кількісно-ймовірнісній оцінці та побудова хибних причинно-наслідкових зв'язків і висновків. Тому пропонується впровадити опцію додаткової експертної перевірки побудованих системою висновків.

В результаті, так як інформація буде в єдиному сховищі (консолідованого ресурсу) та різнотипною (міститиме інформаційні елементи всіх ресурсів), то це надаватиме більшої точності аналітичній підсистемі фактографічного пошуку та дозволить залучати меншу кількість допоміжних робочих ресурсів для підтримки актуальності кожного ресурсу окремо. А також, дозволить зробити оновлення музейної інформації менш ресурсозатратним, за рахунок доступу до цифрових джерел з залученням технології доповненої реальності.

**Література**

1. Пасічник В. В., Кунанець Н. Е., Дуда О. М., Липак Г. І., О Мацюк. В., Семенюк В. В. Актори та діаграми прецедентів системи консолідації соціокомунікаційних інформаційних ресурсів «Розумних міст». Науковий вісник НЛТУ України. 2017. Вип. 27 (10). С. 129–136.
2. Семенюк В. В. Застосування фактографічного підходу для пошуку пов'язаних та актуальних даних в системі консолідації соціокомунікаційних інформаційних ресурсів з використанням технологій обробки природної мови та великих даних: Збірник тез доповідей VIII Міжнародної науково-технічної конференції молодих учених та студентів «Актуальні задачі сучасних технологій», 27–28 листопада 2019 року. Т : ТНТУ, 2019. Том 2. С. 84–85.

УДК 004.056

**С. Сербичанський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

**ДОСЛІДЖЕННЯ ВИМОГ ДО ФІЗИЧНОГО ТА ПРОГРАМНОГО ЗАХИСТУ  
ІНФОРМАЦІЇ НА ОБ'ЄКТАХ КРИТИЧНОЇ ІНФРАСТРУКТУРИ В  
УМОВАХ ЗАГРОЗ І ОБМЕЖЕНЬ**

UDC 004.056

**S. Serbychanskyi**

**STUDY OF REQUIREMENTS FOR PHYSICAL AND SOFTWARE  
PROTECTION OF INFORMATION AT CRITICAL INFRASTRUCTURE  
OBJECTS UNDER THE CONDITIONS OF THREATS AND LIMITATIONS**

Електромережі є основою в сучасному світі, засоби зв'язку, банківська ж система, медична система та більшість інших важливих систем які не можна увити без систем електропостачання. Передбачається, що до 2050 року світове електроспоживання зросте втричі.

Переважно атомні електростанції покривають базову потребу використання (тобто використання електроенергії яке є постійним). Гідроелектростанції покривають в першу чергу перепади в потребах живлення до прикладу час пік.

Теплові електростанції покривають постійні та незначні перепади електроенергії. Звісно існують різні конфігурації мереж в залежності від потреб. Про те швидке інформування місць генерації електроенергії є досить складною задачею. Погіршує проблему саме використання зеленої електроенергетики. Таким чином ж інформування місць генерації та швидка взаємодія набуває основного значення.

Тому різними державами активно виділяється значна кількість коштів на розробку нової системи, що повинна пов'язати місця виробництва електроенергії та їх використання. Нові мережі мають децентралізовану цифрову інфраструктуру, що називаються smart grid.

Smart Grid об'єднує кілька добре відомих, але відмінних галузей, а саме електроенергетику, інформаційні технології та зв'язок.

Кожен пристрій у новій сітці, ймовірно, матиме власну IP-адресу та використовуватиме для зв'язку такі протоколи, як TCP/IP. Таким чином, вони будуть уразливі до подібних загроз безпеці, з якими стикаються сучасні комунікаційні мережі, однак ставки набагато вищі.

Для автентифікації пристрою використовуються алгоритми з відкритим ключем які не є досконалыми та мають ряд недоліків при використанні в мережах smart grid.

Альтернативою використанню ресурсоємної автентифікації відкритого ключа є система на основі дерев Меркла. Дерев Merkle пропонують недорогу автентифікацію для mesh-клієнтів. Порівняно з відкритим ключем, вони легкі та швидкі для генерації та пропонують такий самий, а у деяких випадках, кращий захист.



## РЕАЛІЗАЦІЯ ПАРАЛЕЛЬНОЇ ОБРОБКИ ДАНИХ В МОВІ ПРОГРАМУВАННЯ JAVASCRIPT

UDC 004.4

V. Serbin

## IMPLEMENTATION OF PARALLEL DATA PROCESSING IN JAVASCRIPT PROGRAMMING LANGUAGE

Станом на сьогодні, мова програмування JavaScript більше не обмежується браузером, а використовується для створення програмного забезпечення у будь-якому пристрої, від серверів до пристроїв Інтернету речей. Багато з написаних програм є важкими і можуть отримати значне підвищення продуктивності від використання паралельних обчислень. В загальному, причина розпаралелювати JS така ж, як і причина розпаралелювати будь-яку іншу мову: закон Мура вмирає, а багатоядерна архітектура захоплює світ [1].

Використання API Web Workers, ймовірно, є єдиним способом досягти справжньої мультипроцесорності в Javascript [2]. Web Worker дозволяють користувачеві паралельно запускати JavaScript без втручання в інтерфейс користувача. Робочий сценарій буде завантажено та запущено у фоновому режимі, повністю незалежно від сценаріїв інтерфейсу користувача. Це означає, що Workers не мають доступу до елементів інтерфейсу користувача, таких як DOM і поширених функцій JS, як-от getElementById (але можуть здійснювати виклики AJAX). Основний варіант використання API Web Worker полягає у виконанні обчислювально дорогих завдань у фоновому режимі, без процесу переривання або переривання взаємодії користувача [1].

Щоб реалізувати Worker, необхідно створити його екземпляр за допомогою коду, який він запускатиме [2]:

```
const worker = new Worker(«worker.js»);
```

Web Workers спілкуються з основним документом/основним потоком через техніку передачі повідомлень. Передача повідомлень здійснюється за допомогою API postMessage [1]:

```
worker.postMessage(num);
```

Згідно зі специфікацією існує два типи Web Worker: спільні та виділені.

Web Worker за замовчуванням називається виділеним. Цей тип робочого файлу доступний лише зі сценарію, який його створив, тоді як спільний робочий файл можна отримати з кількох сценаріїв.

Принцип взаємодії Web Workers та основного потоку наведено на рисунку 1.

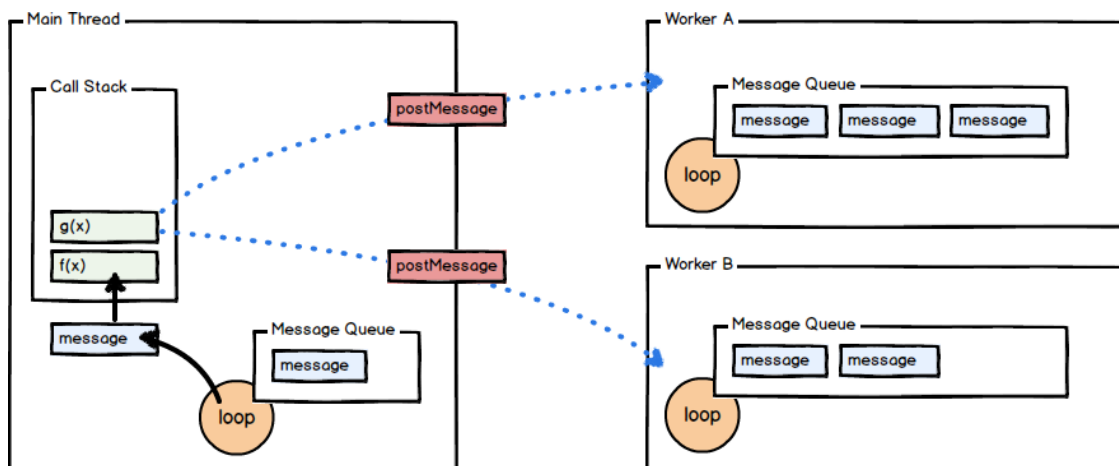


Рисунок 1. Схема взаємодії Web Workers та основного потоку

Для спільного Web Worker потрібен інший конструктор: SharedWorker. Спільний робочий файл доступний кількома сценаріями, навіть якщо до них звертаються різні вікна, iframe або навіть робочі.

Прослуховування події відбувається через обробник «onmessage()», в якому e.data міститиме передане значення

Цей процес працює в обох напрямках, тож можна повернути деякі дані назад із робочого коду у основний потік [2]. Для цього використаємо у коді worker-у «postMessage()» з результатом, а у основному потоці використовуємо «worker.onmessage» для прийому даних.

Отже, потоки Web Worker допомагають нам розвантажити інтенсивні завдання ЦП з циклу подій, щоб виконувати їх паралельно без блокування. Робочий потік виконує частину коду відповідно до вказівок батьківського потоку окремо від батьківського та інших робочих потоків. Кожен робочий потік має власне ізольоване середовище, цикл подій, чергу подій тощо. Робочий і батьківський потік можуть спілкуватися один з одним через канал обміну повідомленнями, а також робочі потоки дають нам можливість запускати кілька потоків в одному процесі [3]. Крім утримання циклу подій від виконання трудомістких операцій ЦП можемо використовувати пул робочих потоків для розділення та паралельного виконання важких операцій ЦП з ціллю підвищення продуктивності нашої програми [3].

### **Література**

1. Розпаралелювання JavaScript для розваги та прибутку. URL: <https://www.codementor.io/@madhugnadig/parallelizing-javascript-for-fun-and-profit-naxmo4lam>.
2. Паралельна обробка в JS. URL: <https://advancedweb.hu/parallel-processing-in-js/>.
3. Паралельна обробка в Node.js з використанням робочих потоків. URL: <https://deepsources.io/blog/nodejs-worker-threads/>.

## **ПОРІВНЯННЯ ПРЕТРЕНОВАНИХ МОДЕЛЕЙ ДЛЯ ДЕТЕКЦІЇ ОБ'ЄКТІВ**

### **COMPARATION OF THE PRETRAINED MODELS FOR OBJECT DETECTION**

Детекція об'єктів – це одне з найважливіших завдань комп'ютерного бачення, завдання якого знаходження об'єктів певних класів на заданих зображеннях. Для вирішення цієї не простої задачі існує два підходи:

1. Використання власноруч налаштованих фільтрів для виділення певних об'єктів на зображеннях. Цей спосіб не є ефективним, адже, зазвичай, об'єкти візуально відрізняються як і кольорами так і формою. Налаштовування фільтрів під кожен випадок може тривати дуже довго і не завжди успішно;

2. Використання моделей машинного навчання, які автоматично налаштовують безліч фільтрів використовуючи для цього промарковані зображення. Цей підхід є більш універсальним, адже він враховує всі можливі варіанти об'єктів, а також, володіє властивістю генералізації, що допомагає справлятися з раніше небаченими об'єктами.

Враховуючи те, що перший метод займає багато часу та потребує багато зусиль – для вирішення задачі детекції об'єктів буде використовуватись, в основному, другий спосіб.

Ця доповідь стосується теми «Порівняння претренованих моделей для детекції об'єктів». Для того, щоб отримати модель, яка здатна знаходити потрібні об'єкти на зображенні потрібно багато промаркованих даних, а також потужні обчислювальні ресурси. Для економії ресурсів та для покращення точності моделей використовується техніка під назвою «Передача навчання». Згідно з дослідженням Proper Reuse of Image Classification Features Improves Object Detection (1, Cristina Vasconcelos, Vighnesh Birodkar, Vincent Dumoulin, 2022) з правильним використанням попередньо тренованих моделей можна досягнути кращих результатів швидше чим без них.

Під час дослідження готових імплементацій моделей для детекції об'єктів було знайдено багато рішень які відрізняються як і датою публікації, точністю і швидкістю. Для вирішення поточної задачі не потрібно враховувати швидкість, адже швидкість роботи не залежить від реального часу. Для порівняння точності моделей машинного навчання потрібно використовувати одні й ті ж дані. Одним з найпопулярніших відкритих наборів даних з готовими промаркованими об'єктами є Microsoft COCO (2, Tsung-Yi Lin Michael Maire Serge Belongie Lubomir Bourdev Ross Girshick, James Hays Pietro Perona Deva Ramanan C. Lawrence Zitnick Piotr Dollar, 2015) – це набір даних, який містить 328 тисяч картинок повсякденних об'єктів та людей.

Для порівняння було обрано такі моделі:

- InternImage;
- EVA;
- Co-DETR;
- FD-SwinV2-G.

Для кращого визначення моделі також буде враховуватись набір даних для семантичної сегментації ADE20K (3, Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso and Antonio Torralba. Computer Vision and Pattern Recognition (CVPR), 2017).

Для порівняння на COCO наборі даних використовувалась метрика box AP, а для порівняння на ADE20K – Validation mIoU.

На рисунку 1 зображена порівняльна таблиця моделей на обох наборах даних.

Model name	COCO test-dev	ADE20K
InternImage	65.0	62.9
EVA	64.7	62.3
Co-DETR	64.5	-
FD-SwinV2-G	64.2	61.4
YOLOR-P6	55.4	-

**Рисунок 1.** Порівняльна таблиця

Як видно з таблиці, більшість порівнюваних моделей дуже близькі по значеннях. Модель YOLOR-P6 має найгірший результат, але моделі YOLO є найпопулярнішими моделями, які використовуються для детекції об'єктів через те, що вони мають декілька конфігурацій з різною швидкодією, що дозволяє налаштувати їх під потрібну задачу.

Отже, є багато моделей та наборів даних для їх порівнянь. Попри те, що є безліч моделей для знаходження об'єктів потрібно використовувати ресурси саме на ті, які дають хороші результати на схожих задачах. Всі моделі, які розглянуті вартують подальшого дослідження вже в рамках поточної задачі знаходження спортивних об'єктів на кадрах відеозапису.

### Літератури

1. Cristina Vasconcelos, Vighnesh Birodkar, Vincent Dumoulin. Proper Reuse of Image Classification Features Improves Object Detection, 2022. URL: <https://arxiv.org/abs/2204.00484>.
2. Microsoft COCO: Common Objects in Context, Tsung-Yi Lin, Michael Maire, Serge Belongie, Lubomir Bourdev, Ross Girshick, James Hays, Pietro Perona, Deva Ramanan, C. Lawrence Zitnick, Piotr Dollár, 2015. URL: <https://arxiv.org/abs/1405.0312>.
3. Semantic Understanding of Scenes through the ADE20K Dataset, Bolei Zhou, Hang Zhao, Xavier Puig, Tete Xiao, Sanja Fidler, Adela Barriuso, Antonio Torralba, 2018. URL: <https://arxiv.org/abs/1608.05442>.
4. InternImage: Exploring Large-Scale Vision Foundation Models with Deformable Convolutions, Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, Xiaogang Wang, Yu Qiao, 2022. URL: <https://arxiv.org/abs/2211.05778>.
5. EVA: Exploring the Limits of Masked Visual Representation Learning at Scale, Yuxin Fang, Wen Wang, Binhui Xie, Quan Sun, Ledell Wu, Xinggang Wang, Tiejun Huang, Xinlong Wang, Yue Cao, 2022. URL: <https://arxiv.org/abs/2211.07636v1>.
6. DETRs with Collaborative Hybrid Assignments Training, Zhuofan Zong, Guanglu Song, Yu Liu, 2022. URL: <https://arxiv.org/abs/2211.12860v1>.
7. Contrastive Learning Rivals Masked Image Modeling in Fine-tuning via Feature Distillation, Yixuan Wei, Han Hu, Zhenda Xie, Zheng Zhang, Yue Cao, Jianmin Bao, Dong Chen, Baining Guo, 2022. URL: <https://arxiv.org/abs/2205.14141v3>.
8. You Only Learn One Representation: Unified Network for Multiple Tasks, Chien-Yao Wang, I-Hau Yeh, Hong-Yuan Mark Liao, 2021. URL: <https://arxiv.org/abs/2105.04206v1>.

**АНАЛІЗ КОНЦЕПЦІЇ ВСЕОСЯЖНОГО ІНТЕРНЕТУ – ІоЕ**

UDC 004.7

A. Stanko,

**ANALYSIS OF THE CONCEPT OF THE INTERNET OF EVERYTHING – IoE**

Концепція «всеосяжного інтернету» або «інтернету усього» (англ. Internet of Everything – IoE) з'єднує мільярди людей, пристроїв, об'єктів і речей з Інтернетом для надання автономних послуг. IoE отримує переваги від цінності, створеної комплексним впливом зв'язку людей, процесів, речей і даних. Взаємодія різномірних компонентів сприяє безперервному обміну даними для надання послуг майже в усіх сферах життя. Крім того, що ця концепція забезпечує величезні переваги в автономному наданні послуг, з іншого боку страждає від проблем реалізації, архітектурних міркувань, а також уразливості безпеки та конфіденційності.

IoE – це концепція, що розвивається, яка охоплює ширший погляд на Інтернет речей (IoT) і передбачала зміну технології шляхом імплементації мільярдів давачів у мільярди пристроїв, машин і звичайних об'єктів, таким чином роблячи їх «розумнішими» за рахунок розширення можливості комунікації.

IoT зосереджується лише на фізичних об'єктах, взаємопов'язаній системі речей (пристроїв), здатних різним ступенем обробки, сприйняття, приводу в дію, передачі та обміну даними через спільну платформу Інтернету. З іншого боку, IoE має на меті підключити все до Інтернету, щоб полегшити прийняття рішень і пропонувати інтелектуальні послуги. В IoE інтелектуальні підключення до Інтернету не обмежуються комп'ютерами, планшетами та смартфонами, а скоріше IoE зосереджується на речах із підвищеною обізнаністю про контекст, енергонезалежністю та потужністю обробки, з'єднуючи людей, процеси та речі, щоб вони могли співпрацювати по-новому, зрештою генеруючи цінну інформацію для прийняття «розумніших» та оптимальніших рішень. Незважаючи на те, що IoT і IoE є різними термінами, проте обидва мають певну схожість; По-перше, обидві системи розподілені, тобто вони не мають єдиного центру, кожен вузол працює як невеликий центр керування, здатний самостійно виконувати певні завдання. По-друге, обидві системи вразливі до проникнення та кібератак. IoE набагато більш сприйнятливий до порушень безпеки, оскільки до мережі підключено більше пристроїв.

Термін IoE був придуманий і використаний CISCO в 2013 [1] і визначив IoE як «Інтернет усього (IoE) об'єднує людей, процеси, дані та речі, щоб зробити мережеві зв'язки більш актуальними та цінними, ніж будь-коли раніше, перетворюючи інформацію на дії, які створюють нові можливості, багатший досвід, а також безпрецедентні економічні можливості для підприємств, окремих осіб і країн».

У середовищі IoE, переходячи на крок далі від керування підключеними пристроями за допомогою дистанційного керування, ми хочемо, щоб вони інтуїтивно знали, що ми хочемо від них робити. IoE змінить те, як ми живемо, купуємо, продаємо, споживаємо та ведемо наше життя завдяки своїм унікальним вимірам цілісності, своєчасності, навчання та автоматичності (STAL) [2]. IoE є одним із чотирьох технологічних середовищ клієнтів.

Інтернет усього можна охарактеризувати такими поняттями:

- Безперервність;
- Своєчасність;
- Доступність;
- Навчання.

Безперервність означає, що в IoE інформація про елементи, що представляють інтерес, і місця, які представляють інтерес для окремих осіб (Люди, процеси, дані та речі), а також взаємодії між особами та їхніми елементами будуть доступні незалежно від того, де знаходяться особа та елементи, який пристрій чи машина, яку вони використовують. Своєчасність означає,

що історія інформації про елементи з минулого зберігається в ІоЕ і доступна в будь-який час. Доступність інформації в будь-який час потребує передбачуваності, одночасно враховуючи поняття реального часу, саме вчасно та саме в потрібний час. Навчання означає, що ІоЕ вивчатиме статичну та динамічну інформацію, а також зв'язуючи причину та наслідок. Навчання разом з інформацією в режимі реального часу формує датчики, які контролюють місця та об'єкти людей для автоматичної обробки даних і автоматичного прийняття відповідних рішень, включаючи переказ грошей, хімічні реакції, фізичні переміщення тощо.

Пристрої з підтримкою ІоЕ можуть допомогти збагатити життя людей завдяки розумним і автоматизованим бізнесам і промисловим процесам. Ця мета може бути досягнута шляхом задоволення трьох очікувань:

- Масштабованість;
- Інтелект;
- Різноманітність.

Масштабованість забезпечує створення масштабованої мережевої архітектури з повсюдним покриттям. Масштабована мережа ІоЕ потребує широкого покриття, масового доступу та повсюдного з'єднання, щоб бути здатною задовольнити різноманітні вимоги зв'язку для різних географічних сценаріїв, включаючи сільську місцевість, місто, землю, підводний світ, повітря та космос. Інтелект забезпечує прийняття інтелектуальних рішень і дій для всіх пристроїв в ІоЕ. ІоЕ вимагає збору великої кількості даних зі своєї широкої та масштабованої мережі, витягання із зібраних даних цінної інформації, включаючи інтелектуальні команди чи рішення, і, нарешті, використання видобутої інформації для прогнозування чи аналізу, інтелектуальних дій і засобів керування для всього. Різноманітність вказує на підтримку різноманітних програм. [3]

Додатки ІоЕ, здатні обслуговувати автоматизовані та орієнтовані на людей процеси, в основному базуються на масштабованості та інтелекті системи ІоЕ, оскільки вони є обов'язковими для досягнення безпеки, обчислювальних можливостей, енергоефективності та продуктивності мережі. Ці три очікування можуть бути досягнуті шляхом подолання деяких обмежень, які є реальною перешкодою для успіху ІоЕ. Таким чином, розгортання великої кількості термінальних вузлів, вузлів доступу до мережі та обчислювальних засобів може призвести до всемогутньої ролі ІоЕ. Однак проблему серйозного дисбалансу між обмеженими ресурсами та трьома очікуваннями, які виникнуть через споживання великої кількості ресурсів, можна вирішити, дотримуючись принципу проектування за вимогою, за якого ресурси використовуються ефективно.

Можна стверджувати що із зростанням обчислювальних можливостей, покращення всеохопного зв'язку, зокрема комунікації на основі ІоТ, концепція інтернету усього буде активно розвиватись в найближче десятиліття.

### Література

1. D. Evans. (2012, Nov.) Internet of everything: Harnessing an exponentially more powerful internet. URL: <https://blogs.cisco.com>.
2. D. Sudharshan, Marketing in Customer Technology Environments: Prospective Customers and Magical Worlds. Emerald Group Publishing, 2020.
3. Y. Liu, H.-N. Dai, Q. Wang, M. K. Shukla, and M. Imran, «Unmanned aerial vehicle for internet of everything: Opportunities and challenges,» Computer Communications, 2020.
4. H.-N. Dai, H. Wang, G. Xu, J. Wan, and M. Imran, «Big data analytics for manufacturing internet of things: opportunities, challenges and enabling technologies,» Enterprise Information Systems. Vol. 14. No. 9–10. P. 1279–1303.
5. M. Stoyanova, Y. Nikoloudakis, S. Panagiotakis, E. Pallis, and E. K. Markakis, «A survey on the internet of things (iot) forensics: Challenges, approaches and open issues,» IEEE Communications Surveys & Tutorials, 2020.
6. K. Mekki, E. Bajic, F. Chaxel, and F. Meyer, «A comparative study of lpwan technologies for large-scale iot deployment,» ICT express. 2019. Vol. 5. No. 1. P. 1–7.
7. W. Rafique, L. Qi, I. Yaqoob, M. Imran, R. ur Rasool, and W. Dou, «Complementing iot services through software defined networking and edge computing: A comprehensive survey,» IEEE Communications Surveys & Tutorials, 2020.

## **ТЕХНОЛОГІЇ ВПЛИВУ СОЦІАЛЬНИХ МЕРЕЖ НА ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ**

## **TECHNOLOGIES OF THE INFLUENCE OF SOCIAL NETWORKS ON ENSURING INFORMATION SECURITY**

Науково-технічна революція початку XXI сторіччя спричинила в усьому світі глибокі системні перетворення. Передусім завдяки поєднанню досягнень у сфері новітніх інформаційно-комунікаційних технологій (ІКТ)[1] із надбаннями, що постали на базі стрімкого розвитку інформаційно-телекомунікаційних систем (ІТС), сформувалися принципово нові глобальні субстанції – інформаційне суспільство, а також інформаційний та кібернетичний простори, які мають нині практично необмежений потенціал і відіграють провідну роль в економічному та соціальному розвитку кожної країни світу.

Разом з тим, феномен «соціальних медіа» пов'язаний більше з культурним позиціонуванням, ніж з технологічними можливостями. Соціальні мережі швидко набули популярності і стали одним із найкращих способів проведення вільного часу та залучення клієнтів. Вони дозволяють людям підтримувати зв'язок з друзями та родиною, надають можливості пошуку кар'єрних можливостей, обміну власними думками, почуттями та ідеями в Інтернеті. Однак, використання соціальних медіа також може мати негативний вплив на особисте життя людини.

Із зростанням репутації LinkedIn, Facebook, Telegram зростають і ризики їх використання. Вільне спілкування не є безкоштовним. Зменшивши вартість інформації, втрачається її цінність і збільшується ймовірність її фальсифікації. Щоб відновити здоров'я інформаційної екосистеми, необхідно зрозуміти вразливі місця перевантаженого розуму та те, як можна використати економіку інформації, щоб захистити людей від введення в оману.

Згідно матеріалів The New York Post щодня лише у Facebook зламують 160 000 облікових записів, а дослідники Університету Фенікса що близько 66% облікових записів громадян США було хоч раз зламано (це означає, що якщо ви знаєте ім'я собаки свого менеджера соціальних мереж, ви на півдорозі до брут форсінгу облікового запису вашої організації). На відміну від інших активів, служби безпеки не можуть відключити зламаний обліковий запис у соціальних мережах, тобто зловмисник може зберігати контроль протягом годин, якщо не днів. Вартість? Кожна секунда, коли ви не контролюєте свій обліковий запис, спричиняє каскад вірусної інформації, що призводить до шкоди стосункам із брендом і клієнтом, втрати бізнесу, кошмарів зі зв'язків із громадськістю та витрат на підтримку клієнтів.

В доповіді наведено детальний аналіз[2] найбільш відомих атак на соціальні мережі. Надано пропозиції щодо забезпечення інформаційної безпеки в інтернет середовищі. До найбільш відомих атак належать:

- фішинг,
- викрадення особистих даних,
- розповсюдження зловмисного програмного забезпечення,
- соціальна інженерія та
- компрометація облікових даних банківського або системного входу.

Зокрема зазначено, що багато нападників координують свої зусилля серед білого дня. Відомо, що атаки розподіленої відмови в обслуговуванні (DDoS) використовують певний хештег Twitter для координації атаки. Зловмисники, особливо хактивісти, краудсорсингують учасників атак через кампанії з хештегами та керують DDoS-атакою в Twitter, публікуючи IP-адреси,

домени, інструменти атаки, час атаки та бажану ціль. Оскільки атаки використовують громадські місця для участі, команди безпеки можуть підготувати стратегію захисту, наприклад, ховати вхідні запити або координувати дії з мережевими командами, професійними службами та постачальниками послуг Інтернету (ISP). Команди безпеки також можуть стежити за розмовами учасників загрози, щоб виявити, чи згадується їх організація. Це одні з найчистіших, найдешевших, найактивніших і доступних у реальному часі даних про загрози. Дивно, але така публічна балаканина досить поширена. Аналізуючи, хто говорить і контекст ключової фрази, служби безпеки можуть отримати вирішальну систему раннього попередження проти атак. Зловмисники часто афішують або хваляться своїми успіхами в соціальних мережах. Вони також рекламують викрадені дані, які можуть продавати. Подібно до того, як соціальні медіа є основною рушійною силою діяльності легального ринку, ними також користуються продавці на чорному ринку. Організації можуть інтегрувати конфіденційну інформацію, виявлену на сайтах соціальних мереж, у фреймворки DLP, щоб швидше визначати, коли стався злом, і ефективніше починати дії з усунення. Витоку або викрадених даних частіше торгують у відкритому доступі, ніж усвідомлюють. Якщо облікові дані співробітників або конфіденційні файли виявлені в соціальних мережах або цифрових каналах, наприклад на сайтах вставлення, служби безпеки можуть оновити тренінги компанії, скинути облікові дані співробітників або відстежити, де заходи запобігання потенційній втраті даних (DLP) не змогли запобігти конфіденційним файлам, таким як медичні записи, інтелектуальну власність або інформацію облікового запису від виходу з мережі.

Для шахрая соціальні медіа є новим потужним інструментом для використання дуже специфічної масової групи користувачів, наприклад підписників певного бренду. Соціальні мережі [3] дозволяють шахраям націлюватися на цих користувачів, оскільки списки підписників бренду та залучення користувачів до фірмового хештегу є загальнодоступними. Таким чином, шахрай має безпрецедентну можливість отримати список жертв і розпочати цілеспрямовану атаку. Шахрайство, орієнтоване на клієнтів, зазвичай обіцяючи винагороду за певну вартість участі та використовує фальшиві логотипи бренду або історії успіху з інших облікових записів маріонеткових учасників, які вказують на те, що шахрайство є «законним». Ці шахрайства процвітають у соціальних мережах, тому що їх дуже легко створити та поширювати серед цільової аудиторії у великих масштабах. Навіть нетехнічний шахрай може створити групу фальшивих облікових записів, створених для коментування один одного та надання довіри, маючи лише підключення до Інтернету з будь-якої точки світу.

Соціальні медіа є неминучою константою для ведення бізнесу в сучасному світі. Оскільки маркетологи, рекрутери, продавці та рекламодавці постійно розширюють свою присутність, групи безпеки повинні працювати разом з ними, щоб гарантувати, що це робиться безпечно та надійно. Щоб усунути ризики соціальних мереж, служби безпеки повинні тісно співпрацювати з кількома іншими відділами. Усі інші департаменти стикаються з ризиками в соціальних мережах, і тепер командам безпеки доручено усунути ризики, одночасно забезпечуючи безпечне використання каналів соціальних мереж. Найголовніше, що команди безпеки повинні очолити цю ініціативу. Ризики в соціальних мережах залишаються.

Для зменшення ризиків[3] необхідно реєструватися не у всіх соцмережах, а лише у тих, які викликають довіру та пропонують надійні механізми аутентифікації і розмежування доступу до особистої інформації користувача, особливу увагу слід приділяти посиланням, які надходять від інших користувачів – вони можуть бути частиною фішингової чи фармінгової атаки.

## Література

1. Інформаційна та кібербезпека: соціотехнічний аспект. URL: [http://ippi.org.ua/sites/default/files/dovsmib\\_46\\_2\\_2015\\_0.pdf](http://ippi.org.ua/sites/default/files/dovsmib_46_2_2015_0.pdf) (дата звернення: 10.09.2022).
2. Рапорт компанії Norton. URL: [https://www.yle.fi/tvuutiset/ uutiset/upics/liitetiedostot/norton\\_raportti.pdf](https://www.yle.fi/tvuutiset/ uutiset/upics/liitetiedostot/norton_raportti.pdf) (дата звернення: 12.09.2022).
3. Забезпечення інформаційної безпеки у соціальних мережах. URL: [http://dspace.kntu.kr.ua/jspui/bitstream/123456789/4999/1/AUConferenceCyberSecurity\\_November2016\\_p204.pdf](http://dspace.kntu.kr.ua/jspui/bitstream/123456789/4999/1/AUConferenceCyberSecurity_November2016_p204.pdf) (дата звернення: 20.09.2022).



УДК 004.056

**Д. Урбан**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АНАЛІЗ ЗАГРОЗ КОМП'ЮТЕРНИХ СИСТЕМ**

UDC 004.056

**D. Urban**

## **ANALYSIS OF COMPUTER SYSTEM THREATS**

Уразливість нульового дня – це слабе місце в комп'ютерній системі, яким може скористатися зловмисник і яке не помічають уражені сторони. Атака нульового дня – це спроба загрози проникнути, пошкодити або іншим чином скомпрометувати систему, уражену невідомою вразливістю. За характером нападу жертва не матиме засобів захисту, тому ймовірність успіху є високою.

Професіонали з IT-безпеки ще ніколи не стикалися з такими загрозами, чи то від величезного зростання віддаленої роботи, чи від агресивних хакерів, спонсорованих національною державою, таких як ті, хто причетний до зламу SolarWinds. Незважаючи на те, що завжди знайдуться нові діри, які потрібно закрити, уразливості системи безпеки зазвичай виникають через кілька тих самих причин: невиправлені вразливості, неправильні конфігурації чи помилки користувача, і навіть найбільш технічно підковані компанії вразливі до цих помилок.

Оскільки комп'ютери та інші цифрові пристрої стали важливими для бізнесу та торгівлі, вони також дедалі частіше стають об'єктами атак. Для того, щоб компанія чи окрема особа могли впевнено використовувати комп'ютерний пристрій, вони спочатку повинні бути впевнені, що пристрій жодним чином не скомпрометовано та що всі комунікації будуть безпечними.[1]

Кіберфізичні системи систем (SoSs) – це великомасштабні системи, створені з незалежних і автономних кіберфізичних складових систем (КС), які можуть взаємодіяти для досягнення цілей високого рівня також за втручання людей. Забезпечення безпеки в таких SoSs означає, серед інших функцій, прогнозування та передбачення розвитку функціональних можливостей SoSs, зрештою виявлення можливих шкідливих явищ, які можуть виникнути в результаті взаємодії КС та людей. Такі явища, які зазвичай називають емерджентними явищами, часто є складними і їх важко зафіксувати: перша поява емерджентного явища в кіберфізичному SoSs часто є несподіванкою для спостерігачів. Адекватна підтримка для розуміння явища, що виникає, допоможе зменшити як ймовірність проектних або експлуатаційних недоліків, так і час, необхідний для аналізу відносин між КС, що завжди має ключове економічне значення. Проте не варто вважати, що використання IDS та автоматизація аналізу log-файлів дозволить виявити всі загрози безпеки. Кожен засіб захисту адресовано конкретній загрозі безпеки в системі. Більше того, кожен засіб захисту має слабкі та сильні сторони. Тільки правильно підібравши та налаштувавши ці засоби, можна захиститися від максимально великого спектру атак. [2]

### **Література**

1. Security of Cyber-Physical Systems. URL: <https://www.powermag.com/security-of-cyber-physical-systems/>.
2. What is a Zero-Day Exploit? URL: <https://www.intel.com/content/www/us/en/business/enterprise-computers/resources/what-is-a-zero-day-exploit.html>.

УДК 004.9

**А. Хом'як**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **СИГНАЛИ ГОЛОВНОГО МОЗКУ, ЯКІ МОЖНА ОТРИМАТИ НЕІНВАЗИВНИМИ МЕТОДАМИ**

UDC 004.9

**A. Khomiak**

### **BRAIN SIGNALS OBTAINABLE VIA NON-INVASIVE IMAGING**

Для якісної медичної діагностики стану головного мозку часто виникає потреба в аналізі його макро- та мікроскопічної активності. Неінвазивні методи дозволяють робити це з мінімальними ризиками для здоров'я пацієнта та не потребують складної і довготривалої підготовки до проведення діагностичних процедур.

Електроенцефалограма (ЕЕГ) дозволяє отримати електрограму зі скальпу, що корелює з макроактивністю кори головного мозку [1]. Часове розширення такого сигналу знаходиться в межах мілісекунд, значно перевершуючи частоту інших методик. ЕЕГ використовується зокрема для діагнозу епілептичних станів та проблем зі сном.

Метод викликаних потенціалів надає змогу аналізувати реакцію ЦНС на зовнішні стимули (візуальні [2], слухові тощо) та виявляти порушення діяльності нейронних шляхів внаслідок травми чи захворювання. Отримані сигнали в десятки разів менші за амплітудою ніж ЕЕГ та інші методи електродіагностики, тому необхідно використовувати додаткові засоби їхньої обробки (перетворення Фур'є, усереднення тощо).

Іонні струми в головному мозку збуджують магнітні поля довкола нього, що дозволяє використовувати чутливі магнетометри для отримання інформації. Магнітоенцефалографія вимірює ці зміни та покладається на надпровідні квантові інтерферометри, однак вимагає спеціального середовища, яке екрановане від зовнішніх магнітних полів (включно з магнітним полем Землі) [3].

Магнітно-резонансна томографія покладається на явище ядерного магнітного резонансу для отримання зображень внутрішньої будови м'яких тканин та органів черепа зокрема. Такі дані можуть бути використані для побудови конектому [4] – «мережевої мапи» нейронних зв'язків головного мозку.

У доповіді проведено порівняльний аналіз таких сигналів у контексті їх застосування для медичної діагностики.

#### **Література**

1. Michel, C. M., & Brunet, D. (2019). EEG Source Imaging: A Practical Review of the Analysis Steps. *Frontiers in Neurology*. URL: <https://doi.org/10.3389/fneur.2019.00325>.
3. Kothari, R., Bokariya, P., Singh, S., & Singh, R. (2016). A Comprehensive Review on Methodologies Employed for Visual Evoked Potentials. *Scientifica*, 2016, 9852194. URL: <https://doi.org/10.1155/2016/9852194>.
4. Singh S. P. (2014). Magnetoencephalography: Basic principles. *Annals of Indian Academy of Neurology*, 17 (Suppl 1), P. 107–S112. URL: <https://doi.org/10.4103/0972-2327.128676>.
5. Toga, A. W., Clark, K. A., Thompson, P. M., Shattuck, D. W., & Van Horn, J. D. (2012). Mapping the human connectome. *Neurosurgery*, 71 (1), 1–5. URL: <https://doi.org/10.1227/NEU.0b013e318258e9ff>.

УДК 004.56

**Г. Шимчук, О. Голотенко, Р. Золотий,**

(Тернопільський державний технічний університет імені Івана Пулюя, Україна)

## **ОСНОВНІ ПРОБЛЕМИ ТА ЗАГРОЗИ ХМАРНОЇ БЕЗПЕКИ**

УДК 004.56

**G. Shymchuk, O. Holotenko, R. Zoloty**

### **USE THE MAIN PROBLEMS AND THREATS OF CLOUD SECURITY**

Майже кожна організація різною мірою використовує хмарні обчислення у своєму бізнесі. Однак із запровадженням хмари виникає необхідність переконатися, що стратегія хмарної безпеки організації здатна захистити від найпоширеніших загроз безпеці хмари.

Неправильна конфігурація параметрів безпеки хмари є основною причиною витоку хмарних даних. Стратегії керування хмарною безпекою багатьох організацій недостатні для захисту їхньої хмарної інфраструктури.

Цьому сприяє кілька факторів. Хмарна інфраструктура розроблена таким чином, щоб її можна було легко використовувати та надавати можливість легкого обміну даними, що ускладнює організаціям забезпечення доступу до даних лише авторизованим сторонам. Крім того, організації, які використовують хмарну інфраструктуру, також не мають повної видимості та контролю над своєю інфраструктурою, а це означає, що вони повинні покладатися на елементи керування безпекою, які надає їхній постачальник хмарних послуг (CSP), щоб налаштувати та захистити свої хмарні розгортання. Оскільки багато організацій не знайомі з захистом хмарної інфраструктури та часто мають багатохмарні розгортання – кожне з різним набором засобів безпеки, наданих постачальником, неправильна конфігурація або недогляд у безпеці можуть легко залишити хмарні ресурси організації відкритими для зловмисників.

На відміну від локальної інфраструктури організації, їх хмарні розгортання знаходяться поза периметром мережі та доступні безпосередньо з загальнодоступного Інтернету. Хоча це є активом для доступності цієї інфраструктури для співробітників і клієнтів, це також полегшує зловмиснику отримання неавторизованого доступу до хмарних ресурсів організації. Неправильно налаштований захист або скомпрометовані облікові дані можуть дозволити зловмиснику отримати прямий доступ, можливо, без відома організації.

CSP часто надають своїм клієнтам низку інтерфейсів прикладного програмування (API) та інтерфейсів. Загалом, ці інтерфейси добре задокументовані, щоб зробити їх зручними для використання клієнтами CSP.

Однак це створює потенційні проблеми, якщо клієнт належним чином не захистив інтерфейси своєї хмарної інфраструктури. Документація, розроблена для замовника, також може бути використана кіберзлочинцем для виявлення та використання потенційних методів доступу та викрадення конфіденційних даних із хмарного середовища організації.

Багато людей мають надзвичайно слабкий захист паролів, включаючи повторне використання паролів і використання слабких паролів. Ця проблема посилює вплив фішингових атак і витоку даних, оскільки дає змогу використовувати один викрадений пароль для кількох різних облікових записів.

Викрадення облікових записів є однією з найсерйозніших проблем безпеки в хмарі, оскільки організації все більше покладаються на хмарну інфраструктуру та програми для основних бізнес-функцій. Зловмисник, маючи облікові дані співробітника, може отримати доступ до конфіденційних даних або функцій, а скомпрометовані облікові дані клієнта дають повний контроль над їхнім обліковим записом в Інтернеті. Крім того, у хмарі організаціям часто не вистачає можливості ідентифікувати ці загрози та реагувати на них так само ефективно, як у локальній інфраструктурі.

Хмарні ресурси організації розташовані за межами корпоративної мережі та працюють на інфраструктурі, якою компанія не володіє. Як наслідок, багато традиційних інструментів для

досягнення видимості мережі неефективні для хмарних середовищ, а деяким організаціям бракує інструментів безпеки, орієнтованих на хмару. Це може обмежити можливості організації контролювати свої хмарні ресурси та захищати їх від атак.

Хмара створена для полегшення обміну даними. Багато хмар надають можливість явно запросити співавтора електронною поштою або надіслати посилання, яке дає змогу будь-кому, хто має URL-адресу, отримати доступ до спільного ресурсу.

Хоча цей простий обмін даними є перевагою, він також може бути серйозною проблемою безпеки хмари. Використання спільного доступу на основі посилань – популярного варіанту, оскільки це простіше, ніж явно запросити кожного співавтора – ускладнює контроль доступу до спільного ресурсу. Спільне посилання може бути перенаправлено комусь іншому, викрадене під час кібератаки або здогадане кіберзлочинцем, забезпечуючи несанкціонований доступ до спільного ресурсу. Крім того, обмін на основі посилань унеможлиблює скасування доступу лише до одного одержувача спільного посилання.

Внутрішні загрози є серйозною проблемою безпеки для будь-якої організації. Зловмисник уже має авторизований доступ до мережі організації та деяких конфіденційних ресурсів, які вона містить. Спроби отримати такий рівень доступу – це те, що відкриває більшість зловмисників до їхньої цілі, що ускладнює для невідготовленої організації виявлення зловмисного інсайдера.

У хмарі виявити зловмисника ще складніше. Завдяки хмарному розгортанню компаніям не вистачає контролю над базовою інфраструктурою, що робить багато традиційних рішень безпеки менш ефективними. Це, а також той факт, що хмарна інфраструктура доступна безпосередньо з загальнодоступного Інтернету та часто страждає від неправильних конфігурацій безпеки, ще більше ускладнює виявлення зловмисників.

Кіберзлочинність – це бізнес, і кіберзлочинці обирають свої цілі на основі очікуваної прибутковості своїх атак. Хмарна інфраструктура доступна безпосередньо з загальнодоступного Інтернету, часто неналежним чином захищена та містить велику кількість конфіденційних і цінних даних. Крім того, хмара використовується багатьма різними компаніями, а це означає, що успішна атака може бути повторена багато разів з високою ймовірністю успіху. Як наслідок, хмарні розгортання організацій є звичайним об'єктом кібератак.

Хмара необхідна для ведення бізнесу багатьма організаціями. Вони використовують хмару для зберігання важливих бізнес-даних і запуску важливих внутрішніх і клієнтських програм.

Це означає, що успішна атака типу «відмова в обслуговуванні» (DoS) проти хмарної інфраструктури, швидше за все, матиме серйозний вплив на низку різних компаній. У результаті DoS-атаки, коли зловмисник вимагає викуп, щоб зупинити атаку, становлять значну загрозу для хмарних ресурсів організації.

## **Література**

1. Khalil I, Khreishah A, Azeem M (2014) Cloud computing security: a survey. *Computers* 3(1):1–35
2. Singh S, Jeong Y-S, Park JH (2016) A survey on cloud computing security: issues, threats, and solutions. *J Netw Comput Appl* 75:200–222
3. Khalil IM, Khreishah A, Azeem M (2014) Cloud computing security: a survey. *Computers* 3(1):1–35
4. Ahmed M, Litchfield AT (2018) Taxonomy for identification of security issues in cloud computing environments. *J Comput Inf Syst* 58(1):79–88
6. Sumitra B, Pethuru C, Misbahuddin M (2014) A survey of cloud authentication attacks and solution approaches. *Int J Innov Res Comput Commun Eng* 2(10):6245–6253
7. Subashini S, Kavitha V (2011) A survey on security issues in service delivery models of cloud computing. *J Netw Comput Appl* 34(1):1–11

УДК 004.5

**А. Мачужак**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АВТОМАТИЗАЦІЯ ЗАДАЧ ТЕСТУВАННЯ ТА РОЗГОРТАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

UDC 004.5

**A. Machuzhak**

### **AUTOMATION OF SOFTWARE TESTING AND DEPLOYMENT TASKS**

В процесі розробки програмного забезпечення (ПЗ) важливими є етапи контролю якості (тестування) та розгортання готових версій ПЗ (реліз) в робочому середовищі. При цьому необхідно вирішити задачі пов'язані з такими проблемами [1]:

- Уніфікована розробка – над проектом можуть працювати декілька людей, тому потрібно забезпечити однакові інструменти та середовище розробки, щоб виключити можливість непередбачуваних помилок;
- Тестування – забезпечувати високу якість ПЗ при швидкому темпі виходу нових версій та з можливістю використання тестового середовища без впливу на робоче середовище;
- Швидке розгортання нових змін – бізнес потребує нові можливості та функції для свого продукту, щоб адаптуватися до змін на ринку та бути конкурентоспроможним.

На сьогоднішній день такі задачі вирішуються за допомогою методологій та практик, які об'єднані спільною назвою DevOps [2] (англ. development operations). До основних інструментів на сьогоднішній день належать:

1. Docker, дозволяє інкапсулювати програми в так звані контейнери, які виконуватиметься однаково в різних середовищах;
2. GitHub actions або аналоги для постійної інтеграції (англ. Continuous integration – CI) та постійної доставки (англ. Continuous delivery – CD) – CI/CD такі як GitLab runners та Jenkins. Робота цих технологій полягає в автоматизації тестування та інтегрування нових змін в різні середовища виконання – тестове чи робоче;
3. New relic – забезпечує моніторинг показників продуктивності та працездатності веб-сайту, а також логування дій та ведення історії виконуваних операцій.

В даній роботі пропонується проект автоматизації процесів тестування та розгортання програмного забезпечення, який полягає у контейнеризації готового додатка в Docker, налаштування етапів CI/CD засобами github actions, в ці етапи включаються автоматичне тестування нових версій ПЗ та їхнє розгортання на сервері, як тестовому так і робочому. Останнім кроком буде інтеграція сервісу для моніторингу стану сервера.

До перспектив подальших досліджень належать теми безпеки в контейнеризованих додатках та налаштування інфраструктури для веб-сайту, методології розробки CI/CD, шаблони в хмарних технологіях, а також знання про бази даних (БД) та резервне копіювання БД, щоб захистити дані та мати змогу їх відновити.

#### **Література**

1. Паляниця В. А. Якість програмного забезпечення та тестування: базовий курс. Навчальний посібник. Тернопіль: ФОП Паляниця В. А., 2020. 478с.
2. Learn to become a DevOps Engineer or SRE. URL: <https://roadmap.sh/devops> (дата звернення: 28.11.2022).

### СЕКЦІЯ 3. КОМП'ЮТЕРНІ СИСТЕМИ ТА МЕРЕЖІ

УДК 004.932

**М. Домарецький**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

#### ОГЛЯД СИСТЕМ ДЛЯ РОЗПІЗНАВАННЯ ЖЕСТІВ

UDC 004.932

**М. Domaretskyi**

#### REVIEW OF GESTURE RECOGNITION SYSTEMS

В даний час існують різні пристрої захоплення та розпізнавання жестів (РЖ). Умовно їх можна розділити на дві групи: пристрої на основі оптичних камер; пристрої на основі портативних давачів (наприклад, акселерометри або тензорезистори.)

Перша група пристроїв - оптичні камери, які записують набір зображень для захоплення рухів жестів на відстані. Методи РЖ, засновані на цих пристроях, розпізнають жести, аналізуючи візуальну інформацію, витягнуту із захоплених зображень. До подібних пристроїв належать такі системи, як мультисенсорна система РЖ рук водія, Deep Sign та Kinect.

До другої групи належать, наприклад, такі пристрої, як проект для рольових ігор рукавичка «Відьмака» і американська рукавичка «Language of Glove».

В таблиці наведені основні технічні характеристики розглянутих систем РЖ.

Аналог	Мобільність пристрою	Серійний випуск	Переносимість ПЗ	Модифікування ПЗ	Служба підтримки
Kinect	-	+	-	+	+
Мультисенсорна система для розпізнавання жестів рук водія	-	-	+	+	-
Deep Sign	-	-	-	+	-
The Language of Glove	+	-	+	+	-
Рукавичка «Відьмака»	+	-	-	-	-

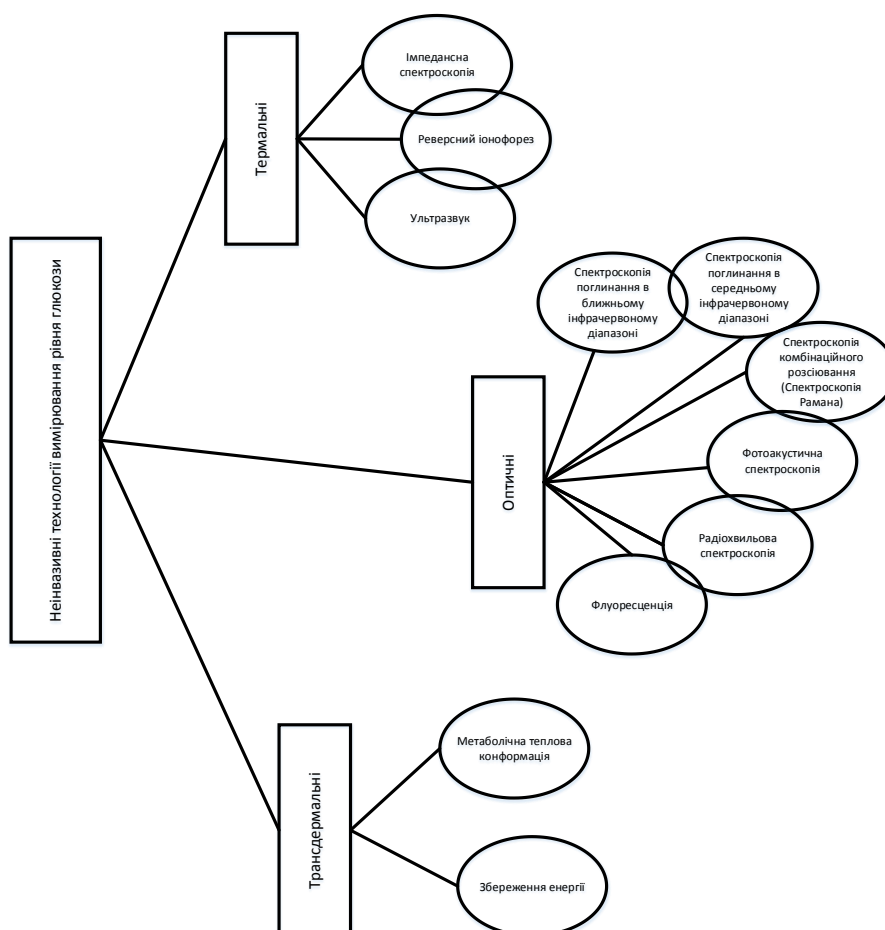
З таблиці видно, що пристрої на основі оптичних камер не мають такої власивості, як мобільність, на відміну від пристроїв на основі портативних давачів. Переносимістю ПЗ володіють лише мультисенсорна система для РЖ рук водія та «Language Of Glove», оскільки тільки ці системи представляють свої бібліотеки у відкритий доступ, і отже, їх ПЗ може бути використане на іншій апаратній платформі. Можливість модифікації ПЗ мають усі перелічені аналоги, крім «Рукавички Відьмака», оскільки цей проект не має якісної документації у відкритому доступі. Серійний випуск і служба підтримки присутні тільки в Kinect – на офіційному сайті постійно з'являються пакети з оновленнями, а також різні засоби і розширення системи.

Розглянуті аналоги мають як переваги, так і недоліки, пов'язані з технічною реалізацією. Перевага пристроїв першої групи полягає у відсутності великої кількості вимірювальних давачів. Це допомагає спростити структуру пристрою. Основними недоліками таких пристроїв є обмежене поле зору камер, високі обчислювальні витрати і необхідність у використанні декількох камер для отримання надійних результатів (через проблеми глибини і оклюзії). Проект «Language of Glove» містить в собі тензорезистори з високою точністю вимірювання, однак, дані давачі є достатньо дорогартісними. Пристрій «Рукавичка Відьмака» навпаки, за рахунок використання акселерометрів має досить низьку вартість, але при цьому не здатен точно формувати сигнали про позицію давача.

## ТЕХНОЛОГІЇ НЕІНВАЗИВНОГО ВИМІРЮВАННЯ РІВНЯ ГЛЮКОЗИ В КРОВІ

## TECHNOLOGIES OF NON-INVASIVE GLUCOSE LEVEL MEASUREMENT IN BLOOD

Суть функціонування неінвазивних вимірювачів глюкози полягає в аналізі за станом кровоносних судин, тобто передбачаються не прямі вимірювання. Окрім цього, існує декілька функціональних моделей, які забезпечують обчислення концентрації глюкози у крові на основі аналізу стану шкірного покриву. Для цього достатньо забезпечити контакт пристрою з ділянкою частини тіла людини. Загалом за технологіями неінвазивного визначення рівня цукру в крові використовуються технології, які показано на рис. 1.



**Рисунок 1.** Технології неінвазивного визначення рівня глюкози в крові

Оптичні методи використовують різні властивості світла для взаємодії з глюкозою в залежності від її концентрації. Трансдермальний метод передбачає вимірювання рівня глюкози через шкіру за допомогою електричних імпульсів або ультразвуку.

## АЛГОРИТМИ МАШИННОГО НАВЧАННЯ ДЛЯ ПРОГНОЗУВАННЯ РІВНЯ ГЛЮКОЗИ В КРОВІ

## MACHINE LEARNING ALGORITHMS FOR PREDICTING THE LEVEL OF GLUCOSE IN THE BLOOD

Для вибору та проведення експериментів щодо прогнозування рівня глюкози в крові для трьох категорій людей: ті, у яких нормальний рівень цукру в крові, ті, хто хворіє діабетом першого типу, ті, хто хворіє діабетом другого типу, потрібно застосувати методи машинного навчання, оскільки кількість даних є надзвичайно велика, а визначити і врахувати кореляції між факторами, або їх групами вручну практично неможливо.

Потрібно відмітити, що при прогнозуванні рівня цукру в крові необхідно розв'язати задачу класифікації та визначити сукупність і пріоритет факторів, які найбільше впливають на розвитку цукрового діабету. У даному випадку, прогнозування трактується як prediction, а не прогнозування розвитку цукрового діабету в часі – forecast. Як приклад задач, методів і алгоритмів, для яких можна застосовувати інструменти машинного навчання належать такі, які проілюстровані на рис. 1.

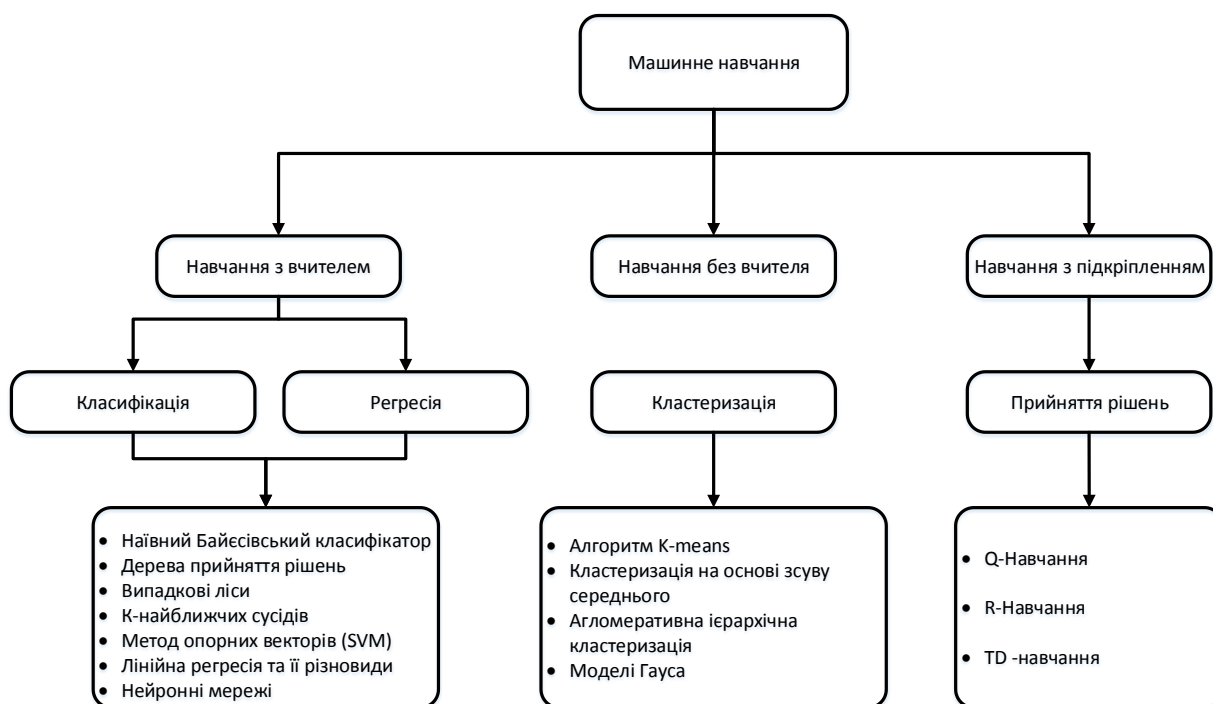


Рисунок 1. Застосування методів і алгоритмів машинного навчання

З практичної точки зору, для розв'язку задачі такого класу найбільш ефективними є алгоритми: логістичної регресії; дерев прийняття рішення; XGBoost; «випадкового лісу». При проведенні експериментів найкращі результати показав алгоритм XGBoost, що забезпечив точність на рівні більше 90%.



## **ОСОБЛИВОСТІ ЗАДАЧ І ФУНКЦІЙ DEVOPS ФАХІВЦІВ**

## **FEATURES OF TASKS AND FUNCTIONS OF DEVOPS SPECIALISTS**

Головна задачам DevOps інженера – максимально збільшити передбачуваність, ефективність та безпеку розробки ПЗ.

Якщо розглядати повний життєвий цикл програмного забезпечення, то на етапі оцінки DevOps фахівці отримують первинну інформацію про необхідність нового кодування та внесення змін до IT-інфраструктури. На етапі проектування визначають вимоги до інфраструктури.

На етапі розробки та тестування займаються розгортанням продукту, а також підтримкою засобів для розробки, інтеграційним та навантажувальним тестуванням ПЗ для перевірки готовності операційного середовища.

Основна частина роботи DevOps інженера посідає етап випуску релізу — поставки продукту замовнику. У центрі уваги перебуває продуктивність всіх потоків процесу доставки. Такий фахівець слідкує за тим, щоб відомі баги ніколи не передавалися на наступний етап робіт, ніколи не розвивалася локальна оптимізація, що веде до створення глобальної деградації.



**Рисунок 1.** Місце DevOps в процесі розробки комп'ютерних систем

До обов'язків DevOps engineer входить:

- розгортання поставленого розробниками релізу у виробництві;
- інтеграція та поглиблення процесів розробки у постачання;
- стандартизація оточення розробки;
- налаштування інфраструктури на особливості пз, що розробляється;
- підготовка продуктивного середовища до частих внесення змін;
- виявлення та виправлення проблем;
- автоматизація процесів.

УДК 004.9

**А. Луцків, І. Барна**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

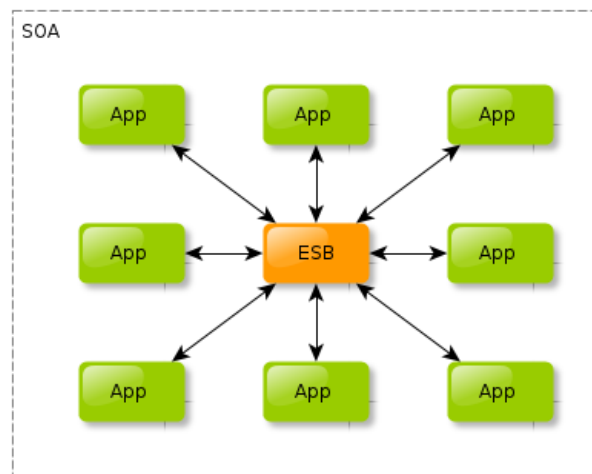
## **АНАЛІЗ СЕРВІСНО-ОРІЄНТОВАНОЇ АРХІТЕКТУРИ В ПРОЦЕСІ ЗАСТОСУВАННЯ DEVOPS ПРАКТИК**

UDC 004.9

**A. Lutskiv, I. Barna**

## **ANALYSIS OF SERVICE-ORIENTED ARCHITECTURE IN THE PROCESS OF APPLICATION OF DEVOPS PRACTICES**

Сервісно-орієнтована архітектура деякий час також відома як централізовано орієнтована архітектура, це архітектура, де кілька послуг, які також називаються користувацькими агентами, використовують або створюють взаємодію з централізованою системою, щоб зробити традиційну монолітну архітектуру менш важкою і слабо зв'язаною. Архітектура, орієнтована на обслуговування, спеціально розроблена для обміну даними між системами. Кожен сервіс забезпечує функціональність на рівні абстракції, розглядається як чорний ящик, автономний, щоб уникнути зайвих витрат на кожен нову розробку (рис. 1).



**Рисунок 1.** Сервісно-орієнтована архітектура

Сервіс абстрактний і може бути розроблений за допомогою будь-якої технології. Зв'язок між ними може здійснюватися за допомогою центральної точки підключення, яка називається Enterprise Service Bus. ESB піклується про всі сервіси та допомагає їм взаємодіяти один з одним.

SOA має свої плюси і мінуси, оскільки SOA надає можливість повторного використання всіх цих незалежних служб, наприклад, якщо потрібно створити модуль аутентифікації входу, замість того, щоб робити його з самого початку, можна використовувати службу аутентифікації facebook/google в SOA. Певним чином гетерогенні послуги SOA роблять систему відмовостійкістю. Якщо одна служба не працює, то система не перестане працювати. Найбільшою перевагою SOA є можливість повторного використання сервісу, якщо створено сервіс, то його завжди можна використовувати знову у нових розробках.

УДК 004.2

**А. Луцків, М. Бондаренко**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ОСОБЛИВОСТІ ОПТИМІЗАЦІЇ СИСТЕМ ПІДТРИМКИ КОРИСТУВАЧІВ ІЗ ЗАСТОСУВАННЯМ ПІДХОДУ СИСТЕМ МАСОВОГО ОБСЛУГОВУВАННЯ**

UDC 004.2

**A. Lutskiv, M. Bondarenko**

## **FEATURES OF USER SUPPORT SYSTEMS OPTIMIZATION USING THE APPROACH OF MASS SERVICE SYSTEMS**

Найбільш ефективним підходом при дослідженні та оптимізації опрацювання потоку звернень у системах підтримки користувачів є застосування алгоритмів і методів теорії систем масового обслуговування (СМО). Теорія СМО зародилась і є розвитком теорії телетрафіка, яка призначена для розв'язування задач, пов'язаних з пропускнуою здатністю систем телекомунікації.

За допомогою методів теорії телетрафіка створено і науково розвинуто підходи щодо оцінювання ефективності систем підтримки користувачів та якості їхнього обслуговування. Принципи даної теорії забезпечують можливість кількісного оцінювання сукупності критеріїв і параметрів систем телекомунікації. У роботі пропонується застосувати такі підходи і до процесу управління зверненнями користувачів у відповідних системах чи підсистемах їх підтримки. Важливою характеристикою та критерієм при застосуванні теорії телетрафіка є врахування стохастичності природи потоку звернень користувачів при їх підтримці. Прогнозування і кількісна оцінка пропускнуої спроможності, а також забезпечення ефективності відповідних процесів є визначальними показниками якості при оптимізації систем підтримки користувачів.

У даному випадку, усі математичні розрахунки засновані на встановленні аналітичної залежності між реакцією системи та зовнішніми факторами впливу. Реакція системи може відображати її стан, наприклад, кількісно виражати ступінь опрацювання одночасних поданих звернень, кількість заявок у черзі, час очікування опрацювання звернення. Зовнішні впливи відображають потоки звернень, помилки системи, простій системи через відмову, фактори, що впливають на надійність підтримки користувача тощо.

Зовнішнім фактором впливу у системах підтримки користувачів є різноманітність інформації, що циркулює в межах системи – джерело одержання звернення і формат звернення. Враховуючи той факт, що потоки звернень користувачів можуть суттєво відрізнятися за пріоритетом, схемою надання послуги, принципами застосування протоколів та ін., то найбільш доцільним є використання багатфакторної моделі. Враховуючи те, що необхідно проводити розрахунки і прогнозування звернень у часі, то варто накласти обмеження на зовнішні фактори з урахуванням їх особливостей або з використанням моделей багатопотоковості.

Опис реакції системи на сукупність усіх зовнішніх впливів є надзвичайно складною задачею. Зовнішні фактори характеризуються:

- великою сукупністю зовнішніх впливів;
- кожен вплив не завжди однозначно описується простими формулами, що дозволяють отримати кінцевий результат із зрозумілим фізичним змістом;
- опис зовнішніх впливів не завжди адекватний реальним процесам, які відбуваються у системі.

УДК 004.4

**А. Луцків, М. Бондаренко**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

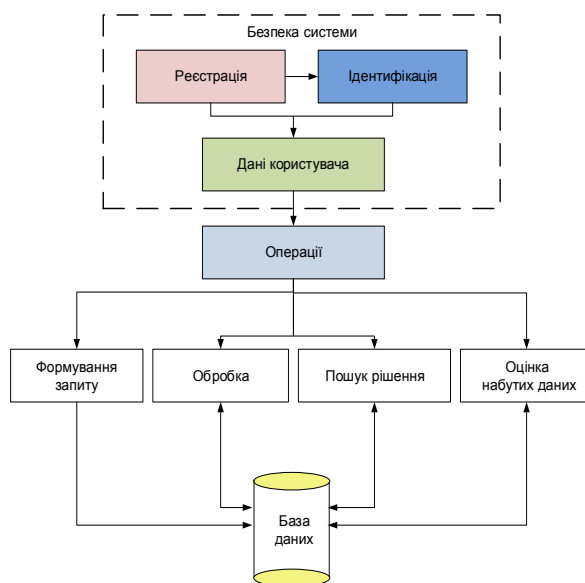
## **АРХІТЕКТУРА СИСТЕМИ ПІДТРИМКИ З ОПТИМІЗАЦІЄЮ ПРОЦЕСУ УПРАВЛІННЯ ЗВЕРНЕННЯМИ КОРИСТУВАЧІВ**

UDC 004.4

**A. Lutskiv, M. Bondarenko**

## **SUPPORT SYSTEM ARCHITECTURE WITH OPTIMIZATION OF THE USER COMPLAINT MANAGEMENT PROCESS**

Важливим етапом розробки системи підтримки користувачів є проектування, тобто відображення концептуальної моделі на реальні компоненти платформи обраної для реалізації даного продукту. Алгоритми функціонування сучасних програмних систем, як правило, є складним і нелінійним, що ставить перед розробниками та інженерами завдання максимально ефективно будувати архітектуру на підставі інтерфейсів взаємодії між її компонентами. Компоненти системи реалізують окремі елементи функціональності програмного продукту. Об'єднати за рахунок надання інтерфейсів взаємодії, потоків передачі даних і використання загальних джерел даних, базуючись на однакових принципах поведінки вони складають єдиний комплекс. Виходячи з побудованої концептуальної моделі системи підтримки користувачів можна побудувати відповідну архітектуру програмного рішення, яка представлена на рис. 1.



**Рисунок 1.** Архітектура програмної реалізації системи управління зверненнями користувачів

Архітектура відображає надані системою сервіси, функції і зв'язки між етапами обробки запитів. Реєстрація користувачів служить для збору ідентифікованих даних, регулювання механізмів доступу до різного функціоналу сайту і збору інформації, яка може виявитися важливою при вирішенні інцидентних ситуацій. Ідентифікація користувачів є механізмом індивідуалізації профілю системи для різних користувацьких ролей і привілеїв доступу, визначає доступність операцій із запитом. Разом з реєстрацією та профілями користувачів ідентифікація складає комплекс безпеки системи.

УДК 004.3

**В. Яцишин, Т. Кобець**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ТЕХНОЛОГІЯ MESH В КОМП'ЮТЕРНИХ СИСТЕМАХ ПЕРЕДАЧІ ДАНИХ**

UDC 004.3

**V. Yatsyshyn, T. Kobets**

## **TECHNOLOGIES OF NON-INVASIVE GLUCOSE LEVEL MEASUREMENT IN BLOOD**

Меш-технологія – це мережева структура, у якій комп'ютери з'єднані один з одним за допомогою різноманітних резервних з'єднань. Існує кілька шляхів від одного комп'ютера до іншого. Він не містить комутатора, концентратора чи будь-якого центрального комп'ютера, який діє як центральна точка зв'язку. Інтернет є прикладом mesh-топології.

Меш-топологія в основному використовується для реалізацій глобальних комп'ютерних мереж (WAN), де збої зв'язку є критичною проблемою. Ця топологія в основному використовується для бездротових мереж.

Mesh-топологію мережі можна описати формулою, поданою нижче, а візуалізація цієї топології показана на рис. 1.6.

$$Line\_number = \frac{n * (n - 1)}{2} \quad (1)$$

де  $n$  – кількість хостів, які представляють мережу.

Меш-топологія поділяється на дві категорії: повністю зв'язана топологія або частково зв'язана (рис. 1.).



**Рисунок 1.** Меш-топологія

У повній mesh-топології кожен комп'ютер підключено до всіх інших комп'ютерів, доступних у мережі. Часткова mesh-топологія передбачає, що деякі хости підключені до тих комп'ютерів, з якими вони часто спілкуються.

УДК 004.023

**В. Яцишин, Т. Кобець**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **МЕТОДИ ВИБОРУ ОПТИМАЛЬНИХ КОМПОНЕНТІВ КОМП'ЮТЕРНИХ СИСТЕМ НА ОСНОВІ АНАЛІЗУ ІЄРАРХІЇ**

UDC 004.023

**V. Yatsyshyn, T. Kobets**

## **METHODS OF SELECTING OPTIMUM COMPUTER SYSTEM COMPONENTS BASED ON HIERARCHY ANALYSIS**

Математично задача щодо прийняття рішень, а також прогнозування пріоритетів на основі методу аналізу ієрархій формулюється наступним чином. Задано множину альтернативних варіантів технічних систем або захищеності проектних рішень  $A = \{a_1, a_2, \dots, a_n\}$ , де необхідно приймати рішення по найкращому варіанту. Необхідно здійснити впорядкування множини  $A$  відповідно до переваг, заданими шляхом парних порівнянь, кількісних оцінок, оцінок відносно стандартів по множині критеріїв якості  $K = \{k_1, k_2, \dots, k_n\}$ .

Множина критеріїв  $K$  представляє собою впорядковану структуру (ієрархію), яка підпорядковується принципу ієрархічної композиції, тобто між елементами цієї множини задані відношення типу «абстрактне-конкретне», які можна виразити за допомогою бінарної матриці зв'язків  $B$ :

$$B = \begin{bmatrix} b_{11} & b_{12} & \dots & b_{1m} \\ b_{21} & b_{22} & \dots & b_{2m} \\ \dots & \dots & \dots & \dots \\ b_{m1} & b_{m2} & \dots & b_{mm} \end{bmatrix} \quad (1)$$

де  $b_{ij} = 1$ , якщо між  $k_i$  і  $k_j$  є зв'язок, і  $b_{ij} = 0$  в іншому випадку.

Рішення досягається шляхом використання лінійної згортки векторів пріоритету існуючих альтернатив на заданій критеріальній структурі. Ці вектори, а також приватні (часткові) критерії одержують шляхом визначення «головного» власного вектора відповідної матриці, причому дані у матрицях парних порівнянь можуть представляти собою функції від часу.

Для того, щоб визначити головний власний вектор потрібно розв'язати матричне рівняння такого вигляду:  $AW = \lambda_{max}W$ .

Якщо ставиться задача прогнозування поведінки системи на певному інтервалі часу, то визначення «головного власного вектора» проводиться шляхом розв'язку матричного рівняння  $A(t)W(t) = \lambda_{max}(t)W(t)$ .

Таким чином, алгоритм розв'язку задачі ранжування і вибору альтернатив, а також прогнозування динаміки пріоритетів на основі МАІ міститиме такі основні кроки: визначення множини альтернатив, множини критеріїв якості; Визначення методів оцінювання; Перевірка узгодженості введених оцінок; Розрахунок власних векторів; Провести ієрархічний синтез і розрахувати глобальний вектор пріоритетів (щодо фокуса ієрархії); Обробка отриманого масиву за допомогою регресійного аналізу.

УДК 004.01

А. Луцків, М. Тимошук

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## ВАЖЛИВІСТЬ ДОКУМЕНТУВАННЯ В ПРОЦЕСІ ВДОСКОНАЛЕННЯ КОМП'ЮТЕРНИХ СИСТЕМ

UDC 004.01

A. Lutskiv, M. Tymoshchuk

## THE IMPORTANCE OF THE DOCUMENTATION IN THE PROCESS OF IMPROVING COMPUTER SYSTEMS

Документування – це процес фіксування інформації, створюваної в рамках життєвого циклу комп'ютерної системи. Фіксація інформації виконується шляхом її документування. Процес генерування документації, зазвичай, відбувається разом з іншими процесами реалізації КС (рис. 1.1).

Процес формування документації включає набір операцій, які мають відношення до планування, проектування, реалізації, випуску, редагування, поширення і супроводу документів необхідних для керівників, інженерів і користувачів. Потрібно відмітити, що тільки програмне забезпечення КС, яке містить документацію, є програмним продуктом. Тому документування – один з ключових процесів ЖЦ, а повна і якісна документація знижує витрати на його розробку і супровід. Навпаки, неповна і застаріла документація ускладнює розробку і супровід комп'ютерної системи, будучи причиною додаткових матеріальних і часових витрат і, врешті-решт стає однією з причин проведення зворотної інженерії.

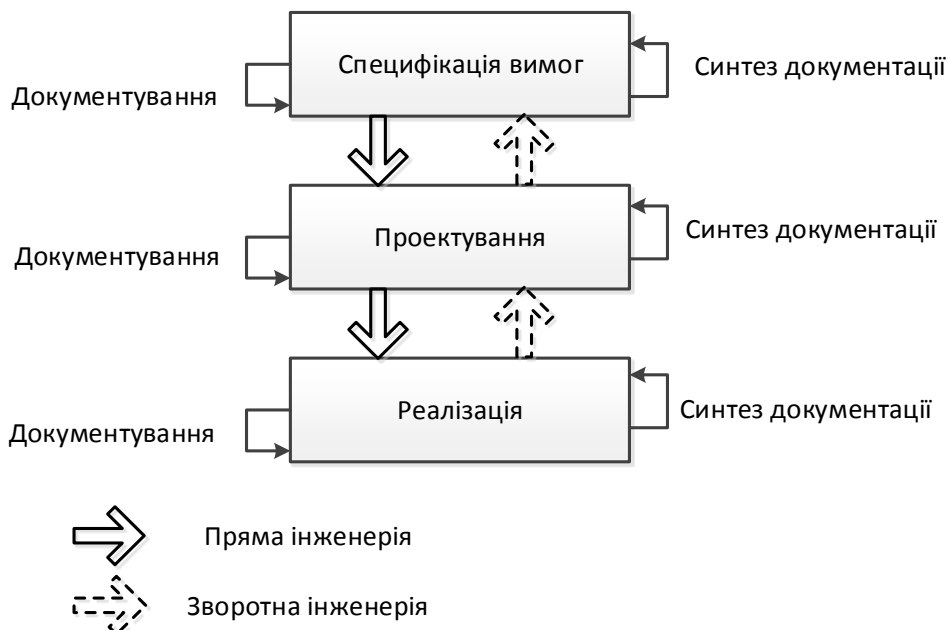


Рисунок 1. Процеси документування в процесі прямої і зворотної інженерії

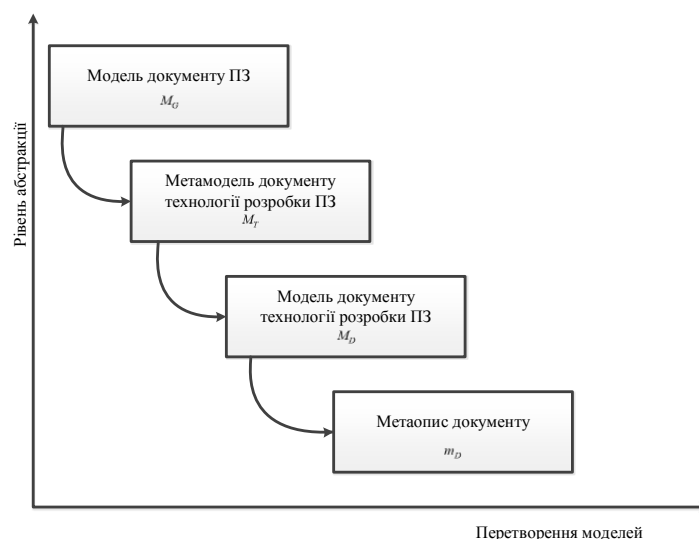
В контексті зворотної інженерії застосовується синтез документації КС (рис. 1.1). Синтез документації в процесі зворотної інженерії практикується при вдосконаленні КС або при реінженерії наслідуваного ПЗ КС, в тому випадку коли документація програмного продукту загублена, неповна, неточна або погано структурована.

## МОДЕЛІ ПРЕДСТАВЛЕННЯ ДОКУМЕНТАЦІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПРИ ВДОСКОНАЛЕННІ КОМП'ЮТЕРНИХ СИСТЕМ

### MODELS OF SOFTWARE DOCUMENTATION VIEW IN THE IMPROVEMENT OF COMPUTER SYSTEMS

Аналіз документації програмного забезпечення з точки зору: структури та загальної організації документації; вимог, які висуваються до технологій реалізації комп'ютерної системи; вимог побудови конкретного документу на різних етапах життєвого циклу, показав, що для представлення документів можна використати наступні моделі (рис. 1):

- структурна модель документу;
- метамодель документу, що описує використовувані технології розробки;
- модель документації технології при розробці КС.



**Рисунок 1.** Моделі документу з точки зору синтезу документації

Модель документу ПЗ  $M_G$  призначена для опису структурних елементів документу ПЗ та враховує особливості автоматизованого створення документації та не залежить від вимог до використовуваних технологій розробки. Метамодель документації  $M_T$  щодо використовуваних технологій реалізації описує структурні компоненти документу ПЗ і враховує вимоги  $R_i$ , що володіють спільними рисами технологій розробки. Модель  $M_T$  є метамоделлю  $M_G$ . Модель  $M_D$ , інтерпретує документ технології розробки, що забезпечує опис структурних елементів документу з врахуванням вимог  $R_i$  для конкретної технології. Ця модель застосовується при створенні описів конкретних документів технології як метаданих, що можуть використовуватись при формуванні екземплярів документів при синтезі документації автоматизованих шляхом.



## ОБРОБКА ЗОБРАЖЕНЬ ІЗ ВИКОРИСТАННЯМ ГЕНЕТИЧНОГО АЛГОРИТМУ

### IMAGE PROCESSING USING GENETIC ALGORITHM

Принцип роботи генетичного алгоритму (ГА) полягає у створенні набору випадкових рішень, що називається населенням. На кожному етапі оптимізації цільова функція обчислюється для всієї популяції, у результаті виходить ранжований перелік рішень. Після ранжування рішень формується нова популяція, котра носить назву наступного покоління. Спочатку до нової популяції включаються найкращі рішення з поточної. Цей процес зветься елітизм [1].

Крім них, до наступної популяції входять абсолютно нові рішення, що отримуються шляхом модифікації найкращих за допомогою генетичних операцій, таких як схрещування та мутація. Конкретні реалізації способів уявлення особин і генетичних операцій породжують різні різновиди еволюційного програмування. Потім цей процес повторюється – нова популяція ранжується та створюється чергове покоління.

Так триває задане число разів або до тих пір, поки протягом кількох поколінь не спостерігається жодних покращень. Принцип методу показаний на рисунку 1.

Для аналізованої задачі генерації системи виявлення об'єкта на зображенні, як генотип був обраний вектор  $S = \{T_1, T_2, c\}$ . Тут  $T_1$  – дерево рішень, відповідне до обробки зображення з камери,  $T_2$  – дерево рішень, відповідне до обробки шаблонного зображення,  $c$  – параметр алгоритму виявлення об'єкта.

Вважається, що в результаті виконання ГА буде отримана система, здатна максимально точно встановлювати положення об'єкта, що цікавить, на зображенні або виявляти факт його відсутності.

Для перевірки якості роботи системи використовується вибірка зображень, елементи якої складаються з:

- зображення, отриманого з камери;
- значень давачів зовнішнього середовища у момент отримання цього зображення;
- координат центральної точки об'єкта, що цікавить, на зображенні або вказівки факту його відсутності.

Система виявлення застосовується до кожного елемента вибірки.

#### Література

1. Goldberg, D.E. Genetic Algorithms in Search, Optimization, and Machine Learning. Reading, Massachusetts: Addison-Wesley, 1989

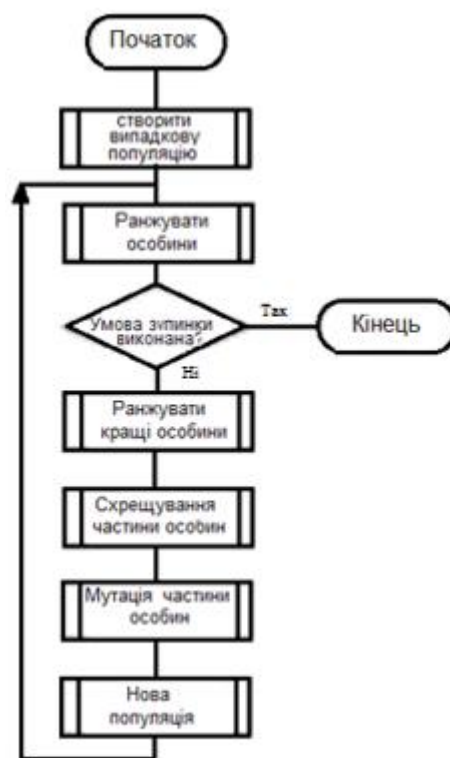


Рисунок 1. Блок-схема ГА

УДК 681.518.3

**Ю. Гук, А. Паламар**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **МЕТОД АДАПТИВНОГО РЕГУЛЮВАННЯ ДОРОЖНЬОГО РУХУ НА ПЕРЕХРЕСТІ НА ОСНОВІ ІНТЕРНЕТУ РЕЧЕЙ**

UDC 681.518.3

**Y. Huk, A. Palamar**

## **METHOD OF ADAPTIVE TRAFFIC CONTROL AT AN INTERSECTION BASED ON INTERNET OF THINGS**

В Україні щороку збільшується кількість транспортних засобів у населення. При цьому темпи розвитку дорожньої інфраструктури в містах залишаються низькими. Це спричиняє суттєве перевантаження центральних міських вулиць автомобільним транспортом, який перебуває в індивідуальному користуванні. Саме тому актуальним завданням є зменшення кількості та тривалості заторів на дорогах міських вулиць.

В результаті огляду наукової літератури по темі дослідження [1] було виявлено, що безпека руху пішохідних та транспортних потоків в містах залежить від багатьох факторів. До них відноситься в першу чергу інтенсивність дорожнього руху, кліматичні та метеорологічні умови, ефективність використання технічних засобів для регулювання руху. Важливе значення для автоматизованого керування пішохідними та транспортними потоками має врахування динаміки змін цих параметрів.

В роботі запропоновано застосувати технологію інтернету речей для підвищення ефективності роботи системи керування транспортними потоками в міському середовищі. Розроблений метод адаптивного регулювання дорожнього руху через перехрестя. Його суть полягає в тому, що завдяки даним, отриманим від давачів здійснюється розрахунок інтенсивності дорожнього трафіку на перехресті і в залежності від того, де ця величина більша, світлофор збільшує тривалість зеленого сигналу світлофора. Це дає змогу зменшити час простою транспортних засобів на перехресті.

Алгоритм керування в онлайн режимі розроблений з врахуванням дорожньої ситуації та параметрів середовища. Робота системи починається з підрахунку кількості транспортних засобів, які рухаються через перехрестя. Також, важливим етапом є вимірювання рівня освітленості доріг та метеорологічних параметрів [2] для того, щоб враховувати їх в процесі керування сигналами світлофорів.

Застосування інтернету речей, завдяки можливості отримувати та аналізувати дані з багатьох перехресть, дозволяє суттєво підвищити ефективність керування дорожнім рухом. Отримана інформація може бути використана для прийняття управлінських рішень з метою підвищення безпеки руху пішоходів та транспорту.

### **Література**

1. Паламар А., Палій Р. Система моніторингу технічного стану транспортних засобів на основі технології інтернету речей: матеріали V науково-технічної конференції «Інформаційні моделі, системи та технології». Тернопіль: ТНТУ. 2018. С. 77.
2. Оконський М.В., Лупенко С.А., Паламар А.М. Інформаційно-вимірювальна система для контролю метеорологічних параметрів на основі Інтернету речей: матеріали ІХ науково-технічної конференції «Інформаційні моделі, системи та технології». Тернопіль: ТНТУ. 2021. С. 118.

УДК 004.3

**О. Гуменюк**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АНАЛІЗ РОБОТИ ІНСТРУМЕНТУ ДЛЯ УПРАВЛІННЯ ТА АНАЛІЗУ ЖУРНАЛІВ GRAYLOG**

UDC 004.3

**O. Humeniuk**

## **PERFORMANCE ANALYSIS OF THE GRAYLOG LOG MANAGEMENT AND ANALYSIS TOOL**

Graylog – це потужна платформа, яка дозволяє легко керувати записами структурованих та неструктурованих даних разом із налагодженням додатків. Він заснований на Elasticsearch, MongoDB і Scala.

Він має головний сервер, який приймає дані від своїх клієнтів, встановлених на різних серверах, та веб-інтерфейс, який відображає дані та дозволяє працювати із записами, доданими основним сервером.

Graylog ефективний під час роботи з необробленими рядками (наприклад, із системним журналом) – інструмент аналізує та переробляє їх у потрібні нам структуровані дані. Він також забезпечує розширений налаштований пошук записів з використанням структурованих запитів. Іншими словами, при правильній інтеграції з веб-програмою, Graylog допомагає інженерам аналізувати поведінку системи майже в кожному рядку коду.

Основною перевагою Graylog є те, що він надає єдиний ідеальний екземпляр збору записів журналів для всієї системи. Це корисно, якщо системна інфраструктура велика та складна. Записи можна було розповсюджувати в кількох місцях, через кілька різних сервісів, і не всі члени команди могли мати негайний доступ до всіх його компонентів, чи ділитися ними. З Graylog ми вирішуємо ці проблеми та гарантуємо швидке реагування на інциденти.

У Logicify його можна використовувати як для додатків, що у розробці, так тих, які вже були опубліковані. В обох випадках деякі режими Graylog унікальні, а інші перетинаються.

Складається Graylog із трьох компонентів:

- graylog-webui – веб-інтерфейс на Rails,
- graylog-server – Java TCP/UDP лог-колектор,
- mongodb для зберігання власне логів та налаштувань всієї системи в цілому.

Graylog-server дозволяє за протоколами TCP/UDP, як і за допомогою звичайного syslog, приймати звідусіль логи, mongodb здійснює зберігання, rails забезпечує візуально-красиве відображення даних та графіків.

Що стосується явних переваг системи, то окрім звичайних для syslog-server функцій хочеться відзначити також такі цікаві моменти в роботі системи:

- Агрегація повідомлень у потоки. За ключовим словом об'єднуємо потік логів з кількох хостів, на один потік «stream», можна створити сповіщення про події і зробити так, щоб ці сповіщення приходили комусь на пошту чи в месенджер Telegram.
- Агрегація хостів у групи. Можна об'єднати потоки з різних хостів до однієї групи.
- Вибірки з усього масиву за допомогою regex, за часом, за важливістю, по об'єктах тощо. Можна знайти все, що завгодно і коли завгодно.
- Blacklists для логів. За допомогою регулярних виразів можна фільтрувати логи. Все, що ми заборонимо, до бази не потрапить.

- Авторотація логів. Не потрібно дбати про очищення старих записів, mongodb сама зробить всю роботу за допомогою механізму capped collections.
- Можливість використання GELF – graylog extended log format, таким чином розширюючи стандартну довжину syslog повідомлення в 1024 байти. За допомогою GELF можна моніторити не тільки системні повідомлення, але й логіку роботи коду, посилаючи розгорнуті повідомлення прямо з програми.

### **Література**

1. Syslog Message Format. 2020. URL: <https://techdocs.audiocodes.com/session-border-controllersbc/mediant-software-sbc/user-manual/version740/content/um/Syslog%20Message%20Format.htm>.
2. What is graylog URL: <https://www.graylog.org/>.
3. HOW TO USE GRAYLOG AS A SYSLOG SERVER. 2018. URL: <https://www.graylog.org/post/how-to-use-graylog-as-a-syslog-server/>.

УДК 004.4

**О. Гуменюк**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АНАЛІЗ РОБОТИ СТАНДАРТУ ЖУРНАЛЮВАННЯ SYSLOG**

UDC 004.4

**О. Humeniuk**

## **ANALYSIS OF THE OPERATION OF THE SYSLOG JOURNALING STANDARD**

Syslog – це стандарт для надсилання та отримання повідомлень у певному форматі від різних мережевих пристроїв. Повідомлення включають тимчасові мітки, повідомлення про події, серйозність, IP-адреси хостів, діагностику та багато іншого. Що стосується вбудованого рівня серйозності, то він може передавати повідомлення в діапазоні від рівня 0 – аварійний, рівня 5 – попередження, нестабільність системи, критичний та рівнів 6 та 7 – інформаційний та налагоджувальний.

Більше того, Syslog є відкритим. Syslog був розроблений для моніторингу мережевих пристроїв і систем з метою надсилання повідомлень при виникненні будь-яких проблем з функціонуванням, він також відправляє попередження про заздалегідь попереджені події та відстежує підозрілу активність через журнал змін/журнал подій мережевих пристроїв, що беруть участь.

Протокол Syslog був спочатку написаний Еріком Оллманом і визначений RFC 3164. Повідомлення надсилаються через IP-мережі на збирачі повідомлень про події або сервери syslog. Syslog використовує для зв'язку протокол User Datagram Protocol (UDP), порт 514. Хоча сервери syslog не надсилають підтвердження отримання повідомлень. З 2009 року syslog стандартизовано IETF в RFC 5424.

Для своєї роботи Syslog використовує наступні компоненти:

- Прослуховувач Syslog – збирає та обробляє дані syslog, надіслані через UDP порт 514. Однак отримання підтвердження не передбачено, і надходження повідомлень не гарантується;

- база даних – сервери syslog потребують бази даних для зберігання величезної кількості даних для швидкого доступу;

- програмне забезпечення для керування та фільтрації - оскільки обсяг даних може бути величезним, пошук певних записів у журналі може зайняти занадто багато часу. Сервер syslog потребує допомоги для автоматизації роботи, а також для фільтрації для перегляду певних повідомлень журналу. Наприклад, він може отримувати повідомлення на основі певних параметрів, таких як критична подія або ім'я пристрою. Ви також можете використовувати фільтр, щоб не бачити певних типів записів за допомогою правила Negative Filter. Якщо ви бажаєте, ви можете показати всі критичні повідомлення журналу від брандмауера.

У стандарті Syslog існує три різні рівні, а саме:

- зміст Syslog (інформація, що міститься в повідомленні про подію);
- додаток Syslog (генерує, інтерпретує, маршрутизує та зберігає повідомлення);
- транспорт Syslog (передає повідомлення).

Сигнали тривоги можуть бути налаштовані на відправлення повідомлень через SMS, спливаючі повідомлення, електронну пошту, HTTP та багато іншого. Оскільки процес автоматизований, IT-команда отримає негайне повідомлення про раптову відмову будь-якого з пристроїв.

Сервери Syslog використовуються для надсилання даних діагностики та моніторингу. Потім ці дані можуть бути проаналізовані для моніторингу системи, обслуговування мережі тощо. Оскільки протокол Syslog підтримується широким спектром пристроїв, вони можуть зручно реєструвати інформацію на сервері Syslog.

Ці дані можна аналізувати визначення поведінки систем. Крім того, журнали вважаються надійним джерелом даних для розуміння поточної статистики системи та прогнозування тенденцій. Не кажучи вже про те, що журнали використовуються для таких дій, як усунення несправностей або відхилення системи після збою.

### **Література**

1. What is Syslog. 2017. URL: <https://www.paessler.com/it-explained/syslog>.
2. Syslog message formats. 2018. URL: <https://support.oneidentity.com/kb/4282913/syslog-message-formats>.
3. SYSLOG Over TCP. 2022. URL: <https://docs.citrix.com/en-us/citrix-adc/current-release/system/audit-logging/reliable-syslog.html>.

УДК 004.415.25

**А. Лупенко, С. Куліков, Д. Денисов**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **КЛАСИФІКАЦІЯ ТА ОСОБЛИВОСТІ ЗАСТОСУВАННЯ ПРИКЛАДНИХ ПРОГРАМНИХ ІНТЕРФЕЙСІВ ПРИ РЕАЛІЗАЦІЇ КОМП'ЮТЕРНИХ СИСТЕМ**

UDC 004.415.25

**A. Lupenko, S. Kulikov, D. Denysov**

## **CLASSIFICATION AND FEATURES OF THE APPLICATION PROGRAMMING INTERFACES IN COMPUTER SYSTEMS IMPLEMENTATION**

API (Application Programming Interface) – набір правил і механізмів, за допомогою яких один додаток або компонент взаємодіє з іншими.

Властивості, якими повинен володіти API:

- простота використання і підтримки – хороший API просто використовувати і підтримувати;
- хороша конверсія в середовищі розробників – стимулює кількість клієнтів і користувачів сервісу;
- популярність сервісу – чим більше користувачів API, тим вища популярність сервісу;
- ізоляція компонентів – чим краща структура API, тим краще ізоляція компонентів;
- зручність використання API – програмний інтерфейс є своєрідним інтерфейсом для розробників на який вони звертають увагу в першу чергу при зустрічі з продуктом. Якщо API не забезпечує попередніх властивостей, то технічний експерт не буде рекомендувати компаніям використовувати такий продукт, купуючи щось стороннє.

Класифікація API за способом реалізації:

- Web service APIs: XML-RPC and JSON-RPC, SOAP, REST, WebSockets APIs, Library-based APIs,
- Java Script;
- Class-based APIs: C# API, Java.

Види API за категоріями застосування:

- OS function and routines – Access to file system; Access to user interface;
- Object remoting APIs – CORBA, .Net Remoting;
- Hardware APIs: Video acceleration (OpenCL...), Hard disk drives; PCI bus та ін.

RPC (remote procedure call – віддалений виклик процедур») – поняття, що поєднує давні, середні і сучасні протоколи, які дозволяють викликати метод в іншому додатку.

XML-RPC – протокол, що з'явився в 1998 р. незабаром після появи XML. Спочатку він підтримувався Microsoft до переходу на SOAP, тому .Net Framework не містить класів для підтримки цього протоколу. Незважаючи на це, XML-RPC продовжує існувати в різних мовах (особливо в PHP).

SOAP з'явився в 1998 р. стараннями Microsoft. Він був анонсований як революція в світі. Не можна сказати, що все пішло за планом Microsoft: була величезна кількість критики через складність протоколу. У той же час, були і ті, хто вважав SOAP справжнім проривом. Протокол продовжував розвиватися десятками нових і нових специфікацій, поки в 2003 р. W3C не затвердила в якості рекомендації SOAP 1.2.

## ПРИЗНАЧЕННЯ І ДОЦІЛЬНІСТЬ ВИКОРИСТАННЯ API GATEWAY У КОМП'ЮТЕРНИХ СИСТЕМАХ

## PURPOSE AND FEASIBILITY OF USING API GATEWAY IN COMPUTER SYSTEMS

Взаємодія сервісів між собою можлива лише через мережу. Розробляючи програму за таким підходом, варто виходити з того, що компоненти можуть не працювати і тому цей момент потрібно враховувати при розробці нових компонентів. Також, варто передбачити системи автоматичного розгортання та моніторингу сервісів, оскільки в системі їх може бути дуже багато і фізично прослідкувати за роботою кожного є, часто, неможливим.

Для виконання взаємодії компонентів між собою використовується API Gateway (рис. 1).

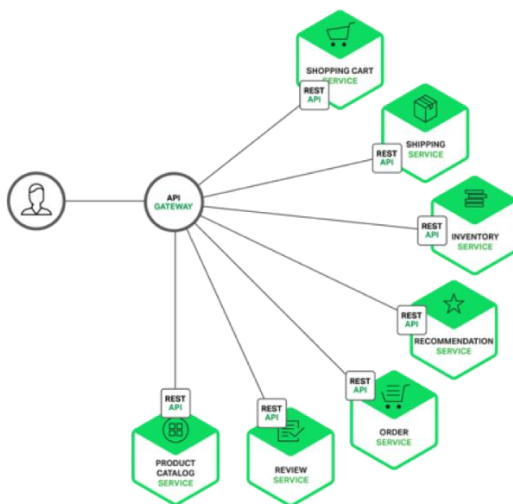


Рисунок 1. Використання API Gateway

Gateway прикладних програмних інтерфейсів доцільно використовувати у випадку проектування мікросервісної архітектури. Функції, які він буде виконувати, полягають у тому, що за наявності в backend декількох сервісів, більш ефективно застосовувати для доступу до них елементарний компонент. Задача API Gateway – забезпечувати збір бізнес-викликів до відповідних сервісів, що дозволить підвищити ефективність процесу маршрутизації повідомлень. Ще однією функцією API Gateway є забезпечення представлення даних для користувача у тому вигляді, в якому він цього потребує і який є наперед визначеним. Для прикладу, за наявності двох версій додатку (для веб і мобільного використання) доцільно використовувати 2 API Gateway, що забезпечать збір даних від однакових сервісів, але формувати їх вигляд будуть по-різному. Вимогою при імплементації API Gateway є задоволення простоти функціональності та мінімальної бізнес-логіки для уникнення надмірного дубляжу та спрощення процесу підтримки. Простота зводиться до того, що API Gateway не виконує жодних функцій окрім передачі даних.



## ПРОЦЕС ФОРМУВАННЯ ПРОГРАМНИХ КОМПОНЕНТІВ ПОВТОРНОГО ВИКОРИСТАННЯ ПРИ РЕАЛІЗАЦІЇ КОМП'ЮТЕРНИХ СИСТЕМ

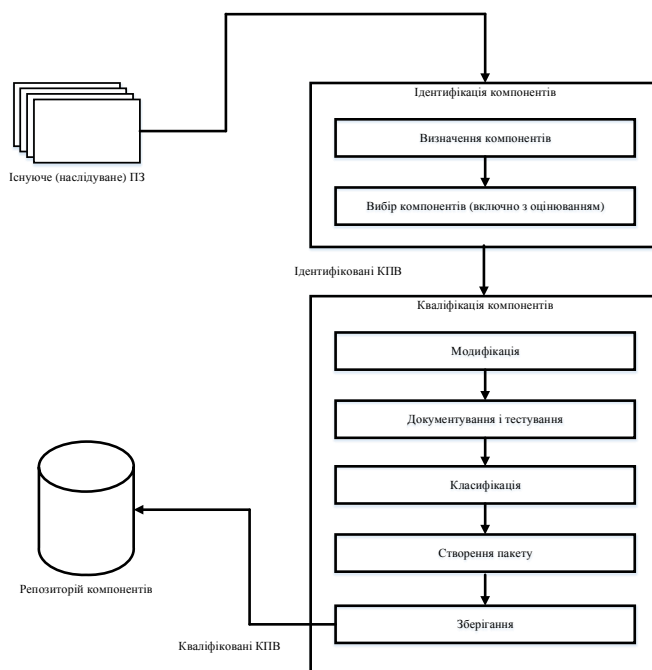
UDC 004.4

V. Yatsyshyn, N. Shabliiy, I. Dyshkant

## THE PROCESS OF FORMING REUSABLE SOFTWARE COMPONENTS IN THE IMPLEMENTATION OF COMPUTER SYSTEMS

Процес створення компонентів повторного використання при реалізації комп'ютерних систем поділяють на операції, які формують два етапи: визначення і затвердження (рис. 1.).

До стадії визначення або, по-іншому ідентифікації, входять операції з виявлення та вибору кандидатів у компоненти повторного використання. Процес виконання операції визначення може виконуватись в ручному режимі або за допомогою засобів автоматизації фахівцем з програмування. Визначені програмістом компоненти можна помітити маркером потенційного компонента до повторного використання і до них застосовується наступна операція – вибору. Оскільки, при аналізі можливе виявлення кількох потенційних компонентів, то для вибору кращих необхідно провести їх оцінювання щодо придатності використання як КПВ. Обрані програмні компоненти переміщують на стадію затвердження (рис. 1).



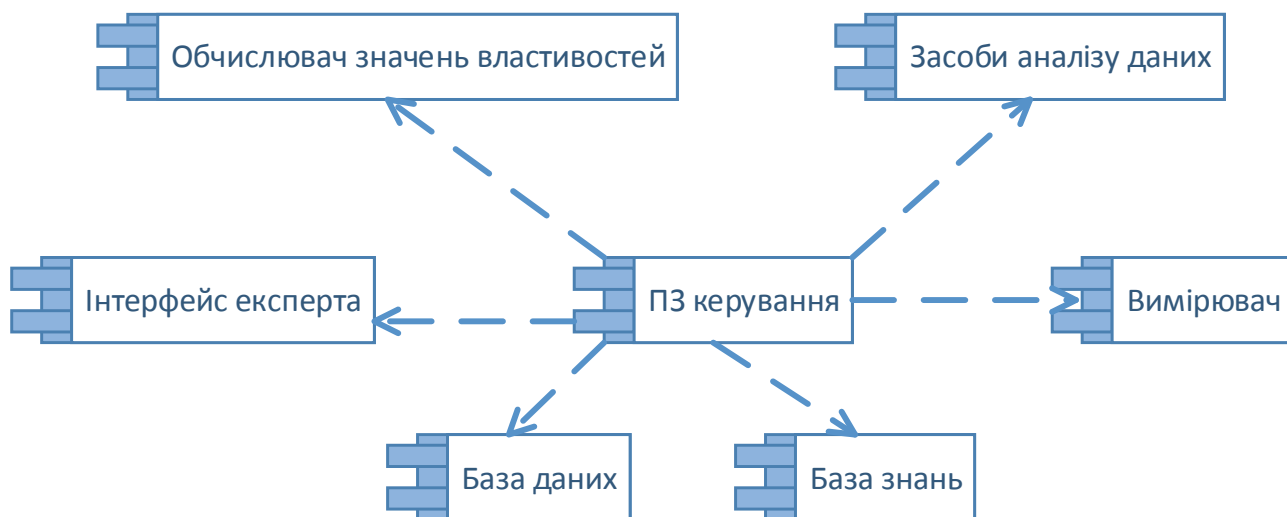
**Рисунок 1.** Процедура формування компонентів повторного використання

Вибір оптимального КПВ дає змогу знизити вартість реалізації КПВ, оскільки забезпечує виключення «гірших» альтернатив серед потенційних компонентів ще до стадії їх затвердження.

## **АРХІТЕКТУРА ЗАСОБУ ПІДТРИМКИ ПРОЦЕСУ ОЦІНЮВАННЯ ПОТЕНЦІЙНИХ КОМПОНЕНТІВ ПОВТОРНОГО ВИКОРИСТАННЯ**

### **ARCHITECTURE OF THE SUPPORT TOOL FOR THE EVALUATION OF POTENTIAL REUSE COMPONENTS**

З метою визначення та затвердження компонентів, які в подальшому можуть бути використані в якості повторного використовуваних при проектуванні комп'ютерних систем, розроблено засіб автоматизації, архітектуру якого представлено на рис. 1.



**Рисунок 1.** Архітектура засобу підтримки процесу оцінювання потенційних компонентів повторного використання

Як видно з рис. 1, основними компонентами засобу автоматизації є:

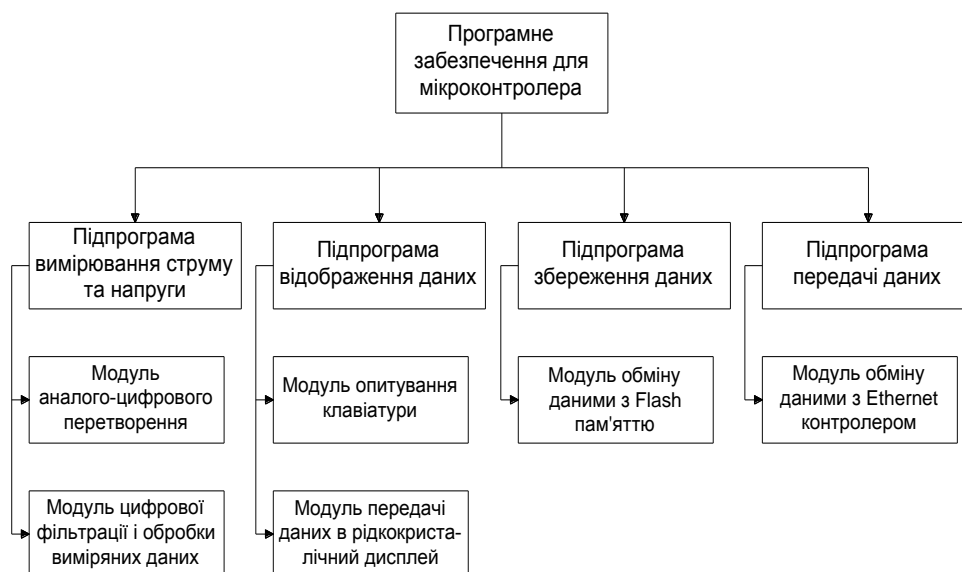
- вимірювач кількісних значень атрибутів компонента – призначений і використовується при обчисленні кількісних значень атрибутів компонента і формує результат вимірювання;
- інструменти аналізу даних – дають змогу забезпечити та уточнити існуючі залежності між оцінками і метриками атрибутів компонентів;
- користувацький інтерфейс експерта – інтерфейс, що забезпечує діалог експерта з системою оцінювання;
- ПЗ управління – відповідає за узгодженість роботи компонентів системи;
- вимірювач значень атрибутів – забезпечує одержання та інтерпретацію значень метрик на основі аналізу програмного коду компонента;
- база даних – відповідає за збереження експертних оцінок і метрик, результатів вимірювань та іншої додаткової і службової інформації;
- база знань – використовується для зберігання та опрацювання знань про залежності між властивостями і метриками компонентів повторного використання.

## ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ ДЛЯ МОНІТОРИНГУ СТАНУ ПРИСТРОЇВ БЕЗПЕРЕБІЙНОГО ЖИВЛЕННЯ

## COMPUTER SYSTEM SOFTWARE FOR CONDITION MONITORING OF UNINTERRUPTIBLE POWER SUPPLY DEVICES

Для забезпечення високого рівня якості електропостачання, захисту електронних приладів від виходу з ладу використовують пристрої безперебійного електроживлення (ПБЕЖ). Одним з найважливіших компонентів сучасних ПБЕЖ є модуль моніторингу і керування. Актуальною задачею є модернізація мікропроцесорного модуля шляхом додавання можливості передачі даних з використанням технології Ethernet, яка дозволить здійснювати контроль електричних параметрів ПБЕЖ для віддалених об'єктів в реальному часі.

В роботі розроблене програмне забезпечення для мікроконтролера системи моніторингу стану ПБЕЖ [1] написане на мові C, структура якого наведена на рис. 1. Результати моніторингу (значення струмів та напруг, стан аварійних сигналів, рівень заряду акумуляторів) передаються до віддаленого ПК для відображення та зберігання.



**Рисунок 1.** Структура програмного забезпечення для мікроконтролера ПБЕЖ

Отже, розроблене програмне забезпечення для моніторингу пристроїв безперебійного електроживлення на основі використання технології Ethernet забезпечила можливість дистанційного контролю за основними електричними параметрами та станом електроживильного обладнання.

### Література

1. Palamar A. Intelligent control and monitoring module for uninterruptible power supply system. II International Scientific and Practical Conference «Theoretical and Applied Aspects of Device Development on Microcontrollers and FPGAs», Kharkiv, Ukraine. 2020. P. 12–13.

## СТРУКТУРА МОДУЛЯ ДЛЯ МОНІТОРИНГУ СТАНУ ПРИСТРОЮ БЕЗПЕРЕБІЙНОГО ЖИВЛЕННЯ

## MODULE STRUCTURE FOR CONDITION MONITORING OF UNINTERRUPTIBLE POWER SUPPLY DEVICE

В час військової агресії в Україні, яка супроводжується ракетними атаками ворога на лінії електропостачання, актуальною проблемою є забезпечення безперебійного електропостачання об'єктів критичної інфраструктури. Одним із способів досягнення цього є застосування пристроїв безперебійного електроживлення (ПБЕЖ). Важливою задачею є моніторинг їх стану для забезпечення стабільної та довготривалої роботи [1].

Метою даної роботи є розробка структури модуля для моніторингу стану ПБЕЖ, яка зображена на рис. 1. Сигнали струму та напруги знімаються з вхідних та вихідних ліній живлення за допомогою шунтів та подільників. Отримані сигнали підсилюються та подаються на вхід аналого-цифрового перетворювача, який інтегрований в мікроконтролер. Енергонезалежна пам'ять даних призначена для зберігання параметрів та налаштувань ПБЕЖ. Контролер інтерфейсу Ethernet служить для передачі дани в ПК. Кнопки керування призначені для зміни параметрів системи та для перегляду налаштувань, архіву і вимірних даних на LCD-дисплеї.

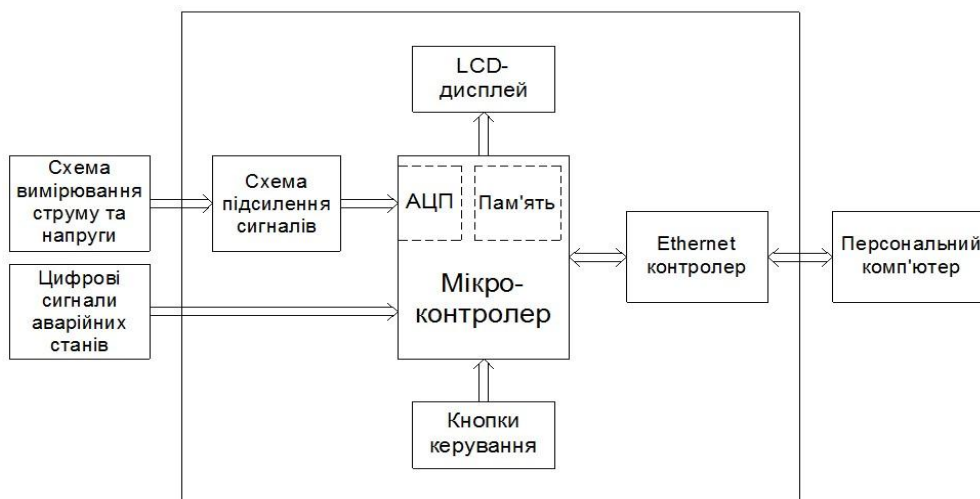


Рисунок 1. Структурна схема модуля для моніторингу ПБЕЖ

На основі структурної схеми розроблена електрична схема та прототип модуля для моніторингу стану ПБЕЖ. Його тестування показало високу ефективність запропонованого рішення.

### Література

1. Паламар А. М., Осов'як І. І. Комп'ютерна інформаційно-вимірювальна система для моніторингу пристроїв безперебійного електроживлення. Матеріали V Міжнародної науково-технічної конференції «Світлотехніка й електротехніка: історія, проблеми, перспективи», Тернопіль. 2015. С. 111–112.

УДК 681.518.3

**А. Паламар, І. Купратий**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **СИСТЕМА ДЛЯ ДИСТАНЦІЙНОГО МОНІТОРИНГУ СТАНУ ЗДОРОВ'Я ПАЦІЄНТІВ НА ОСНОВІ ІНТЕРНЕТУ МЕДИЧНИХ РЕЧЕЙ**

UDC 681.518.3

**A Palamar, I. Kupratyi**

## **PATIENT HEALTH REMOTE MONITORING SYSTEM BASED ON INTERNET OF MEDICAL THINGS**

Сучасна концепція в сфері охорони здоров'я потребує впровадження безперервного віддаленого моніторингу фізіологічного стану пацієнта з використанням мініатюрних датчиків на тілі людини [1]. Сенсорні мережі дають змогу контролювати відхилення показників від норми. Такий підхід забезпечує економічно вигідну та ефективну альтернативу моніторингу фізіологічного стану пацієнтів в лікарнях. Проте в нашій країні на сьогоднішній день відсутня розвинена інфраструктура для реалізації віддаленого спостереження за станом пацієнта. Тому розробка та впровадження методів та засобів дистанційного моніторингу стану здоров'я людини в режимі реального часу є актуальною задачею.

В роботі запропоновано використати концепцію інтернету медичних речей (ІоМТ) [2] для створення системи моніторингу стану здоров'я людини. Застосування ІоМТ в сфері охорони здоров'я відіграє ключову роль у профілактиці виникнення хвороб та у процесі лікування хронічних захворювань [3]. Новітні технології в сфері інтернету медичних речей, бездротових комунікацій та мініатюрних високоточних датчиків сприяють у вирішенні існуючих проблем та пропонують шляхи підвищення якості надання медичних послуг.

Центральним компонентом розробленої системи моніторингу стану здоров'я є пристрій на базі мікроконтролера ESP32 з вбудованим Wi-Fi модулем. Він передбачає розміщення мініатюрних датчиків на тілі людини, що дозволяє здійснювати моніторинг її фізіологічних параметрів. Пристрій отримує дані від датчиків та надсилає їх до хмарної платформи для зберігання та аналізу.

Розроблена система для дистанційного моніторингу стану здоров'я дає змогу здійснювати постійний контроль фізіологічних параметрів пацієнтів медичних закладів та попередження виникнення в них критичного стану. Процес моніторингу здоров'я здійснюється з метою спостереження за станом пацієнтів, аналізу наслідків лікування та ранньої діагностики захворювань. Це дозволяє уникнути частого відвідування лікарень та економити витрати на оплату медичних послуг.

### **Література**

1. Марценюк В. П., Качур І. В., Сверстюк А. С., Бондарчук В. І., Завіднюк Ю. В., Коваль В. Б., Мочульська О. М. Моніторинг стану здоров'я за функціональними показниками за допомогою сенсорів у реабілітаційній медицині: систематичний огляд. Вісник наукових досліджень, № 2. 2019. С. 5–12.
2. Vishnu S., Ramson S. J., Jegan R. Internet of Medical Things (IoMT) – An overview. 5th international conference on devices, circuits and systems. 2020. P. 101–104.
3. Харковлюк-Балакіна Н. В., Горго Ю. П., Медвидчук К. В. Імплементация мониторинговых технологий контролю стану здоров'я пацієнтів для населення територіальних громад. Біомедична інженерія і технологія, № 4. 2020. С. 107–116.

УДК 621.396

**В. Ліщина, Р. Жаровський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## МЕТОДИ ПІДВИЩЕННЯ ПРОПУСКНОЇ ЗДАТНОСТІ В МЕРЕЖАХ LTE

UDC 621.396

**V. Lishchyna, R. Zharovskyi**

## METHODS OF INCREASE BANDWIDTH IN LTE NETWORKS

Основною характеристикою будь-якої мережі зв'язку, як системи розподілу інформації, є ефективність її функціонування. Інтенсивне зростання трафіку ставить перед операторами завдання оптимізації доступу до ресурсів мережі. Технологія LTE є найбільш придатною для застосування алгоритмів динамічного управління пропускну здатністю мережі. Тому проведемо огляд та аналіз методів управління пропускну здатністю у мобільних мережах з метою розробки оптимального методу управління.

Під пропускну здатністю мережі зв'язку мають на увазі

$$D = \sum_{\forall i,j} c_{ij} l_{ij} ,$$

де  $c_{ij}$  – номінальна пропускну здатність лінії між вузлами  $i$  та  $j$  в біт за одиницю часу, або Ерл при заданій якості обслуговування;  $l_{ij}$  – довжина лінії між вузлами  $i$  та  $j$  у км.

Продуктивність мережі зв'язку визначають наступним чином:

$$P = \sum_{\forall s,t} V_{st} l_{st} / T ,$$

де  $V_{st}$  – обсяг переданих за час  $T$  повідомлень (у бітах) між кінцевими пунктами  $s$  і  $t$ ;  $l_{st}$  – довжина найкоротшого шляху між цими пунктами,  $T$  – час.

Одною з характеристик, що визначають рівень завантаженості обладнання системи протягом доби, є коефіцієнт концентрації навантаження. Величина коефіцієнта концентрації переважно залежить від структурного складу абонентів мобільної мережі і зазвичай лежить у межах 0,09-0,15. Чим рівномірніший графік цього коефіцієнта концентрації протягом доби, тим вище пропускну здатність мережі.

Існуючі способи підвищення пропускну здатності мобільних мереж наступні:

- використання пільгових тарифів.

- диференційоване обслуговування абонентів. поділ абонентів на категорії, за якого деякі обслуговуються з підвищеною якістю. Спочатку цей спосіб застосовувався в телеграфії (термінові телеграми) та на міжміській мережі (терміновий виклик з 3-кратною оплатою) при напівавтоматичному встановленні з'єднання. В даний час він знаходить широке застосування в мобільних мережах за допомогою різних рівнів обслуговування різних користувачів. При цьому знижується взаємний вплив користувачів різних категорій

- зниження навантаження за рахунок відмови від попереднього резервування каналів. У мережах стандарту GSM, згідно зі специфікацією MCE -T Q.764, при передачі по мережі ОКС-7 початкового адресного повідомлення одночасно відбувається резервування розмовних каналів в інформаційній мережі.

- оперативне керування мережею. Застосовується для зменшення навантажень, які різко знижують пропускну здатність мережі. При цьому застосовуються методи оперативного керування мережею, які передбачають динамічне управління потоками як користувальницької, так і керуючої інформації, аж до обмеження доступу користувачів в мережу.

Саме цей спосіб і буде розглянутий у роботі, оскільки поява нових алгоритмів керування обладнанням дозволяє реалізувати його без втрат якості.

УДК 004.93

**О. Марчук**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **МЕТОД ІДЕНТИФІКАЦІЇ ДОРОЖНІХ ЗНАКІВ НА ОСНОВІ ЗГОРТКОВОЇ НЕЙРОМЕРЕЖІ**

UDC 004.93

**O. Marchuk**

## **ROAD SIGN IDENTIFICATION METHOD BASED ON A CONVULSIONAL NEURAL NETWORK**

Головним завданням спеціалізованих систем автоматичного повідомлення водія про дорожню ситуацію є виявлення та розпізнавання дорожніх знаків (ДЗ). Алгоритмів, котрі лежать в основі такої системи, розроблено багато, проте рівень їх точності та швидкості детектування не завжди дає змогу якісно виявляти та ідентифікувати об'єкти із заданою точністю. Отже, проблема створення такого методу, який швидко та якісно розпізнаватиме ДЗ залишається актуальною.

При використанні автоматичної системи розпізнавання надзвичайно важливим є правильна і своєчасна ідентифікація ДЗ у автомобілі під час його руху. Сьогодні для виконання завдання детектування та розпізнавання застосовуються різні комерційні системи, які встановлюються «у комплекті» з автомобілем. Вони є закритими, наприклад, компанія Opel із системою «Opel Eye», Mercedes – «Speed limit assist», Volvo – «Road sign information». Такі засоби є вбудованими в автомобільну систему як опція та не підлягають модифікації чи покращенню. Окрім того результативність таких систем різко падає за наявності різного рівня розмитості, шуму, поганого освітлення та за різних спотворень ДЗ.

Для вирішення задачі ідентифікації ДЗ пропонується використовувати разом два алгоритми: пошуку за формою із використанням фільтра Габоора та метод максимальних екстремальних областей (MSER). Схема роботи алгоритмів є такою: зображення із навчальної вибірки подаються на згорткову нейронну мережу (ЗНМ) для навчання та тестування; інша вибірка подається послідовно на два алгоритми і після їх роботи детектовані області подаються на навчену ЗНМ для розпізнавання.

Були проведені експериментальні запуски архітектури ЗНМ для пошуку кращої точності. Визначена архітектура ЗНМ, яка дає найкращі результати ідентифікації.

Побудований на основі запропонованого алгоритму програмний засіб, здатен ідентифікувати основні з видів ДЗ. У програмі реалізовані всі складові, потрібні для коректної роботи за різних умов, зокрема налаштування фільтрів, автоматичне форматування різних видів вхідних та вихідних даних.

Для вирішення завдань та реалізації алгоритмів обрано мову Python; для реалізації програмного продукту – інтерпретатор Anaconda Python версії 3.6 та середовище розробки Spyder; також середовищем для розробки визначено Google Colaboratory – хмарний сервіс, спрямований на спрощення досліджень у галузі машинного та глибокого навчання; для реалізації розпізнавання обрано відкриту нейромережеву бібліотеку Keras, написану мовою Python; використано OpenCV – бібліотеку функцій та алгоритмів комп'ютерного зору і обробки зображень; застосовано NumPy – розширення мови Python, що має підтримку великих багатовимірних масивів і матриць.

Звичайно, в реалізації залишаються місця, які можна оптимізувати для кращої роботи, а також можна реалізувати розпізнавання комбінованих (які складаються з різних геометричних форм) ДЗ, вигляд яких не дозволяє розпізнавати їх на даному етапі.

УДК 004.93

**I. Мудрий**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ЛОКАЛІЗАЦІЯ ТА КЛАСИФІКАЦІЯ ОБ'ЄКТІВ НА ЗОБРАЖЕННІ**

UDC 004.93

**I. Mudryi**

## **LOCATION AND CLASSIFICATION OF IMAGE OBJECTS**

Процес розпізнавання предмету у неструктурованих сценах – складна сфера поточних досліджень комп'ютерного зору. Робототехніка є одним із важливих напрямків, де здатність чітко і швидко ідентифікувати об'єкти, що цікавлять, має вирішальне значення для універсальних роботів, що виконують завдання в неструктурованому повсякденному середовищі, такій як будинок, офіс або склад.

Традиційна система розпізнавання об'єктів завжди опрацьовує значне число різних об'єктів. Один із способів упоратися з цим – запровадити ієрархію, виконуючи розпізнавання на рівні категорії, а не на рівні примірника. Завдання узагальнення від кількох екземплярів до цілої категорії об'єктів залишається складним, і існують численні тести та завдання, такі як Caltech 256 та PASCAL VOC, щоб сприяти прогресу у цій галузі. З іншого боку, для конкретного робота в конкретному середовищі кількість унікальних об'єктів відносно невелика (можливо, близько сотень). Це дозволяє розглядати завдання як проблему розпізнавання екземплярів, збираючи великий обсяг навчальних даних кожного об'єкта. Крім того, робот може використовувати дані, одержані від різних сенсорних пристроїв, таких як камери та датчики глибини.

Локалізація об'єктів у робототехніці також є додатковими проблемами, яких немає в тестах розпізнавання об'єктів на рівні категорій. Середовищ реального світу дуже захарашені, містять багато перекриттів і часто містять більшу кількість різних об'єктів в одній сцені. Роботам часто доводиться взаємодіяти з одними об'єктами у своєму середовищі, оминаючи інші. Це означає, що роботизована система сприйняття повинна точно локалізувати об'єкти після виявлення. Крім того, щоб робот міг швидко реагувати на зміни у навколишньому середовищі, роботизована система сприйняття має працювати у режимі реального часу. В даний час існує ряд різних підходів до розпізнавання та локалізації предметів у складних неструктурованих сценах. Першим є використання сенсорів для збирання кольорових та глибинних зображень. Прикладами даних сенсорів є різні лідари, або більш прості давачі типу Kinect. Під час навчання розпізнаванню та локалізації об'єктів на основі інформації про глибину вдається точніше виділити контури кожного об'єкта. На основі контурів, кольору та глибини будується повна хмара точок, з якої виділяються тривимірні сіткові моделі об'єктів. Потім з отриманих моделей витягуються і реєструються локальні особливості об'єктів.

Розпізнавання об'єктів за локальними ознаками на зображеннях відбувається у кілька етапів. Спершу знаходяться відповідності. Цей етап відноситься до точного зіставлення характеристик зображення з характеристиками, що належать певному предмету. Далі йде процес оцінки пози об'єкта, яка буде геометрично узгодженою зі знайденими відповідностями. Після цього відповідно до масштабу предмета у базі даних і зображення знаходиться точна відстань до об'єкта. Важливо, що вибір камери та об'єктива може сильно вплинути на точність оцінки пози. Тим не менш, камери стають якіснішими та дешевшими, а їх габарити зменшуються. Це призводить до того, що стають легко доступними паралельне використання декількох камер для отримання виду сцени з різних ракурсів та високою роздільною здатністю. Цей підхід ефективний, тому що він забезпечує оцінку глибини, стійкість до оклюзій прямої видимості та зростання ефективного кола огляду.



УДК 004.622

**Т. Патральський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ЗБЕРІГАННЯ ТА ТРАНСФОРМАЦІЯ ДАНИХ У ХМАРНОМУ СЕРЕДОВИЩІ GOOGLE CLOUD BIGQUERY**

UDC 004.622

**T. Patralskyi**

## **DATA STORAGE AND TRANSFORMATION IN THE CLOUD ENVIRONMENT GOOGLE CLOUD BIGQUERY**

BigQuery – це безсерверне мультихмарне сховище даних, яке спрощує процес роботи з усіма типами даних, тож можна зосередитися на швидкому отриманні цінної бізнес-аналітики. BigQuery, який є ядром хмари даних Google, дозволяє спростити інтеграцію даних, економічно й безпечно масштабувати аналітику, ділитися багатими даними за допомогою вбудованої бізнес-аналітики, а також навчати та розгортати моделі ML за допомогою простого інтерфейсу SQL [1].

До особливостей BigQuery відносяться:

- керування даними – можна створювати та видаляти такі об'єкти, як таблиці, подання та визначені користувачем функції, імпортувати дані з Google Storage у таких форматах, як CSV, Parquet, Avro або JSON;

- запити, які виражаються стандартним діалектом SQL, а результати повертаються у форматі JSON із максимальною довжиною відповіді приблизно 128 МБ або необмеженим розміром, якщо ввімкнено великі результати запиту;

- інтеграція - BigQuery можна використовувати з Google Apps Script (наприклад, як зв'язаний сценарій у Google Docs ) або з будь-якої мови, яка може працювати з REST API або клієнтськими бібліотеками.

BigQuery має багато інтегрованих платформ, найпопулярніші та часто застосовані це Google spreadsheet та REST API, де з останнього можна отримувати дані в реальному часі з різних платформ які дозволяють це робити.

Щоб використати API, потрібно використати одну з мов програмування, краще всього підійде python, за допомогою якого можна написати скрип для витягування даних, обробки та агрегації даних з потрібної вам платформи та записати її у хмарне сховище Bigquery, для того щоб оновлювалися дані потрібно це скрипт у Google cloud function, щоб дані постійно були актуальними. Після цього потрібно записати робоче місце, де можна написати SQL запит для бази даних Bigquery та зробити потрібні операції для кращого виводу даних.

У доповіді розглянуто основні етапи трансформації відкритих даних та їх зберігання і оновлення, наведено приклади практичної реалізації Google Cloud Bigquery.

### **Література**

1. Огляд Google Cloud BigQuery. G2. – 2022. URL: <https://www.g2.com/products/google-cloud-bigquery/reviews>.
2. Trino 403 Documentation. trino.io. URL: <https://trino.io/docs/current/connector/bigquery.html>.

УДК 004.9

**В. Савчук, Н. Луцик**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АНАЛІЗ ІСНУЮЧИХ СИСТЕМ КЛІМАТ-КОНТРОЛЮ**

UDC 004.9

**V. Savchuk, N. Lutsyk**

## **ANALYSIS OF EXISTING CLIMATE CONTROL SYSTEMS**

Навколишнє природне середовище є засобом, джерелом та місцем життєдіяльності людини, існування всіх інших біологічно активних організмів [1].

За останні десятиріччя, що минули відбулися масштабні зміни як у розвитку технічного забезпечення спостережних систем, так і теоретичних обґрунтувань необхідних для удосконалень існуючої екологічної системи. Необхідність оновлення екологічної системи моніторингу зумовлена також значними змінами у номенклатурі виробництв, у кількості транспортних засобів, у розвитку урбанізаційних процесів [2].

Значне погіршення клімату в загальному відображається і на мікро-клімат приміщень, в якому люди постійно знаходяться.

Відхилення від нормативних параметрів в мікрокліматі викликає незадовільний стан та порушення нормальної роботи організму. При довготривалому перебуванні в погіршених умовах у людини можуть розвиватися хронічні симптоми та недуги.

Як приклад це аудиторія, приміщення, в якому знаходяться студенти та викладачі. Після кожного тривалого перебування в аудиторії, потрібно її провітрювати. Появляється необхідність в системі з датчиком вуглекислого газу, яка могла би сповіщати присутніх про рівень забрудненості і контролювала якість повітря з-за допомогою підключеного до мережі кондиціонера або ж вбудованого моторчика у рамі вікна.

Таким чином, системи клімат-контролю набули неабиякої актуальності на сьогоднішній день.

Система, що дозволяла б ефективно керувати мікрокліматом приміщення, містить у собі цілісне, комплексне вирішення різноманітних завдань. Сюди входить моніторинг відносної вологості, температури приміщення, швидкість руху повітряних мас, а також концентрація кисню, вуглекислого газу та етилену в повітрі, тощо.

Наразі, у побуті можна зустріти готові фабричні рішення систем моніторингу, в яких реалізовані датчики, які відображають дані на дисплеї. Основні недоліки даних пристроїв моніторингу:

- 1) відсутність довговічного зберігання даних;
- 2) відсутність передачі даних до мережі;
- 3) робота від електромережі.

При таких тенденціях, постає питання у реалізації системи, яка змогла би зберігати дані у внутрішній пам'яті, надсилати їх в онлайн-середовища для зручного представлення, а також працювати при відсутності електромережі від акумуляторної батареї.

На основі Arduino та під'єднованих до нього модулів можна самотужки реалізувати систему моніторингу та клімат-контролю, врахувавши основні недоліки інших пристроїв.

### **Література**

1. Берзіна С. В., Борейко В. Є., Бузан Г. С. Громадський екологічний контроль. Київ: Інститут екологічного управління та збалансованого природокористування, 2017. 150 с.
3. Бахарев В. С., Маренич А. В. Аналітичний огляд результатів наукових досліджень з проблем моніторингу довкілля в Україні. Екологічна безпека. № 2. 2016. С. 35–42.

УДК 004.9

**В. Савчук, Н. Луцик**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **РОЗРОБКА СИСТЕМИ КЛІМАТ-КОНТРОЛЮ НА БАЗІ МІКРОКОНТРОЛЕРА ТА СЕНСОРІВ**

УДК 004.9

**V. Savchuk, N. Lutsyk**

## **DEVELOPMENT OF THE CLIMATE CONTROL SYSTEM BASED ON THE MICROCONTROLLER AND SENSORS**

Система моніторингу даних є основою для будь-якої системи клімат-контролю [1].

Наразі існує багато портативних систем моніторингу температури, вологості, забрудненості повітря. Кожна з них відрізняється: кількістю датчиків, тривалістю роботи, можливістю підключення до смартфона, ціною тощо.

Серед великого різноманіття можна виділити такі основні тенденції:

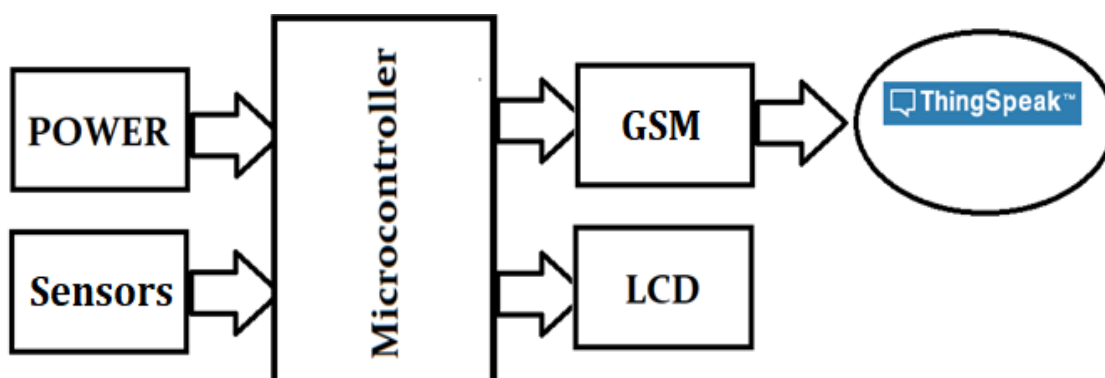
1) Дешеві пристрої, які працюють на простих схемах і можуть показувати результати лише на вбудованому дисплеї. Переваги таких систем це ціна, портативність та помірна тривалість роботи. Недоліки дешевих пристроїв – малий перелік функцій.

2) Дороговартісні пристрої містять у собі складнішу елементну базу, відповідно – більш точніші дані з датчиків, їхній запис у пам'ять та підключення до інших пристроїв, наприклад, до смартфона. Основним недоліком дорогих систем моніторингу клімату є те, що вони відправляють дані за допомогою вбудованого Wi-Fi модуля і при відсутності мережі пристрій не відсилає дані.

Для того щоб вирішити дану проблему запропоновано портативну систему клімат-контролю на базі Arduino, яка за допомогою GSM модуля та акумуляторної батареї постійно моніторить дані з датчиків та відсилає їх у онлайн-сервіс ThingSpeak (рис. 1).

ThingSpeak – це програма з відкритим кодом Internet of Things (IoT) для зберігання та отримання даних за допомогою протоколу HTTP та MQTT. ThingSpeak дозволяє створити програми реєстрації журналів, відстеження місцеположення та соціальні мережі речей із оновленнями статусу. Дані з датчиків будуть постійно надходити в онлайн-сервіс. У разі виявлення неполадки або перевищення допустимих норм, ThingSpeak має змогу повідомити про це користувача у його соціальних мережах.

Така система буде виконувати свої функції навіть при відсутності електроживлення в приміщенні.



**Рисунок 1.** Структурна схема системи клімат-контролю

### **Література**

1. Bert Metz. Controlling Climate Change. Cambridge University Press, 2010. 359 p.

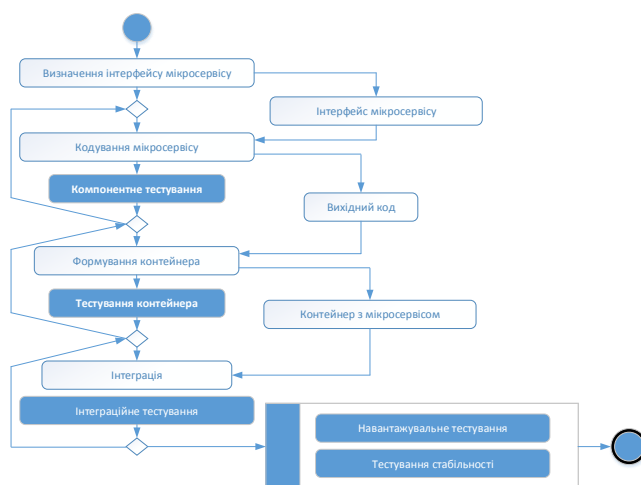
## ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПОБУДОВАНОГО НА МІКРОСЕРВІСІЙ АРХІТЕКТУРІ

### TESTING OF SOFTWARE BUILT ON MICROSERVICE ARCHITECTURE

Останнім часом особливої актуальності у сфері інформаційних технологій набувають питання якості програмного продукту. Якість програмного продукту безпосередньо залежить від того, наскільки результат розробки відповідає очікуванням замовника. Одним із методів забезпечення якості програмних продуктів є тестування.

Тестування програмного забезпечення є по суті послідовним проходженням програми через створені тестові набори (тест-кейси), які дозволяють всебічно перевірити кожен окрему функцію та весь функціонал у цілому.

Мікросервісна архітектура стала найкращою альтернативою для монолітних, складних та негнучких додатків. При тестуванні ІТ -продукту, побудованого на мікросервісній архітектурі, вся система поділяється більш дрібні, модульні, спільно працюючі компоненти. Їх набагато простіше створювати, оновлювати та тестувати, ніж додаток загалом. Весь процес розробки та тестування мікросервісів представлений на рисунку 1.



**Рисунок 1.** Етапи розробки та тестування ПЗ побудованого на мікросервісній архітектурі

Особливістю автоматизованого тестування ІТ -продуктів, побудованих на мікросервісній архітектурі полягає в необхідності автоматизованої інфраструктури, що забезпечує управління життєвим циклом, адресацію та масштабування мікросервісів залежно від поточного завантаження. Необхідність безперервної інтеграції розроблених та/або модифікованих мікросервісів у існуючу систему вимагає всебічного тестування як окремих мікросервісів, так і їхнього спільного функціонування в комплексі з іншими мікросервісами

Таким чином, при виборі будь-якого сценарію тестування мікросервісного ІТ -продукту необхідне всебічне тестування як окремих мікросервісів, так і їхнього спільного функціонування в комплексі з іншими мікросервісами. Ефективне тестування мікросервісного ІТ -продукту неможливе без використання тестових дублерів, які спеціально розробляються для задоволення необхідних вимог недоступних модулів.

## ТЕСТУВАННЯ ПРОГРАМНОГО ПРОДУКТУ, ПОБУДОВАНОГО НА МІКРОСЕРВІСНІЙ АРХІТЕКТУРІ НА ОСНОВІ BDD

### TESTING OF SOFTWARE PRODUCT BUILT ON MICROSERVICE ARCHITECTURE BASED ON BDD

Відповідно до методології BDD (Behavior Driven Development) тест-кейси в приймальному тесті повинні охоплювати всі функції програми за різними сценаріями користувачів. BDD методологія передбачає принцип, що полягає в тому, що перед написанням тест-кейсу слід спочатку визначити і сформулювати предметно-орієнтованою мовою результат від проєктованої функціональності. Потім усі складені тести перебудовуються фахівцями в BDD-сценарій тестування.

Відповідно до методології BDD необхідно визначити наступні пункти:

- із чого починається процес тестування;
- яку функціональність слід тестувати, а яку ні;
- скільки перевірок відбувається одночасно;
- який тест є перевіркою;
- у якому випадку тест вважається незавершеним чи результат некоректним.

Тобто дана методологія передбачає, що імена тестів повинні являти собою цілі пропозиції, які починаються з дієслова в умовному способі і відображають суть бізнес-мети. BDD-сценарій складається із пропозицій, побудованих з деяких елементів:

- конструкція Given визначає початкові умови здійснення операції (визначає те, що спочатку «дано»). Наприклад, вікно введення команди, пошуковий рядок тощо;
- слово When визначає дії, які здійснюють користувач або підсистема і ініціюють процес тестування функції (відповідає на питання «коли?»);
- фраза Then описує очікуваний результат тестування (наприклад, перехід іншу сторінку чи вибірка за заданими критеріями).

**Feature:** To check that home page has loaded in Sistem Informasi SPI  
**Skenario:** To check that home page has loaded  
**Given** I am on Sistem Informasi SPI  
**When** I click on the Login Link  
**And** input username and password  
**Then** I should be on the Home SPI page

#### Рисунок 1. Вхід до тестованої системи відповідно до методології BDD

Відповідно до рисунка 1 сценарій входу в тестовану систему описує, що користувач знаходиться на сайті System Informasi SPI (GIVEN). При натисканні на клавішу «Авторизуватися» (WHEN) та введення імені користувача та пароля (AND) потрапляє на сторінку Home SPI page (THEN).

Перевагою методології BDD є те, що автотести створюються одночасно за участю і функціональних тестувальників, і фахівців з тестування, що дозволяє заощадити робочий час і бюджет ІТ-проєкту. Крім того повний набір критеріїв якості програмного продукту формується не тільки тестувальниками, а всіма учасниками процесу розробки. Тобто даний підхід є найбільш ефективний з точки зору автоматизації. Він дає швидке та дуже ефективне складання BDD-сценарію тестування.

УДК 004.416.2

**І. Слюз, Р. Жаровський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ПРИНЦИПИ ТА ОСНОВНІ ЕТАПИ КОМПЛЕКСНОГО ТЕСТУВАННЯ КОМП'ЮТЕРНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ**

UDC 004.416.2

**I. Slyuz, R. Zharovskyi**

## **PRINCIPLES AND MAIN STAGES OF COMPLEX TESTING OF A COMPUTER INFORMATION SYSTEM**

Тестування комп'ютерної інформаційної системи охоплює основні стадії життєвого циклу, аналогічний до послідовності процесів розробки програмного забезпечення: постановка задачі для тесту, проектування, написання тестів, тестування тестів, виконання тестів та вивчення результатів тестування

Якщо врахувати, що комп'ютерна інформаційна система - це не тільки програмні компоненти, що використовуються в її складі, але й апаратне та організаційне забезпечення, то і в результатах її випробувань повинні бути відображені показники роботи серверів, робочих станцій, мережевого обладнання (їх надійність та продуктивність), а також ефективність розробленого регламенту експлуатації системи. У зв'язку з цим виникає необхідність у проведенні комплексного тестування на відповідність усім вимогам, що висуваються.

Комплексне тестування комп'ютерної інформаційної системи складається з наступних етапів:

1. Детальне вивчення проекту системи та її експлуатаційних документів.
2. Створення та впровадження автоматизованої системи відстеження помилок.
3. Безпосереднє тестування роботи інформаційної системи, яке включає:
  - тестування роботи всіх модулів інформаційної системи,
  - перевірка внутрішньосистемних зв'язків,
  - тестування обладнання, у тому числі на наявність необхідних ліцензій,
  - перевірка програмного забезпечення, у тому числі перевірка коду,
  - тестування продуктивності та максимальних навантажень інформаційної системи,
  - тестування на відмови: несподівані програмні збої, вихід з ладу модулів системи, людський фактор та ін.
  - тестування на захищеність інформаційної системи – включає комплекс досліджень з виявлення способів злому системи і витоків інформації,
  - загальна перевірка роботи системи.

4. Тестування роботи системи управління ІТ-структурою.

5. Тестування персоналу.

6. Аналіз отриманих даних та вироблення стратегії з виправлення виявлених помилок.

Комплексне тестування не призначене для тестування всіх функцій повністю зібраної системи. Воно спрямовано на пошуку невідповідності системи її вихідним цілям [1]. Таким чином, у комплексному тестуванні бере участь комп'ютерна інформаційна система, опис її цілей та вся документація, яка поставлятиметься разом із системою.

### **Література**

1. Changyou Xing, Guomin Zhang, Ming Chen. Research on Universal network performance testing model. International Symposium on Communications and Information Technologies, 2007. P. 780–784.

УДК 004.052.3

**В. Тимошук, Д. Тимошук**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ВІРТУАЛІЗАЦІЯ В ЦЕНТРАХ ОБРОБКИ ДАНИХ - АСПЕКТИ ВІДМОВОСТІЙКОСТІ**

UDC 004.052.3

**V. Tymoshchuk, D. Tymoshchuk**

## **VIRTUALIZATION IN DATA CENTERS – ASPECTS OF FAILURE TOLERANCE**

Сучасні центри обробки даних повністю віртуалізовані. Віртуалізація серверів це ефективне використання наявних ресурсів ІТ-інфраструктури. Без віртуалізації сервери використовують лише невелику частину своєї обчислювальної потужності. Це призводить до простою. Центри обробки даних стають переповненими недостатньо завантаженими серверами, що призводить до марної витрати ресурсів і енергії.

Вибираючи гіпервізор для віртуалізації в центрах обробки даних потрібно враховувати, окрім вартості та продуктивності такий основний параметр, як безвідмовність кластера. Гіпервізори, які орієнтовані на датацентри, як правило не допускають жодних помилок або інших проблем із продуктивністю. Але ніхто не застрахований від виходу з ладу апаратного забезпечення гіпервізора.

Однозначно, що у центрах обробки даних використовуються гіпервізори першого типу (bare metal). Як правило вибір гіпервізора здійснюється між VMware ESXi, Microsoft Hyper-V, XEN (Citrix XenServer, XCP-ng Xen Hypervisor) і KVM (Oracle Linux KVM, Red Hat Virtualization). Усі перелічені вище типи гіпервізорів мають можливість створення кластера високої доступності (High Availability). Що дає високий рівень безвідмовності, але не майже стовідсотковий. Справа в тому, що у більшості випадків при відмові фізичного вузла кластера, віртуальні машини автоматично перезапускаються на іншому, і, коли вони запускаються, користувач не може взаємодіяти з ними. Виникає певний період простою (Down Time).

Проведені дослідження в лабораторних умовах можливостей кластерів, побудованих на чотирьох типах гіпервізорів описаних вище, показали, що лише кластер на базі VMware дає можливість переключення на вторинну копію віртуальної машини без втрати жодного пінга.

Ось чому гіпервізор від VMware можна вважати одним із найкращих рішень для високої доступності, оскільки він має функцію відмовостійкості (Fault Tolerance).

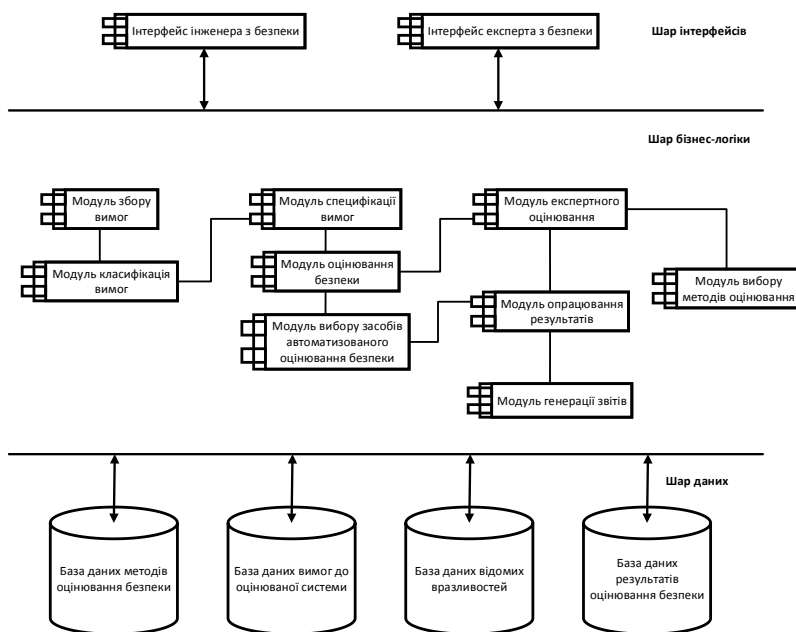
VMware Fault Tolerance розроблена для захисту критично важливих віртуальних машин з майже стовідсотковою доступністю. Для операційних систем, захищених VMware Fault Tolerance, виконується постійне копіювання віртуальної машини в реальному часі. У разі відмови одного з гіпервізорів кластера або частини кластера де розміщена робоча первинна віртуальна машини (Primary) відбувається миттєве переключення на вторинну (Secondary) копію, що працює на іншому вузлі кластера та є суцільною «тіньовою» копією первинної віртуальної машини. Створення «тіньової» копії первинної віртуальної машини в реальному часі вимагає додаткових ресурсів апаратного забезпечення, але при виході з ладу первинного вузла користувачі навіть не помічають процесу переключення на вторинний вузол.

Отже з технологією VMware Fault Tolerance повністю відсутній період простою.

## АРХІТЕКТУРА СИСТЕМИ ПІДТРИМКИ ПРОЦЕСІВ ВИЯВЛЕННЯ ТА ОЦІНЮВАННЯ ВРАЗЛИВОСТЕЙ ВЕБ-СЕРВЕРІВ У КОМП'ЮТЕРНИХ СИСТЕМАХ

## ARCHITECTURE OF THE SUPPORT SYSTEM FOR THE DETECTION AND ASSESSMENT OF WEB SERVER VULNERABILITY PROCESSES IN COMPUTER SYSTEMS

У загальному випадку архітектуру системи підтримки процесу виявлення та оцінювання вразливості web-серверів можна представити за шарами Фаулера, як показано на рис. 1.



**Рисунок 1.** Архітектура системи підтримки процесу оцінювання безпеки web-серверів

На рівні шару користувацьких інтерфейсів визначено два інтерфейси – інтерфейс інженера з безпеки та інтерфейс експерта. У шарі бізнес-логіки передбачено модуль для збору вимог до програмного забезпечення. Модуль класифікації вимог може бути реалізований як інтелектуальний модуль з реалізацією можливості автоматичної класифікації на функціональні та нефункціональні вимоги, а після цього класифікації нефункціональних вимог за критеріями безпеки web-серверів. Модуль оцінювання безпеки дає змогу проводити оцінювання безпеки web-серверів на основі вимог безпеки із застосуванням як автоматизованих засобів визначення мір відповідних критеріїв, так і експертним шляхом. База даних методів оцінювання безпеки містить назви методів оцінювання безпеки, статичні параметри для ефективної їх роботи.



## КОМП'ЮТЕРНІ ІГРИ ЯК СПОСІБ МОДЕЛЮВАННЯ ПОВЕДІНКИ РЕАЛЬНИХ КОМП'ЮТЕРНИХ СИСТЕМ

UDC 004.85

V. Yatsyshyn, V. Tsymbalistyi, V. Yatsyshyn

## COMPUTER GAMES AS A WAY OF REAL COMPUTER SYSTEMS BEHAVIOUR MODELLING

Важливу роль при проектуванні комп'ютерних систем відіграє їхнє моделювання. Одним з ефективних способів аналізу поведінки і функціональності складних комп'ютерних систем щодо аналізу траєкторії руху об'єктів є розробка симулятора у вигляді гри. Комп'ютерні ігрові системи можуть бути реалізовані різними технологіями та для різних предметних областей. Як приклад симуляції комп'ютерної системи розглянемо розроблену нами гру «Астероїди».

У даній системі є кілька сутностей, найбільш важливі з яких – корабель та астероїди. Основне завдання, яке потрібно реалізувати у грі полягає у знищенні астероїдів кораблем. Бонуси і рейтингова таблиця користувачів формуються на основі кількості знищених цілей. Для реалізації логіки гри «Астероїди» запропоновано скористатися мовою високого рівня програмування JavaScript, а для забезпечення ефективності поведінки корабля – реалізовано нейронну мережу, в основі якої лежить алгоритм її навчання за допомогою Back Propagation. Нейронна мережа забезпечує ефективність керування кораблем при збитті астероїдів. Функціональність, що покладається на нейронну мережу дозволяє встановити поточне місце об'єкта (астероїда) з певною точністю і здійснює його ураження. На рис. 1 показано один з фрагментів інтерфейсу гри, де корабель виконує потрапляння в астероїд, а також алгоритм навчання нейронної мережі.

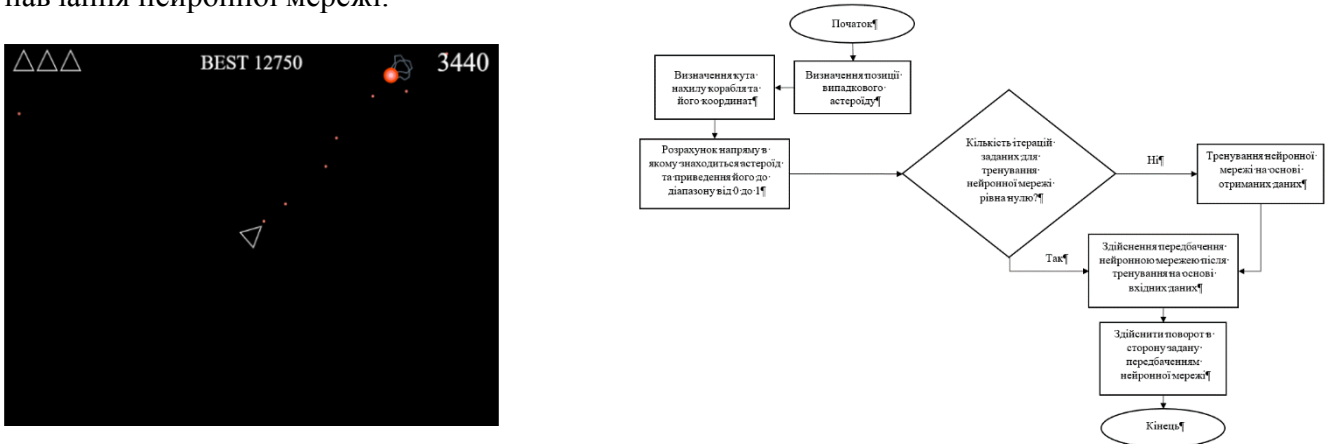


Рисунок 1. Фрагмент інтерфейсу гри «Астероїди» та алгоритм навчання нейронної мережі

Запропонована комп'ютерна гра реалізована у середовищі веб-браузера та керується за допомогою нейронної мережі з алгоритмом навчання прямого поширення.

Застосування розробленої моделі гри є доцільним у системах, де необхідна імплементація модуля з інтелектуальним перехопленням об'єктів, що рухаються.

## **ОСНОВНІ ПОНЯТТЯ СИСТЕМ ПЕРЕТВОРЮВАЧІВ СОНЯЧНОЇ ЕНЕРГІЇ**

### **BASIC CONCEPTS OF SOLAR ENERGY CONVERTER SYSTEMS**

У наш час використання не традиційних та поновлюваних джерел енергії є важливим та досить актуальним. Основним нетрадиційним поновлюваним джерелами енергії є сонце та вітера, але ще є геотермальна енергія, біомаса та енергія Світового океану.

На території України найбільш дієвою та популярною є сонячна енергія. На практиці сонячна енергія може бути перетворена в електроенергію безпосередньо чи опосередковано. Непряме перетворення може бути здійснено шляхом концентрації радіації за допомогою стеження дзеркал для перетворення води в пару і подальшого використання пара для генерування електрики звичайними способами. Така система може працювати тільки при прямому освітленні сонячними променями.

Пряме перетворення сонячної енергії в електричну може бути здійснено з використанням фотоелектричного ефекту. Елементи, виготовлені зі спеціального напівпровідникового матеріалу, наприклад, силікону, при прямому сонячному опроміненні виявляють різницю в вольтажі на поверхні, тобто наявність електричного струму.

Розрізняють три основних види перетворювачів сонячної енергії в електричну [1]:

1. Фотоелектричні перетворювачі – ФЕП-напівпровідникові пристрої, що перетворюють сонячну енергію в електрику. Кілька об'єднаних ФЕП називаються сонячною батареєю (СБ).
2. Геліоелектростанції (ГЕЕС) – сонячні установки, що використовують висококонцентроване сонячне випромінювання в якості енергії для приведення в дію теплових та ін. машин (паровий, газотурбінної, термоелектричної і ін.).
3. Сонячні колектори (СК) - сонячні нагрівальні низькотемпературні установки.

Найбільш ефективними з енергетичної точки зору пристроями для перетворення сонячної енергії в електричну (тому що це прямий, одноступінчатий перехід енергії) є напівпровідникові фотоелектричні перетворювачі (ФЕП). При характерній для ФЕП рівноважної температурі близько 300–350 кельвінів, їх граничний теоретичний ККД  $> 90\%$ . Це означає, що, в результаті оптимізації структури і параметрів перетворювача, спрямованих на зниження необоротних втрат енергії, цілком реально вдасться підняти практичний ККД до 50% і більше (в лабораторіях вже досягнуто ККД 40%) [2].

#### **Література**

1. Konar A, Mandal AK. Microprocessor based automatic Sun tracker. In: IEE Proceedings A Science, Measurement and Technology. 2015. Vol. 138. Issue 4. P. 237–241.
2. Боровик Ю. Т., Слагін Ю. В Проблеми та перспективи розвитку альтернативної енергетики в Україні. Вісник економіки транспорту і промисловості. № 65. Харків. 2019.

УДК 004.043

**В. Шаварський, Є. Тиш**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ОСОБЛИВОСТІ РОЗРОБКИ ОДНОВІСНОГО СОНЯЧНОГО ТРЕКЕРА**

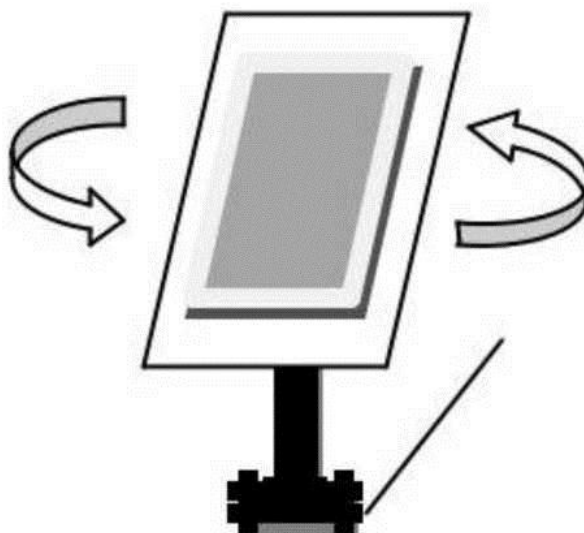
UDC 004.043

**V. Shavarskiy, Ie. Tysh**

## **FEATURES OF THE DEVELOPMENT OF A SINGLE-AXIS SOLAR TRACKER**

Сонячна енергія напряму залежить від сонячного світла і освітленості панелі. Рівень освітленості панелі не постійний, він змінюється протягом дня в залежності від положення Сонця, саме тому панель має змінювати своє положення відповідно до кута падіння сонячного світла. Звичні, стаціонарні сонячні панелі не можуть в повній мірі використовувати всю доступну сонячну енергію. Для підвищення ефективності поглинання світла та в наслідок виробництва електроенергії з меншими втратами, необхідно використовувати метод відстеження сонячного світла [1, 2]. В основі даного методу лежить відстеження руху Сонця протягом дня.

Сонячний трекер – це пристрій, за допомогою якого відстежується положення Сонця і орієнтується платформа так, щоб сонячні батареї мали найбільший ККД. На трекері розміщені датчики, які дозволяють визначити найбільш оптимальну орієнтацію сонячних панелей, а серводвигун, відповідно, повертає панель в потрібне положення. Слідкуюча система використовується для виставки сонячної панелі під таким кутом, щоб потік світла був перпендикулярний панелі. Одновісний трекер (рисунок 1) – це система, що може вільно обертатися навколо однієї осі, коли інша вісь зафіксована на визначений кут. Одноосьові трекери мають одну степінь свободи – обертання навколо осі. Вісь обертання одновісного трекера, як правило, орієнтована уздовж осі дотичної до меридіана (північ-південь).



**Рисунок 1.** Сонячний одновісний трекер

Найпопулярніший алгоритм одновісного сонячного трекера був запропонований Конаром А. та Мандалом А. К. Даний трекер розроблений на базі схеми автоматичного контролю положення. Плоский сонячний перетворювач або циліндричний параболічний відбивач мають бути закріплені в оптимальному положенні по одній осі, а трекер керує

поворотом по іншій осі, змінюючи кут азимута. Даний алгоритм був розроблений для пошуку максимального рівня сонячного освітлення в кожному положенні азимутального кута –  $360^\circ$ . Отже, трекер виконує повний оберт навколо своєї осі і не обмежений географічним положенням установки. В даній системі використовується схема ступінчастого відстеження замість безперервного відстеження, яка довше тримає двигуни в режимі холостого ходу для економії енергії [3].

В роботі А. Кассема і М. Хамада представлено алгоритм роботи сонячного трекера. Перетворювач також буде подавати живлення на вхід зарядного пристрою, який буде заряджати акумулятор постійного струму. Друга функція сонячного перетворювача полягає в тому, щоб подавати точну напругу трекера мікроконтролеру, щоб отримати найбільш ефективну орієнтацію системи, яка дозволить отримати максимальну кількість сонячного світла [4].

### **Література**

1. Гончаренко О. Р., Тиш Є. В. Системи керування сонячних трекерів. Актуальні задачі сучасних технологій: збірник тез доповідей X міжнародної науково-практичної конференції молодих учених та студентів, (Тернопіль, 24–25 листопада 2021 р.). Тернопіль, ТНТУ. 2021. С. 90.
2. Тиш Є. В., Гончаренко О. Р. Алгоритм автоматизованого режиму роботи сонячного трекера. GRAIL OF SCIENCE. № 10. 2021. С. 268–271.
3. Konar A, Mandal AK. Microprocessor based automatic Sun tracker. In: IEE Proceedings A Science, Measurement and Technology. 2015. Vol. 138. Issue 4. P. 237–241.
4. Kassem A. and Hamad M. A Microcontroller-Based Multi-Function Solar Tracking System. Department of Electrical and computer and communication Engineering, Notre Dame University Louaize. 2019. 13 (4). P. 181–184.

## СХЕМА РЕЛЯЦІЙНОЇ БАЗИ ДАНИХ ДЛЯ ЗБЕРІГАННЯ ТА ОПРАЦЮВАННЯ ВРАЗЛИВОСТЕЙ ВЕБ-СЕРВЕРІВ У РОЗУМНИХ КОМП'ЮТЕРНИХ СИСТЕМАХ

UDC 004.4

V. Yatsyshyn, B. Kharyton

## A RELATIONAL DATABASE SCHEME FOR STORING AND PROCESSING WEB SERVER VULNERABILITIES IN SMART COMPUTER SYSTEMS

Проведення кількісного оцінювання показників надійності та функціональної безпеки програмних компонентів і систем передбачає роботу з інформацією. З метою систематизації, зберігання і подальшої обробки інформації можна використовувати реляційні бази даних. Ключовим моментом є попереднє коректне проектування бази даних: вона повинна включати необхідні відношення, збережені процедури, атрибути, у базі даних повинні бути проставлені індекси і задані зв'язки між реляційними відношеннями.

Організація бази даних, структура її відношень і атрибутів повинні бути зручними для подальшої роботи з інформацією, автоматичної обробки даних та оцінювання різних характеристик. При оцінюванні OTS компонентів доцільним є побудова бази даних з урахуванням аналізу семантичної інформації з відкритих джерел даних про уразливість. На рис. 1 показана модель бази даних у вигляді діаграми «сутність-зв'язок», що відображає логічне представлення і взаємозв'язок основних відношень.

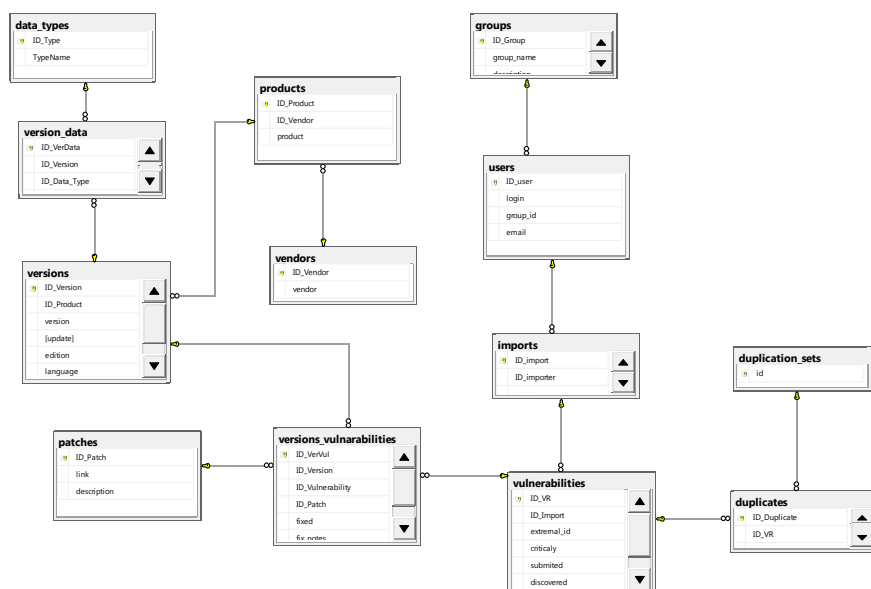


Рисунок 1. Схема бази даних для зберігання та керування загрозами

Ключовим є відношення «вразливості» («vulnerabilities»), яке включає в себе інформацію про уразливість, таку як ідентифікатор уразливості в розробленій базі даних, ідентифікатор уразливості в CVE або інших ресурсах (якщо такий є), опис уразливості на природній мові, рейтинг критичності, дату виявлення і дату появи інформації про уразливість у відкритому доступі, а також ідентифікатор користувача, який повідомив про знайдену уразливість, в також ідентифікатор імпорту, під час якого цю уразливість внесли в базу даних.

УДК 681.518.3

**Р. Ясінський, Г. Осухівська, А. Паламар**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АПАРАТНО-ПРОГРАМНА СИСТЕМА ДЛЯ РЕГУЛЮВАННЯ МІКРОКЛІМАТУ ТЕПЛИЦЬ**

UDC 681.518.3

**R. Yasinskyi, H. Osukhivska, A. Palamar**

## **HARDWARE AND SOFTWARE SYSTEM FOR GREENHOUSES MICROCLIMATE REGULATING**

На сьогоднішній день велика кількість фермерських господарств переходить до вирощування продукції у закритих приміщеннях. Тому тепличні комплекси розвиваються та набувають все більшого поширення. Процес вирощування рослин в теплицях характеризується суттєвими витратами на енергетичні ресурси, частка яких в структурі собівартості може становити 60–70%. Таким чином, показник енерговитрат в теплицях є важливим фактором з точки зору економічної доцільності виробництва рослинної продукції. Тому задача збільшення енергоефективності є актуальною для усіх тепличних комплексів [1].

Проблема підвищення енергоефективності пов'язана з контролем за витратою ресурсів, а також з моніторингом параметрів мікроклімату, на які ці ресурси мають прямий вплив. Для вирішення цієї проблеми в роботі запропоновано використати концепцію «розумних теплиць», яка являє собою автоматизовану систему, що дозволяє регулювати параметри мікроклімату. Автоматизація роботи теплиці передбачає моніторинг різних показників, а також керування параметрами мікроклімату з метою забезпечення росту та дозрівання рослин. До основних параметрів, які може контролювати «розумна теплиця», можна віднести: освітлення; водопостачання; температуру; циркуляцію повітря та вологість [2, 3].

Для ефективного регулювання параметрів мікроклімату теплиці усіма цими підсистемами потрібно керувати одночасно, об'єднавши їх у одну велику систему, котра дозволить оптимізувати їх роботу. Для цього було використано технології інтернету речей (IoT), що є сучасною концепцією, яка створена з метою інтеграції різних об'єктів за допомогою мережевої інфраструктури для обміну даними з використанням IP протоколу.

В запропонованій системі IoT дозволив автоматизувати вирощування сільськогосподарської продукції, збільшуючи ефективність і фінансові результати.

### **Література**

1. Дудник А. О. Методи побудови ресурсоефективних систем керування тепличними комплексами. Актуальні проблеми наук про життя: матеріали IV Міжнародної науково-практичної конференції молодих вчених. Київ. 2018. С. 58–59.
3. Шарапа О. В., Бердников А. Г. Модель системи управління технологічним процесом в тепличному агропромисловому комплексі. Вісник Харківського національного університету імені В.Н. Каразіна, серія «Математичне моделювання. Інформаційні технології. Автоматизовані системи управління». Вип. 47. 2020. С. 86–92.
4. Романов Д. В., Осухівська Г. М., Паламар А. М. Система управління зовнішнім освітленням на основі інтернету речей: збірник тез доповідей X Міжнародної науково-практичної конференції молодих учених та студентів «Актуальні задачі сучасних технологій». Т. 1. 2021. С. 117.

## **СЕКЦІЯ 4. ПРОГРАМНА ІНЖЕНЕРІЯ ТА МОДЕЛЮВАННЯ СКЛАДНИХ РОЗПОДІЛЕНИХ СИСТЕМ**

УДК 004.41

**А. Буй**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

### **ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ ВИРІШЕННЯ ПРОБЛЕМ ІЗ РЕАЛІЗАЦІЄЮ СІЛЬСЬКОГОСПОДАРСЬКОЇ ПРОДУКЦІЇ**

UDC 004.41

**A. Bui**

### **INFORMATION SYSTEM FOR SOLVING PROBLEMS WITH SALE OF AGRICULTURAL PRODUCTS**

Виникнення коронавірусу, стало випробуванням на міцність для економіки країни, особливо для агропромислового комплексу.

Становище в сфері дуже сильно погіршилось через проблеми з логістикою, збутом, панікою покупців та стрибками цін [3]. Це все зробило ринок більш конкурентним [2]. Також, це призвело до виділення та створення Кабінетом Міністрів України централізованого онлайн-сервісу, яким будуть користуватись ферми [1].

Таким чином популярність різних онлайн сервісів, які можуть задовільнити потреби агропромислового бізнесу різко зросли, і потреба в них теж. Як одне з рішень всіх описаних проблем пропонується веб платформа Market Place.

Ця платформа є ефективним рішенням, але вона має бути максимально зручною і простою для користувачів, а також має вирішувати всі проблеми, які були описані вище. Сервіс мусить містити два типи оголошень: на продаж і на купівлю. Платформа Market Place задовільняє ці вимоги. Такий підхід добре спрацює як для тих хто шукає товари на купівлю – так і для тих, хто хоче продати їх. Існуючі рішення не дуже добре покривають саме ринок агропромисловості, існує потреба саме в цій сфері.

Наприклад, фермер, який шукає добрива – може розмістити оголошення для покупки добрив і продовжити займатись важливими для його бізнесу справами, не витрачаючи час на пошуки постачальників. Також після успішного врожаю – він відразу може створити оголошення по продажу товарів, і потім вибрати найбільш актуальну для нього ціну.

Ця веб платформа є чудовим рішенням, яке вирішить проблеми фермерів і людей які теж займаються реалізацією агро продукції. Market Place, крім того, допоможе швидше оголутись після карантину [5] фермам і підприємствам, і допоможуть в рази покращити реалізація у майбутньому.

#### **Література**

1. Матеріали для обговорення Програми стимулювання економіки для подолання наслідків COVID-19: «Економічне відновлення». URL: <https://www.kmu.gov.ua/news/programma-stimulyuvannya-ekonomiki-dlya-podolannya-naslidkiv-covid-19-ekonomichne-vidnovlennya>.
2. Вплив COVID-19 та карантинних обмежень на економіку України. URL: <https://surl.li/gtnp>.
3. Цінові гойдалки: що і як подорожчає у квітні. URL: [https://enovosty.com/uk/news\\_economy-ukr/full/514-cenovye-kacheli-hto-i-kak-podorozhaet-v-aprele](https://enovosty.com/uk/news_economy-ukr/full/514-cenovye-kacheli-hto-i-kak-podorozhaet-v-aprele).
4. Вільний Ринок. URL: <https://www.prostir.ua/?news=vidkrytyj-rynok-startuje-onlajn-projekt-zapidtrymky-amu>.
5. Врожайний карантин: як фермери пережили період обмежень. URL: <https://mind.ua/publications/20211682-vrozhajnij-karantin-yak-fermeri-perezihili-period-obmezhen>.

УДК 004.41

**В. Волович, Б. Береженко, І. Боднарчук**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ЗАДАЧА ПРОЄКТУВАННЯ ПРОГРАМНОЇ АРХІТЕКТУРИ В ПРОЦЕСАХ ЗАБЕЗПЕЧЕННЯ ЯКОСТІ**

UDC 004.41

**V. Volovych, B. Berezhenko, I. Bodnarchuk**

## **THE PROBLEM OF SOFTWARE ARCHITECTURE DESIGN IN THE PROCESSES OF QUALITY ASSURANCE**

При проектуванні програмних систем (ПС) широко застосовується компонентна технологія, яка базується на використанні компонентів, взятих з раніше виконуваних проєктів (компоненти повторного використання).

Архітектура в цій технології проєктується шляхом вибору каркасу, на основі вимог до ПС, і заповненням його необхідними компонентами, взятими з репозиторію. Оскільки існує декілька компонентів, які реалізують одну і ту ж функціональність, то отримуємо множину альтернативних архітектур ПС. Для вибору найбільш прийняттого рішення необхідно або проранжувати альтернативи відносно значень критеріїв якості ПС, або використати деякий інтегральний критерій, значення якого обчислюється для кожної альтернативи.

На практиці використовується декілька методів оцінювання якості програмної архітектури. Найбільш відомими з них є методи, які базуються на розробці сценаріїв використання та перевірки, чи задовольняє даний варіант архітектури вимозі по певному критерію якості. Найбільш відомими з них є методи SAAM, АТАМ і СВАМ [1].

Спільним недоліком перерахованих методів є те, що для їх реалізації необхідно створювати та аналізувати експертам достатньо велику кількість сценаріїв використання, що робить їх трудомісткими, вартісними і складними для формалізації. Тому поява робіт, в яких для рішення цієї задачі було використано метод аналізу ієрархій (МАІ), дозволила значно покращити процедуру вибору архітектури і формалізувати її для автоматизації процесів прийняття рішень [2].

Суттєвим недоліком застосування МАІ є обмежена кількість альтернатив, які можна оцінювати одночасно ( $n \leq 7 \pm 2$ ), що породжується неузгодженістю елементів матриць парних порівнянь, яка збільшується при збільшенні кількості альтернатив [3].

Для вирішення цієї проблеми О. Павловим в [4] запропонована модифікація МАІ, в якій вагові множники альтернатив визначаються з умови мінімізації неузгодженості матриці парних порівнянь, що приводить вихідну задачу до задачі математичного програмування. В роботах [5], [6] розглянуті питання застосування модифікованого МАІ (ММАІ) до задачі оцінювання альтернативних варіантів архітектури програмних систем при великій кількості альтернатив.

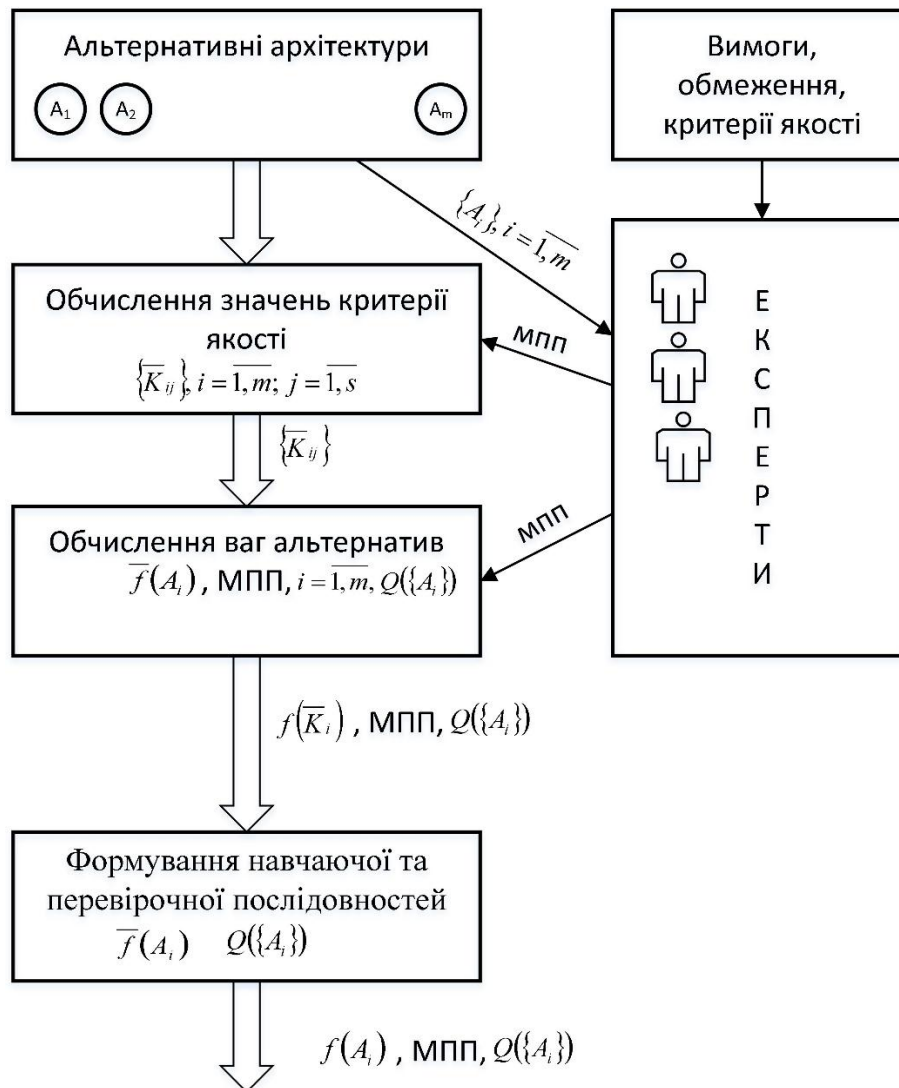
Остаточний вибір варіанта архітектури з врахуванням сукупності критеріїв частіше всього виконується заміною задачі багатокритерійного вибору архітектури однокритерійною, з використанням лінійної адитивної згортки частинних критеріїв. Її використання є обґрунтованим лише у невеликому околі базових точок.

Також проблематичним є призначення ваг частинних критеріїв у скалярній згортці експертним методом який є неформалізованим, носить суб'єктивний характер і служить додатковим джерелом помилок. Для вирішення вказаних проблем необхідно вибрати прийнятну структуру скалярної згортки і застосувати формалізовані методи визначення ваг частинних критеріїв.

Для цього пропонується використовувати універсальну скалярну згортку запроповану в [7]. В ній оптимізується цільова функція, яка залежить від міри напруженості ситуації, котра визначається близькістю значень критеріїв до своїх обмежень. Схема задачі оцінювання та багатокритеріального вибору архітектури ПС з множини альтернатив зображено на рис. 1.



Тут використовуються такі позначення:  $K_j^1, j = \overline{1, p}$  критерії якості ПС, визначені у відповідності з вимогами в термінах стандарту з якості ISO/IEC 25010;  $K_i^2, i = \overline{1, n}$  – критерії якості архітектури, визначені з множини  $K_j^1, j = \overline{1, m}$  методом SQFD (Software Quality Function Deployment) або методом парних порівнянь [5];  $K^0$  – інтегральний критерій якості ПС;  $R_i, i = \overline{1, n}$  – задані обмеження на показники якості архітектури;  $A_i, i = \overline{1, m}$  – альтернативні архітектурні рішення. Оскільки множина критеріїв  $\{K_i^2\}$  отримана з множини  $\{K_j^1\}$ , то можна рівень показників якості ПС виключити з розгляду.



**Рисунок 1.** Структурна схема процедури вибору програмної архітектури

Отримати порівняльні оцінки альтернатив  $\{A_i\}$  по кожному критерію  $K_i^2, i = \overline{1, n}$  можна методом аналізу ієрархій Сааті (МАІ), або модифікованим методом (ММАІ), використання яких детально описано в роботах [2], [5]. Відмінність ММАІ від МАІ полягає в тому, що в ньому оцінки альтернатив по реалізації критеріїв якості шукаються з умови мінімуму міри неузгодженості матриці парних порівнянь, що дозволило розширити межі застосування методу на більшу кількість альтернатив (критеріїв) ( $n \leq 30$ ) [6].

При обчисленні інтегрального критерія якості альтернатив з використанням скалярної згортки ваги критеріїв визначають експертним методом.

В експертному оцінюванні, як правило, беруть участь декілька груп фахівців, які мають різне бачення стосовно впливу кожного з атрибутів якості на загальну якість ПА. Для підвищення достовірності результуючої оцінки і досягнення компромісу призначаються показники компетентності кожній з груп  $(\beta_1, \beta_2, \dots, \beta_r)$ ,  $\sum_{i=1}^r \beta_i = 1$ ,  $\beta_i \geq 0$ .

Потім кожна з груп формує матриці парних порівнянь для критеріїв якості і застосуванням МАІ обчислюються ваги критеріїв  $\{\alpha_i^s\}, i=1, \dots, n, s=1, \dots, r$  – номер групи експертів. Компромісне рішення можна знаходити, як середнє геометричне  $\alpha_i = \sqrt[r]{\alpha_i^1 \cdot \alpha_i^2 \cdot \dots \cdot \alpha_i^r}$ , або як усереднене, з врахуванням показника компетентності груп експертів  $\alpha_i = \alpha_{i1}^{\beta_1} \alpha_{i2}^{\beta_2} \cdot \dots \cdot \alpha_{ir}^{\beta_r}$ ,  $i=1, \dots, n$ . Однак, при великих розбіжностях в оцінках, таке усереднення може не привести до компромісу інтересів. Так, як слідує з даних, взятих з [6], отримані за результатами опитування різних груп фахівців, відрізняються більше, ніж в два рази.

В цьому випадку, використання усереднених значень оцінок ваг критеріїв може не забезпечити компроміс інтересів експертів. Тоді використання лінійної згортки для оцінки альтернатив ПА і вибору найкращої з них може бути некоректним. Також при виборі ПА потрібно враховувати можливість зміни вимог до ПА в процесі проектування, а відповідно і ваг критеріїв якості.

### Література

1. Len Bass, Paul Clements, Rick Kazman Software architecture in practice. 2nd ed., Addison-Wesley, 2003, 575 p.
2. M. Svahnberg, C. Wholin, and L. Lundberg. A Quality-Driven Decision-Support Method for Identifying Software Architecture Candidates. Int. Journal of Software Engineering and Knowledge Engineering. 2003. 13 (5). P. 547–573.
3. Saaty T., Vargas L. Decision Making with the Analytic Network Process. N.Y.: Springer, 2006. 278 p.
4. Павлов А. А., Лищук Е. И., Кут В. И. Математические модели оптимизации для нахождения весов объектов в методе парных сравнений. Системні дослідження та інформаційні технології. К.: ИПСА, 2007. № 2. С. 13–21.
6. Харченко О. Г., Боднарчук І. О., Галай І. О. Метод багатокритеріальної оптимізації програмної архітектури на основі аналізу компромісів. Інженерія програмного забезпечення. К.: НАУ, 2012. № 3–4 (11–12). С. 5–11.
7. Kharchenko A., Bodnarchuk I., Yatsyshyn V. The method for comparative evaluation of software architecture with accounting of trade-offs. American Journal of Information Systems. Vol. 2. No. 1. 2014. P. 20–25. URL: <http://pubs.sciepub.com/ajis/2/1/5>.
8. Воронин А. Н., Зиятдинов Ю. К. Теория и практика многокритериальных решений: Модели, методы, реализация. LambertAcademicPublishing, 2013, 305 p.

УДК 004.41

**С. Глинянчук, І. Стадник**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **РОЗРОБКА WEB-ДОДАТКУ ДЛЯ РЕКОМЕНДАЦІЇ ІГОР**

UDC 004.41

**S. Hlynianchuk, I. Stadnyk**

### **DEVELOPING WEB-APPLICATION FOR GAME RECOMMENDATION**

Кількість інформації, доступної в Інтернеті, неухильно зростає: щодня в мережі з'являється безліч нових ресурсів, а вже існуючі сайти завантажують ще більше нових даних.

Зміст цієї інформації може бути будь-яким. Наприклад, нам пропонують художні книги, фільми, відеоігри, новини, наукові статті, товари, і дуже часто знайти потрібні дані важко. Це залежить не тільки від неймовірного обсягу інформації, а й від необізнаності її споживачів.

Фільтрування цих даних здається логічною необхідністю для користувачів мережі Інтернет. Такі великі платформи, як Amazon, Netflix, YouTube та Steam мають свої системи для отримання персоналізованого вмісту.

Ці системи називаються системами рекомендацій. Вище наведені платформи використовують їх для обслуговування потреб своїх споживачів, надаючи персоналізований вміст з метою покращення взаємодії з користувачами, таким чином підвищуючи лояльність клієнтів. Це допомагає їм збільшити продажі своїх продуктів і підвищити доходи компаній.

Системи рекомендацій необхідні в сучасному Інтернеті. Тому існує багато систем для різних типів інформації. Наприклад, web-сайти для оцінки фільмів містять дані про безліч фільмів та телесеріалів, а також дають можливість користувачам проголосувати за свої улюблені фільми і навпаки, проголосувати проти нелюблених. Існує і багато ресурсів на музичну тематику, які будують свою систему рекомендацій на основі прослуховувань клієнтів.

Проте, для користувачів, що цікавляться відеоіграми, основними джерелами інформації завжди були тематичні гілки на форумах та відгуки в інтернет-магазинах. Зважаючи на те, якою популярною стала ця індустрія і як багато ігор існує зараз, виникла необхідність створити відповідний ресурс. Така платформа може допомогти гравцям у відеоігри по всьому світу вибирати їх на основі власних вподобань, а також відкинути неприйнятний для них вміст.

#### **Література**

1. Adomavicius, Gediminas and Alexander Tuzhilin (2005). «Toward the Next Generation of Recommender Systems: A Survey of the State-of-the-Art and Possible Extensions.» In: IEEE Transactions on Knowledge and Data Engineering 17.6. P. 734–749.

УДК 004.41

**В. Гречаник**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **СИСТЕМАТИЗАЦІЯ ЛОГІСТИЧНИХ ПОСЛУГ**

UDC 004.41

**V. Hrechanyk**

## **SYSTEMATIZATION OF LOGISTICS SERVICES**

**Ключові слова:** логістика, перевезення, вантаж, попутні перевезення.

**Key words:** logistics, transportation, cargo, transit transportation.

Сьогодні підприємство підвищує свою конкурентоспроможність не тільки за рахунок капіталомісткого освоєння випуску нового товару, а в результаті поліпшення якості значимих для клієнта характеристик поставки.

Як нам відомо, існують перевізники та клієнти для перевезень. А також клієнти для попутних перевезень у разі неповного завантаження транспортного засобу. Добре було б їх подружити. Найкращий спосіб – це сайт – з оголошеннями перевізників та оголошеннями замовників. Оголошення бачать обидві сторони, надсилають одна одній свої пропозиції. Аналог відомого сервісу для перевезень «BlaBlaCar» тільки у вантажній галузі.

Оголошення, звичайно, проходять модерацію, внесення комісії за розміщення, що обумовлює економічну частину з боку адміністратора сайту.

Економічний ефект з погляду клієнтів – зменшення вартості доставки у разі попутних рейсів.

Результати дослідження мають практичне значення і можуть бути корисними науковцям, зацікавленим у вивченні логістичного аутсорсингу, а також підприємствам, де аутсорсинг використовується найчастіше.

### **Література**

1. Кальченко А. Г. Формування систем логістики. 2003 р. 43 с.

УДК 004.41

**І. Грибун**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АНАЛІЗ ТА МОДИФІКАЦІЯ БАЙТ-КОДУ СКОМПІЛЬОВАНОГО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

UDC 004.41

**I. Hrybun**

## **ANALYSIS AND MODIFICATION OF BYTECODE IN COMPILED SOFTWARE**

Модифікація існуючого програмного забезпечення є перспективним напрямком дослідження в сучасній сфері розробки програмних продуктів. Великий обсяг створюваного програмного забезпечення та мінливість вимог користувачів іноді призводить до ситуацій, у яких потреби клієнтів змінюються після завершення розробки. Зазвичай, клієнт може легко зробити запит до розробників щодо внесення змін. Однак, існують ситуації у яких це є або неможливим, або небажаним. У таких випадках найняття спеціаліста для внесення мінімальних змін є допустимим варіантом.

Поширення багатоплатформених та швидкодійних програмних технологій, наприклад .NET Framework або Mono, залишається досить суттєвим [1]. Оскільки вони використовують компіляцію до проміжного байт-коду – їх модифікація потребує спеціалізованих знань щодо аналізу та зміни їх алгоритмів.

Метою даного дослідження є створення загальної методології пошуку та аналізу коду, відповідального за функціонал існуючого програмного продукту, та використання методів зміни даного коду відповідно до потреб клієнта. Належне застосування цих методів у поєднанні з усвідомленням загальноновживаних структур та парадигм програмування дозволяє набагато швидше орієнтуватись у коді великих проєктів. Також важливим є використання ефективних інструментів відповідно до поставленої задачі: програмне забезпечення ILSpy є висококласним продуктом, ціллю якого є детальний аналіз байт-коду Common Intermediate Language, який використовується платформою .NET Framework [2].

Також доцільним є порівняння даної методології та систем порівняння математичних та програмних моделей [3]. Деякі методи використані у процесі аналізу цих моделей можуть бути перепрофільовані для аналізу структур скомпільованого байт-коду.

### **Література**

1. Most used libraries and frameworks among developers, worldwide, as of 2022. URL: <https://www.statista.com/statistics/793840/worldwide-developer-survey-most-used-frameworks/>.
2. ILSpy: .NET Decompiler with support for PDB generation, ReadyToRun, Metadata (&more). URL: <https://github.com/icsharpcode/ILSpy>.
3. Petryk M., Mykhalyk D., Petryk O. Comparison of solving algorithms for a mathematical model of filtration-diffusion transfer in the medium of spherical moisture-saturated microporous particles. Scientific Journal of TNTU. Tern.: TNTU, 2021. Vol. 101. No. 1. P. 15–21.

УДК 004.41

**О. Гузеляк, Ю. Шевчук, Б. Береженко, І. Боднарчук**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ПРОГРАМНА АРХІТЕКТУРА В РОЗПОДІЛЕНИХ КОМАНДАХ ГНУЧКИХ ПРОЄКТІВ**

UDC 004.41

**O. Huzeliak, Yu. Shevchuk, B. Berezhenko, I. Bodnarchuk**

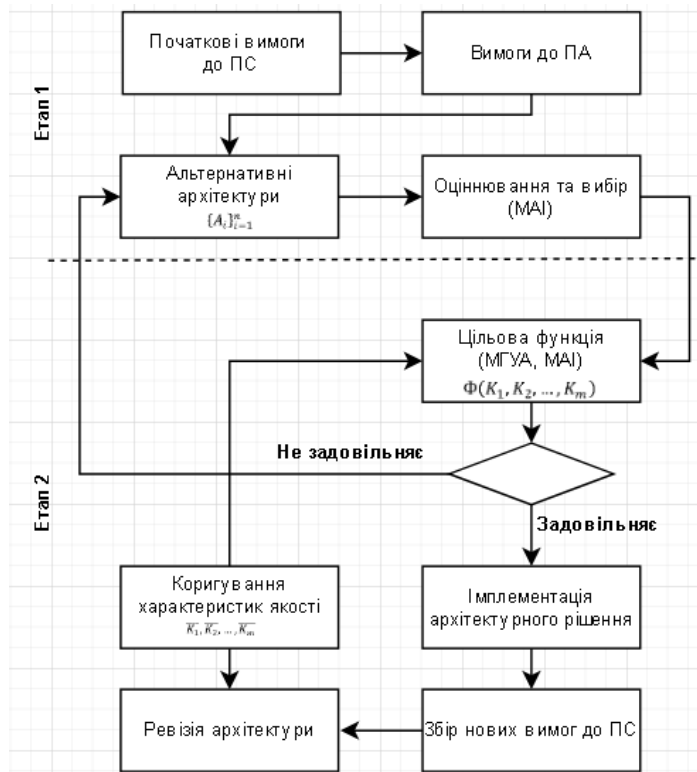
## **SOFTWARE ARCHITECTURE DESIGN IN DISTRIBUTED TEAMS OF AGILE PROJECTS**

Базуючись на принципах Agile Маніфесту [1] практики Agile розглядають первинну розробку програмної архітектури (ПА), як витрати ресурсів, які можуть не окупитись. В Agile традиційно приділяли основну увагу при плануванні ітерацій потребам замовника і сценаріям використання (так званим user stories), залишаючи поза увагою архітектурне проектування.

Однак, при застосуванні Agile до великих проєктів ці погляди змінились. Так в роботі [2] показано, що ігнорування архітектурного проектування може спричинити неоптимальні рішення, а в роботі [3] автор стверджував, що невідповідність розробки великих проєктів архітектурному проєкту може привести до їх провалу. В роботах [3], [4] підкреслена ключова роль архітектурного проєкту при застосуванні Agile до розробки великих і складних проєктів, а також фактичну неможливість їх виконання без використання архітектури. Це відбувається через втрату зв'язку беклогів (специфікацій вимог) з усією системою, що робить неможливим передбачити реальні наслідки своїх рішень при плануванні кожної ітерації.

Для вирішення цих проблем були впроваджені в ітерації гнучких методів елементи архітектурного проектування та почали реалізовувати зв'язок архітектурного проєкту всієї програмної системи з проєктами локальних команд [6]. При реалізації великих проєктів в рамках гнучких методів створювалась «координуюча» команда, яка вела архітектурний проєкт всієї програмної системи і здійснювала координацію внесення змін в архітектуру компонентів, які розробляли локальні команди [6]. Для цього були введені нові ролі в координуючій команді. Це архітектор системи, архітектор бізнес-процесів та інші. Процеси архітектурного проектування в локальній команді може вести або архітектор, або один з членів команди, суміщаючи ці обов'язки з роллю розробника чи іншою роллю на проєкті [7]. Для забезпечення ефективності реалізації взаємодії процесів на локальному і глобальному рівнях необхідно було розробити відповідні моделі цих процесів та створити CASE-засіб, який би автоматизував процеси створення та обміну архітектурною інформацією. В [8] було запропоновано двоетапну схему такої взаємодії (див. рис. 1). На першому етапі створюються альтернативні варіанти архітектури з врахуванням вимог за технологією, яка залежить від прийнятого стилю документування. Для оцінювання альтернатив обчислюються їх відносні оцінки по кожному з критеріїв якості з використанням модифікованого методу аналізу ієрархій (MAI). Вибір кращого варіанта архітектури виконується методом аналізу компромісів або на основі значення інтегрального показника якості [9].

На другому етапі вибраний варіант архітектури впроваджується в гнучкому методі розробки. В разі виявлення нових вимог до програмної системи (ПС) або зміни існуючих вносяться відповідні зміни в архітектуру. Зміни виконуються шляхом корегування коду відповідного патерну (шаблону) проектування або його заміною на альтернативний. Для дослідження та оптимізації впливу цих змін на якість проводиться корекція значень критеріїв якості. Процедура корекції розроблена на основі застосування методу «заміщення – компенсації», в якому збільшення деяких критеріїв якості відбувається за рахунок зменшення інших, що дозволяє оптимізувати ці зміни і не виходити за межі бюджету проєкту [10].



**Рисунок 1.** Схема проектування архітектури в розподілених командах для гнучких проектів

Перший і другий етапи об'єднуються процедурою обчислення цільової функції якості архітектури. Цільова функція визначається методом групового урахування аргументів в поєднанні з МАІ [8]. Запропонована процедура дозволить, крім координації процесів, також забезпечити якість проекту. Для адаптації процесів архітектурного проектування до вимог Agile необхідно максимально зменшити час на процеси архітектурного проектування. Для цього пропонується створити систему, яка б могла підтримувати спільне використання командами архітектурної інформації. Ці процедури повинні виконуватись командами в ітераціях Agile, а також координуючою командою для архітектури всієї системи. Архітектурний проект приймається і створюється на фазі аналізу вимог перед спринтами розробки. Проте в ході розробки може виникнути потреба змінювати архітектуру, і командою SCRUM або архітектором приймаються певні нові рішення. Основною задачею тут є оперативне оцінювання можливих архітектурних рішень та вибору найбільш прийнятної для задоволення вимог всіх зацікавлених сторін. Тут може бути використаний МАІ, а для коригування архітектури в ітераціях той же МАІ або метод групового урахування аргументів (МГУА), які набагато ефективніші ніж метод сценаріїв АТАМ, що розглядався в [14]. Також необхідно автоматизувати процедури формування альтернативних архітектур на основі архітектурно значимих вимог та при внесенні змін в програмний код і здійсненні реінжинірингу. Ці процедури, а також метод оцінювання МАІ та процедури зберігання архітектурної інформації реалізовані в засобі «Архітектор» [15], [16]. Тому систему можна будувати на його основі.

Отже, хоча Agile підхід має багато переваг він все ж таки не є універсальним рішенням, і компанії часто використовують поєднання гнучких та планово орієнтованих методів, що називається гібридним підходом. Він є результатом компромісу використання переваг ітеративного, логічного та спільного підходів із підтримкою певного рівня планування та структури. При цьому гібридний підхід має ряд переваг, таких як: належна підтримка бізнес-проектів, допомагає досягти хорошої якості продукції, адекватно відповідає зовнішнім вимогам (наприклад, стандартам) та допомагає пришвидшити розробку й скоротити час виходу продукту на ринок, а також дає можливість постійно вдосконалювати якість завдяки гібридним конструкціям, покращити задоволеність команди. Гібридний підхід має також

недоліки: проблеми забезпечення продуктивності проекту та затримки релізів. Мінімізацію недоліків реалізують застосуванням ширшого впровадження архітектурного проектування в Agile, а також застосуванням засобів автоматизації цих процесів.

### Література

1. Agile alliance: manifesto for agile software development. URL: [http:// agilemanifesto.org](http://agilemanifesto.org) [retrieved: 10.10.2022]
2. Dybå, T., & Dingsøyr, T. (2008). Empirical studies of agile software development: A systematic review. *Information and software technology*, 50 (9–10), 833–859.
3. Bowers, J., May, J., Melander, E., Baarman, M., & Ayoob, A. (2002, August). Tailoring XP for large system mission critical software development. In *Conference on Extreme Programming and Agile Methods* (pp. 100–111). Springer, Berlin, Heidelberg.
4. Ghanam Y. Andreichuk D., Maurer F. Reactive variability management in agile Software development. In: *Agile conference*. Orlando, FL: Computer Society; 2010, p. 27–34.
5. Abrahamsson, P., Babar, M. A., & Kruchten, P. (2010). Agility and architecture: Can they coexist? *IEEE Software*, 27 (2), 16–22.
6. Dingsøyr, T., Moe, N. B., & Seim, E. A. (2018). Coordinating knowledge work in multiteam programs: findings from a large-scale agile development program. *Project Management Journal*, 49 (6), 64–77.
7. Eloranta, V. P., & Koskimies, K. (2013). Software architecture practices in Agile enterprises. In *Aligning Enterprise, System, and Software Architectures* (p. 230–249). IGI Global.
8. Bodnarchuk, I., Lisovyi, V., Kharchenko, O., & Galai, I. (2018, September). Adaptive Method for Assessment and Selection of Software Architecture in Flexible Techniques of Design. In *2018 IEEE 13th International Scientific and Technical Conference on Computer Sciences and Information Technologies (CSIT)* (Vol. 1, p. 292–297). IEEE.
9. Kharchenko, A., Bodnarchuk, I., Halay, I., & Yatcyshyn, V. (2016). An Optimal Trade-off Solution of the Software Architecture Choice Problem. *Journal of Information and Computing Science*. 11 (4). P. 281–290.
10. I. Bodnarchuk, et al. «Multicriteria choice of software architecture using dynamic correction of quality attributes.» *International Conference on Computer Science, Engineering and Education Applications*. Springer, Cham, 2019, p. 419–427.
11. Galster M., & Avgeriou P. (2014). Supporting variability through agility to achieve adaptable architectures. In *Agile Software Architecture* (p. 139–159). Morgan Kaufmann.
12. Kharchenko A., Halay I., & Bodnarchuk I. (2016, September). Multicriteria architecture choice of software system under design and reengineering. In *2016 XIth International Scientific and Technical Conference Computer Sciences and Information Technologies (CSIT)* (p. 4–8). IEEE.
13. Харченко О. Г., Боднарчук, І. О., Райчев І. Е., & Галай І. О. (2015). Інструментальний засіб порівняльного оцінювання і багатокритеріального вибору архітектури програмних систем. *Інженерія програмного забезпечення*. Київ: НАУ, 2015. № 1. С. 10–24.



УДК 004.4

**О. Дзюма, І. Мудрик**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ДОСЛІДЖЕННЯ СИСТЕМ ТЕСТУВАННЯ НА ОСНОВІ РОЗРОБЛЕНОГО ІНТЕРНЕТ СЕРВІСУ ПОТОКОВОГО АУДІО**

UDC 004.4

**O. Dziuma, I. Mudryk**

## **RESEARCH OF TESTING SYSTEMS BASED ON A DEVELOPED INTERNET AUDIO STREAMING SERVICE**

Інтернет сервіс потокового аудіо – це одностороння онлайн-аудіоплатформа, яка експортує через мережу даних, наприклад аудіотреки, аудіокниги. На ПК, планшетах, смартфонах, переносних пристроях ми використовуємо його для прослуховування музики і радіопередач з Інтернету. Головна відмінність потокового аудіо – відсутність необхідності зберігати файли на жорсткому диску.

З початком пандемії попит на потокові послуги зріс. І це стосується не тільки музики. Маючи багато вільного часу, але брак можливостей для особистих зустрічей, багато людей перейшли на різні потокові програми.[1]

Існують різні типи рішень потокового аудіо, наприклад потокове аудіо за запитом або пряма трансляція. В першому варіанті потоки обробляються таким чином, щоб користувач в любий проміжок часу, з любої точки місцезнаходження зміг прослухати вибрану композицію. Онлайн-трансляція покращується завдяки доступності для запуску, зупинки або зміни на наступний список відтворення. Деякі з прикладів постачальників послуг потокового передавання музики за запитом: Apple Music, SoundCloud, Spotify тощо. В другому варіанті, аудіотрансляція в прямому ефірі пропонує контент, який транслюється в режимі реального часу. Деякі з найпопулярніших платформ потокового аудіо в прямому ефірі: BBC Live Audio, Capital FM, Radio X тощо [2].

Розроблена платформа забезпечує потокове аудіо за запитом, тому повинна містити музичний плеєр з основними функціями та виглядом: назва відтвореної композиції та виконавець, обкладинка композиції або альбому, кнопки керування (пауза, попередній/наступний звук, гучність), список відтворення з наступними піснями, поставити вподобання і додання музичних треків в список відтворення. Також не менш важливим функціоналом являється можливість створення облікового запису.

Оскільки, система буде опрацьовувати великий об'єм даних, то для надійної та стабільної роботи платформа розділена на декілька незалежних частин до бізнес логіки. Під час розробки сервісу для серверної частини використовувалась мова програмування PHP 8, зумовлено це тим, що доволі легка сумісність з інтеграціями фреймворків для тестування, наприклад PHPUnit, що дозволяє якісно провести різні види тестування платформи.

### **Література**

1. Contus. Contus. URL: <https://blog.contus.com/>.
2. Stormotion. Stormotion. URL: <https://stormotion.io/>.

УДК 004.41

**Н. Доскоч, Г. Цуприк**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **РОЗРОБКА СИСТЕМИ ЗАБЕЗПЕЧЕННЯ БЕЗПЕЧНОГО ОБМІНУ ІНФОРМАЦІЄЮ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ ВЕБ ПРОГРАМУВАННЯ**

UDC 004.41

**N. Doskoch, H. Tsupryk**

## **DEVELOPMENT OF SAFE INFORMATION EXCHANGE SYSTEM USING WEB PROGRAMMING TECHNOLOGIES**

Актуальність обраної теми зумовлена всім відомим давнім, але не на хвилину не втрачаючим своєї сили висловом «Хто володіє інформацією, той володіє Світом». Цікаво, що кожним поколінням він сприймається по своєму, враховуючи стан науки і техніки на конкретний період часу, проте це нічого не міняє. Інформація – це можливість вдосконалюватись, розвиватись і рухатись вперед, це товар, який може принести як казкові статки і та і спрямувати до повного фіаско, в один момент винести на олімп слави і з силою грюкнути об землю. Від моменту, коли хтось зрозумів, що інформація, це такий самий товар як і продукти харчування, промислові, корисні копалини в цілому чи цінні метали тощо, стало зрозуміло і те, що її потрібно «захищати» від зовнішніх впливів та втручань.

Сучасний стан інформаційних технологій, науки і техніки дає можливість забезпечувати ефективний захист інформації на різних рівнях, але, час від часу, з'являється щось нове, невідоме, і проблема вмить стає гострою та потребує нагального вирішення.

Метою роботи є представлення програмної реалізації, яка забезпечить можливість ефективного та безпечного обміну інформацією, а також реалізація базових функцій обробки інформації для обраних напрямку та тематики.

Об'єктом дослідження визначено власне явище, що породжує проблемну ситуацію і яке я й обрав для вивчення, а саме: це створена програмної реалізації, яка дозволить забезпечити повноцінний, ефективний та, головне, безпечний обмін інформацією, та реалізувати базові функції обробки інформації для обраної сфери.

Предмет дослідження міститься в межах об'єкту і власне й визначає тему роботи, а основні проблеми предметної області, дають змогу окреслити задачі, котрі потребують вирішення, зокрема задовольнити потребу в швидкому скануванні, вирішити питання потреби значних ресурсів системи, врахувати можливі помилкові спрацювання чи не спрацювання системи та зробити таку систему, яка була б ефективною та доступно широкому колу користувачів.

В роботі взято до уваги та застосовано одну із методик проектування архітектури системи із модульною структурою та при застосуванні сучасних технологій web-програмування. В представленій архітектурі сервісу враховано недоліки найпростішої існуючої архітектури, такі як складність горизонтального масштабування та низька швидкодія, проте недоліком все ж вважаю overhead від віртуальних машин. Методи дослідження, тобто розробки базуються на технологіях Java, Spring Framework – для API сервера, Python – для воркерів (процеси, які виконують завдання), для збереження даних – MySQL, для нотифікацій воркерам і синхронізації запитів до бази даних в деяких випадках NoSQL сервері Redis, JPA (Java Persistence API) та Hibernate ORM для роботи з базою даних, WebSockets – для нотифікацій користувачам сервісу і AngularJS, SockJS для розробки.

### **Література**

1. Погорілий С. Д. Програмне конструювання; за редакцією академіка АПН України Третяка О.В. ВПЦ Київський університет. 2005.
2. Петрик М. Р., Михалик Д. М., Петрик О. Ю., Цуприк Г. Б. Методичні вказівки до виконання атестаційної роботи магістра за спеціальністю 121 «Інженерія програмного забезпечення» для усіх форм навчання. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя. 2020. 27 с.

УДК 004.4

**О. Кишкевич, А. Кашосі, Д. Михалик**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ПРОЕКТУВАННЯ ТА РОЗРОБКА ПЛАТФОРМИ КЕРУВАННЯ МАРКЕТИНГОВИМИ ДАНИМИ НА ОСНОВІ AIRFLOW ТА HADOOP**

UDC 004.4

**O. Kyshkevych, A. Kashosi D. Mykhalyk**

## **DESIGN AND DEVELOPMENT OF MARKETING DATA MANAGEMENT PLATFORM BASED ON AIRFLOW AND HADOOP**

На даний момент важко знайти приклад бізнесу, який би був успішним без використання маркетингу. Маркетинг на пряму впливає на конкурентоспроможність, а щоб бути конкурентоспроможним потрібно вміло працювати зі своєю аудиторією. Також важливим є той факт, що від розміру і типу аудиторії залежать прибутки самої компанії. Аудиторію або ж користувачів потрібно не тільки привабити і переконати їх скористатися послугою, а ще й утримати. Таким чином ми бачимо що маркетинг завдяки роботі з аудиторією суттєво впливає на бізнес [1].

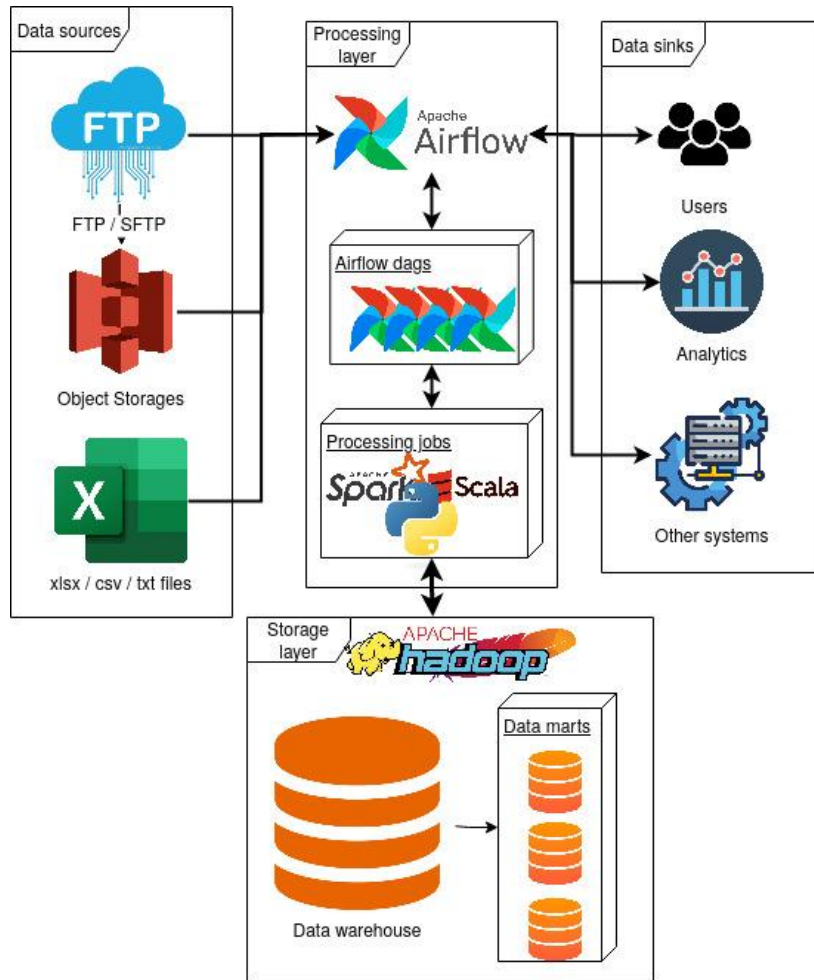
Завдання маркетолога не є простим, в його обов'язки входить: пошук нових користувачів, збір даних про користувача, розподіл користувачів по категоріях, створення пропозицій, фільтрування груп і власне саме проведення маркетингової кампанії [2]. Багато чого з цих завдань потрібно розробити автоматизоване рішення, яке пришвидшить і покращить роботу маркетолога.

Розробка такої платформи не є простим завданням. Дана система складається з багатьох бізнес процесів, в кожного з яких власні вимоги до об'ємів та швидкостей роботи з даними. Важливим критерієм платформи є можливість надавати сервіс для різних компаній. Дані компанії можуть обмінюватися власними даними, в результаті чого створювати більш точний профіль користувача що покращить маркетинг.

Дана платформа проектувалась для роботи з великими об'ємами даних, з різних джерел. Саме тому при проектуванні платформи були обрані технології які здатні працювати з великими об'ємами даних ефективно. В даному випадку група технологій Hadoop призначена для опрацювання та зберігання великих об'ємів маркетингових даних [3]. А Airflow використовується для керування потоками даних в платформі [4].

Дана платформа надає можливості по керуванню даними для маркетингових компаній. Маркетингова компанія яка хоче скористатися послугами платформи, може завантажити власні дані, або отримати доступ до даних інших компаній. Збір даних може відбуватися з різних джерел. Після чого дані обробляються, фільтруються дублікати та невірні записи, і приводяться до однієї уніфікованої структури, після чого зберігаються в сховище даних певним чином, щоб пізніше було зручно швидко отримувати ці дані [5].

Дана платформа поділена на незалежні частини у відповідності до бізнес логіки на такі ланцюги даних: дані по користувачам, дані з відписок, дані з блок листів, дані з підписок, дані з розсилок.



**Рисунок 1.** Загальна діаграма архітектури платформи

Запропоноване архітектурне рішення (рис. 1) було реалізовано та на практиці показало свою ефективність. Успішно пройшла перевірку на масштабованість в якій збільшився об'єм даних від 1 TiB до 20 TiB.

### Література

1. Asia R. Lockett, Online Marketing Strategies for Increasing Sales Revenues of Small Retail Businesses, Walden University, 2018. 148 p.
2. Donald Miller, Building a StoryBrand: Clarify Your Message So Customers Will Listen, ISBN-13: 978-0718033323
3. Hadoop. URL: <https://hadoop.apache.org/>.
4. Airflow. URL: <https://airflow.apache.org/>.
5. Data Warehouse. URL: <https://www.kimballgroup.com/>.

УДК 004.41

**А. Коваль, М. Петрик**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **МІКРОСЕРВІСНА АРХІТЕКТУРА В РОЗРОБЦІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

UDC 004.41

**A. Koval, M. Petryk**

## **MICROSERVICE ARCHITECTURE IN SOFTWARE DEVELOPMENT**

Мікросервісна архітектура є відносно новим рішенням для побудови програмних продуктів. Вона складається з набору невеликих автономних сервісів, кожен з яких може реалізовувати окрему бізнес модель. Мікросервіси невеликі, незалежні та слабо пов'язані. Кожна служба є окремою кодовою базою, якою може керувати невелика команда розробників. Сервіси відповідають за збереження власних даних або зовнішнього стану. Це відрізняється від традиційної моделі, де окремий рівень даних обробляє збереження даних.

Сервіси взаємодіють одне з одним за допомогою чітко визначених API. Деталі внутрішньої реалізації кожної служби приховані від інших сервісів. Також вони підтримують поліглотне програмування і не потребують спільного використання одного стеку технологій, бібліотек або фреймворків.

Таке архітектурне рішення має очевидні переваги. Так, оскільки мікросервіси розгортаються незалежно, керувати виправленнями помилок і випусками функцій легше. Розробники можуть вибрати технологію, яка найкраще підходить для їхніх послуг, використовуючи відповідну комбінацію стеків технологій. Якщо сервіс стає недоступним, це не призведе до переривання роботи всієї програми. Сервіси можна масштабувати незалежно, дозволяючи вам масштабувати підсистеми, які вимагають більше ресурсів, без масштабування всієї програми. Також вони ізолюють дані.

Також мікросервісна архітектура має ряд недоліків, які є важливими факторами, для вибору її як основу для проектів. Кожний мікросервіс простий, але вся система в цілому складніша. Написання невеликого сервісу, який покладається на інші залежні служби, вимагає іншого підходу, ніж написання традиційної монолітної або багаторівневої програми. Також складно тестувати залежності сервіси, особливо коли проектування у вас можуть виникнути проблеми із зворотною чи прямою сумі програма швидко розвивається. Може відбутися перевантаження мережі та затримка, через використання великої кількості сервісів. Кожний мікросервіс відповідальний за власне збереження даних. У результаті узгодженість даних може бути проблемою, як і узгодженість версій проекту.

### **Література**

1. Мікросервісна архітектура. URL: [learn.microsoft.com/en-us/azure/architecture/guide/architecture-styles/microservices](https://learn.microsoft.com/en-us/azure/architecture/guide/architecture-styles/microservices).

УДК 004

**Р. Ковальчук**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АДАПТАЦІЯ ТЕХНОЛОГІЙ КОНТРОЛЮ ТА УПРАВЛІННЯ ПРОЄКТАМИ В УМОВАХ НЕВИЗНАЧЕНОСТІ НА ОСНОВІ AGILE МЕТОДОЛОГІЇ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ**

UDC 004

**R. Kovalchuk**

## **ADAPTATION OF PROJECT CONTROL AND MANAGEMENT TECHNOLOGIES IN CONDITIONS OF UNCERTAINTY BASED ON AGILE SOFTWARE DEVELOPMENT METHODOLOGY**

Ключовою проблемою ІТ-бізнесу в Україні є поступовий відтік проєктів та, як наслідок, переміщення компаній за кордон. Це відбувається через відсутність технологічних та управлінських можливостей збереження ефективності розробки програмного забезпечення в умовах невизначеності та зростаючих загроз.

Діючі програмні рішення та патерни менеджменту допускають наявність людського фактору, несподіваних обставин непереборного характеру і т.д., але жодна система не розрахована на постійну невизначеність. Вже звичні для усіх «стендапи» та «спринти» не можуть бути достатньо ефективними без належної адаптації. Проте, за пошуком рішень потрібно звертатись до тієї ж методології Agile, бо саме вона передбачає постійну адаптацію і вже давно використовується в ІТ. Цікавим фактом є те, що автор Scrum [3] та співавтор Agile Manifesto[4] – Джеф Сазерленд, був військовим льотчиком та здійснив понад 100 бойових вильотів у В'єтнамі. І ідеї концентрації на цілі, адаптуючись до обставин, лягли в основу Scrum та Agile методології загалом.

Фактом залишається те, що значна частина управлінських методологій та технологій перейшла до цивільної сфери саме з військової справи. Обставини склались так, що саме зараз потрібно зробити науково-технічний прорив, а не диверсифікувати бізнес-ризик у класичному розумінні.

Отже, одним з ключових завдань для ІТ спільноти є збереження індустрії в Україні. Не лише бізнесу як такого, шляхом переміщення капіталів та персоналу у інші країни, але й пошуком технологічних рішень, які дозволять ефективно працювати над проєктами в умовах невизначеності для кожного учасника команди. Якщо неможливо ефективно вплинути на обставини, то необхідно адаптувати управління та видозмінити процеси розробки, знайти необхідні технічні рішення. Саме така філософія закладена у Agile методології.

### **Література**

1. «Війна в Україні – це війна технологій». Основні тези Михайла Федорова з виступу на Web Summit 2022. URL: <https://dou.ua/lenta/news/fedorov-web-summit/>.
2. Баран І. О., Воронін В. С. До питання розробки системи збереження даних для хмарної платформи openstack: збірник тез доповідей ІХ Міжнародної науково-технічної конференції молодих учених та студентів «Актуальні задачі сучасних технологій», 25–26 листопада 2020 року. Т.: ТНТУ, 2020. Том 2. С. 6. URL: <http://elartu.tntu.edu.ua/handle/lib/34692>.
3. Scrum: The Art of Doing Twice the Work in Half the Time J. Sutherland. URL: <https://www.scruminc.com/new-scrum-the-book/>.
4. Manifesto for Agile Software Development. URL: <http://agilemanifesto.org/>.

УДК 004.41

**В. Масловський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ВЕБ-ЗАСТОСУНОК МЕРЕЖІ ДОСТАВОК І АСОРТИМЕНТУ ЗАКЛАДІВ ХАРЧУВАННЯ**

UDC 004.41

**V. Maslovskiy**

## **WEB APPLICATION OF THE DELIVERY NETWORK AND ASSORTMENT OF FOOD INSTITUTIONS**

**Ключові слова:** мережа, уніфікований дизайн, асортимент, ранжирування.

**Key words:** network, unified design, assortment, ranking.

Ранжирування – це сортування результатів видачі по запитах користувачів, що застосовується пошуковими системами для видачі актуальної інформації пошуку. При ранжируванні пошукова система аналізує сайти, розставляючи їх в певному порядку в результатах видачі. Основний критерій ранжирування релевантність, тобто відповідність веб-ресурсу запиту користувача.

Мікросервісна архітектура (або просто «мікросервіси») являє собою метод організації архітектури, заснований на ряді служб, що незалежно розгортаються. Ці служби мають власну бізнес-логіку та базу даних з конкретною метою. Оновлення, тестування, розгортання та масштабування виконуються всередині кожної служби. Мікросервіси розбивають великі завдання, характерні для конкретного бізнесу, на кілька незалежних баз коду.

Монолітна архітектура - це традиційна модель програмного забезпечення, яка є єдиним модулем, що працює автономно і незалежно від інших додатків. Моноліт часто називають щось велике і непереможне, і ці два слова добре описують монолітну архітектуру для проектування ПЗ. Монолітна архітектура – це окрема велика обчислювальна мережа з єдиною базою коду, де об'єднані всі бізнес-завдання.

### **Література**

1. Чендлер Харіс. Порівнянн мікросервісної і монолітної архітектури. URL: <https://www.atlassian.com/ru/microservices/microservices-architecture/microservices-vs-monolith>.

УДК 004.41

**Т. Матвієнко, Г. Цуприк**

Тернопільський національний технічний університет імені Івана Пулюя, Україна

## **РОЗРОБКА СИСТЕМИ КООРДИНАЦІЇ ТА ВИЗНАЧЕННЯ КРИТИЧНОГО ШЛЯХУ ПЕРЕМІЩЕННЯ РУХОМОГО СКЛАДУ НА МОВІ JAVA І ФРЕЙМВОРКІВ SPRING, REACT**

UDC 004.41

**T. Matviienko, H. Tsupryk**

### **DEVELOPMENT OF SYSTEM FOR TRAFFIC COORDINATION OF TRACKS MOVING USING JAVA LANGUAGE AND SPRING, REACT FRAMEWORKS**

Актуальність. Сьогодні тяжко уявити звичайний день без використання сучасних технологій. Технології дуже сильно змінили наше життя та впливають на нього постійно. Багато сфер життєдіяльності все більше і частіше впроваджують технологічні інструменти в своє користування. Транспорт, освіта, медицина, державні структури та багато чого іншого не обходяться без цього теж. Все це породжує попит на все нові та вузьконаправленіші застосунки, які можуть використовуватись у певній сфері.

Об'єктом дослідження є: розробка системи координації та визначення критичного шляху переміщення рухомого складу.

Постановка завдання:

- Вибір напрямку дослідження, та сериовища для розробки а також мови програмування.
- Проаналізувати засоби, які допоможуть підвищити продуктивність системи.
- Розробити алгоритми роботи системи.
- Розробити та протестувати програму та дослідити її функціонування.

Вирішення завдання. Для створення повноцінної системи потрібно використати методи, які підходять для цього найкраще. Для цієї роботи був обраний алгоритм Дейкстри. Основне використання алгоритму полягає у знаходженні найкоротшого шляху від вершин графу до всіх інших його вершин. Застосунок буде перетворювати числові координати у координати на мапі, а також в найблищому радіусі знаходити до 300 інших точок для прокладання маршрутів. За допомогою алгоритму Дейкстри і буде знаходитись з усіх варіантів найкоротший або найшвидший.

Для створення серверного застосунку буде використана технологія REST. Завдяки тому, що технологія REST дозволяє звертатись до серверу через порти, то застосунком зможе користуватись одночасно кілька клієнтів. Для реалізації буде використовуватись мова програмування Java та фреймворк Spring.

Для зберігання інформації в базі даних буде використовуватись MySQL.

Оскільки це веб-застосунок з яким будуть працювати інші користувачі, то буде реалізовано графічний інтерфейс користувача. Буде використано мову програмування JavaScript та фреймворк React.js. Цей фреймворк дозволить реалізувати SPA (single page application) – односторінковий застосунок, який полегшить кінцевому користувачеві розібратись з можливостями системи та не буде її перевантажувати зайвими блоками.

#### **Література**

1. «Grokking Algorithms» by Aditya Y. Bhargava. 2016.
2. URL: <https://spring.io/>.
3. URL: <https://uk.reactjs.org/>.
4. URL: <https://www.java.com/en/>.



УДК 004.41

**О. Мельник, М. Петрик**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **РОЗРОБКА АРХІТЕКТУРИ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДІЛОВИХ КОМУНІКАЦІЙ НА ОСНОВІ ОПТИМІЗАЦІЙ СТРІМІНГОВОЇ ПЛАТФОРМИ**

UDC 004.41

**O. Melnyk, M. Petryk**

## **DEVELOPMENT OF ARCHITECTURE FOR BUSINESS COMMUNICATIONS INFORMATION SYSTEM BASED ON STREAMING PLATFORM OPTIMIZATIONS**

**Ключові слова:** Node.js, демультіплексор подій, async/await.

**Key word:** Node.js, demultiplexer of events, async/await.

В сучасному світі більшість людей користуються комп'ютерною технікою, починаючи від смарт годинників закінчуючи великогабаритними обчислювальними серверами. Мінімальні затримки в програмному забезпеченні завжди були однією з найпопулярніших цілей. Бізнес сьогодні потребує хороших і надійних, а найголовніше оптимізованих і швидких інструментів для ведення комунікацій.

З роками почали з'являтися багато платформ та підходів для збільшення ефективності програм. Величезним кроком було створення асинхронної моделі взаємодії запитів та створення новітніх протоколів для передачі даних оминаючи сервер на пряму від клієнта до клієнта.

Асинхронне програмування дозволяє уникнути появи вузьких місць продуктивності та збільшити загальну швидкість реагування програми.

У 2009 році з'являється нова платформа – Node.js. Завдяки подіє-орієнтованій архітектурі вона з легкістю дозволяє працювати з асинхронними запитами.

WebRTC – відносно нова технологія, яка дозволяє оптимізовано та ефективно передавати бінарні дані, в поєднанні з Node.js і асинхронним підходом отримуємо оптимізовану платформу, котра може витримувати великі навантаження та працювати безперебійно. Таке вдале поєднання дозволить одночасно обслуговувати надзвичайно велику кількість запитів та маніпулювати стрімами даних.

Платформа надає різні можливості для ділових комунікацій, зокрема це відеострімінг з великим розширенням, що є досить важким навантаженням.

### **Література**

1. Документація Node.js. URL: <https://nodejs.org/en/docs/>.
2. Petryk M., Khimitch A., Petryk M.M., Fraissard J. Experimental and computer simulation studies of dehydration on microporous adsorbent of natural gas used as motor fuel. Fuel. 2019. Vol. 239. P. 1324–1330.

УДК 004.41

Є. Музиченко, Ю. Стоянов

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## СИСТЕМА МОНІТОРИНГУ ТРАФІКУ В ІЄРАРХІЧНИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ

UDC 004.41

E. Muzychenko, Y. Stoyanov

## TRAFFIC MONITORING SYSTEM IN HIERARCHICAL COMPUTER NETWORKS

**Ключові слова:** трафік, система моніторингу, ієрархічна комп'ютерна мережа, сенсор, колектор.

**Key words:** traffic, monitoring system, hierarchical computer network, sensor, collector.

Система моніторингу трафіку – це мережевий аналітичний інструмент, який перевіряє використання локальної мережі та забезпечує відображення статистики вивантаження та завантаження. Основною метою системи є моніторинг (і підрахунок) IP-трафіку між локальною мережею (LAN) та Інтернетом.

Система моніторингу трафіку забезпечує облік і моніторинг трафіку в реальному часі. Він дуже динамічний, кожне нове підключення реєструється та відстежується, ви можете використовувати його для підрахунку корисного трафіку завантаження та вивантаження комп'ютера або розширити його для побудови системи обліку трафіку для всіх комп'ютерів у локальній мережі вашої компанії.

Системи моніторингу дозволяють контролювати сотні і навіть тисячі параметрів, що стосуються роботи різних апаратних і прикладних підсистем. Крім того, вони забезпечують не тільки збір цих параметрів, але й виконують попередню статистичну обробку, полегшуючи наступний аналіз. На основі зібраних даних виявлені проблеми, що знижують загальну продуктивність системи, перспективний і сценарний аналіз. Зокрема, ми можемо оцінити завантаженість сервера за місяць, квартал чи рік, розрахувати параметри його роботи за рахунок збільшення кількості користувачів (запити, обсяг трафіку, тощо) або визначити завантаженість сервера підсистеми після оновлення.

### Література

1. Даниліна Г. В., Гузій М. М., Ігнатов В. О. Методи і алгоритми оптимального управління трафіком в обчислювальних мережах. Проблеми інформатизації та управління. К: НАУ, 2006. Вип. 17. С. 32–37.

УДК 004

**Н. Новак**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ПОКРАЩЕННЯ ЯКОСТІ РОЗРОБЛЕНОГО ПРОДУКТУ ЗА ДОПОМОГОЮ СИСТЕМ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ**

UDC 004

**N. Novak**

### **IMPROVEMENT OF THE QUALITY OF THE DEVELOPED PRODUCT WITH THE HELP OF THE AUTOMATED TESTING SYSTEM**

Потреба запровадження автоматизованих процесів для перевірки програм в наш час актуалізована зростанням кількості великих інтернет ресурсів електронної комерції і покращення їх якості.

Для застосування автоматизації тестування було обрано платформу Magento2. Це ресурсоємний продукт, який дозволяє управляти продуктами, категоріями, користувачами та замовленнями для кількох різними онлайн магазинів одночасно на основі спільної бази даних.

Забезпечення роботи системи представлено мовою Java і набором бібліотек Maven, який дозволяє конфігурувати систему, використовуючи різні залежності інструментів автоматизації. Також в даному проекті застосовано фреймворк Selenium WebDriver, який дозволяє керувати властивостями веб-браузерів.

Організація функціонального коду представлена шаблоном проектування Page Object, а безпосередньо код тестів забезпечується фреймворком TestNG, з допомогою якого можна організувати структуру тестів.

Автоматизовані тести повинні забезпечити перевірку найбільш повторюваних процесів у системі, таких як реєстрація користувачів та вхід у систему вже раніше зареєстрованих, додавання продуктів в кошик та оформлення покупок, що лежать в основі кожного інтернет-магазину.

Для забезпечення організованого запуску вже готової системи автоматизації було вибрано систему Jenkins, що дає змогу виконувати тести у визначений час, чи в певній ситуації. Також в системі Jenkins інтегрована система управління звітами проходження тестів Allure, що візуалізує кінцеві результати виконання тестів.

Система організована методами контейнеризації, використовуючи Docker.

Таким чином структура організації системи автоматизації забезпечить покращення якості продукту, уникаючи помилок, спричинених людським ресурсом.

#### **Література**

1. Баран І. О., Воронін В. С. До питання розробки системи збереження даних для хмарної платформи openstack: збірник тез доповідей ІХ Міжнародної науково-технічної конференції молодих учених та студентів «Актуальні задачі сучасних технологій», 25–26 листопада 2020 року. Т.: ТНТУ, 2020. Том 2. С. 6. URL: <http://elartu.tntu.edu.ua/handle/123456789/8584>.
2. Arnon Axelrod. Complete Guide to Test Automation: Techniques, Practices, and Patterns for Building and Maintaining Effective Software Projects 1st ed. Edition, 2018.
3. C. Kaner, J. Falk, H.C. Nguyen. Testing Computer Software, 2nd Edition, 2001. 544 p.

УДК 004.41

**О. Остапчук, Г. Цуприк**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ТЕХНІЧНІ ОСОБЛИВОСТІ ВЗАЄМОДІЇ МІЖ КЛІЄНТОМ ТА СЕРВЕРОМ У РЕАЛЬНОМУ ЧАСІ**

UDC 004.41

**О. Ostapchuk, H. Tsuprik**

### **TECHNOLOGICAL FEATURES OF REAL-TIME CLIENT-SERVER COMMUNICATION**

**Ключові слова:** WebSocket, REST, API, зв'язок в реальному часі.

**Key word:** WebSocket, REST, API, real-time communication.

В сучасному світі людство розвивається та живе в шаленому темпі. Дедалі важливішим стає економія часу. Все більше й більше людей починає використовувати комп'ютерні технології в різних галузях життя. Перебуваючи в такому середовищі ми починаємо привикати до швидко отриманої інформації, яка нас може зацікавити. В галузі освіти це є дуже важливий аспект. Також при взаємодії з ментором або викладачем, ми хотіли б отримати інформацію чим швидше. Потрібні методи передачі даних наближені до "в реальному часі".

Історично склалося так, що створення веб-додатків які потребували двонаправності зв'язку між клієнтом і сервером вимагало надмірного використання HTTP запитів для опитування серверу та для оновлення даних з метою наближення показників отриманої інформації до запитів у реальному часі. Це призводить до різноманітних проблем: Сервер змушений використовувати низку різних базових TCP підключень для кожного клієнта: одне підключення для надсилання інформації до клієнта і новий для кожного вхідного повідомлення; Сценарій на стороні клієнта змушений підтримувати відображення вихідних підключень до вхідних з'єднань для відстеження відповідей.

Простішим рішенням було б використовувати одне TCP-з'єднання для запитів в обох напрямках. Ось для чого використовується протокол WebSocket. У поєднанні з API WebSocket він надає альтернативу запиту HTTP для двостороннього зв'язку веб-сторінки з віддаленим сервером. Той самий прийом можна використовувати для різноманітних веб-додатків: ігри, багатокористувацькі програми з одночасним редагування, освітні системи, тощо. Протокол WebSocket призначений для заміни існуючої технології двонаправленого зв'язку, які використовують HTTP як транспортний рівень, щоб отримати переваги від існуючої інфраструктури (проксі, фільтрація, автентифікація). Такі технології були впроваджені як компроміси між ефективністю та надійністю, оскільки спочатку HTTP не був призначений для використання для двонаправленого зв'язку. Дизайн WebSocket не є обмеженим HTTP технологією, і майбутні реалізації можуть використовувати простіше рукостискання через виділений порт без переосмислення всього протоколу. Цей останній момент важливий, оскільки моделі трафіку інтерактивного обміну повідомленнями не відповідає стандартному трафіку HTTP і може викликати незвичайні навантаження на деякі компоненти.

Саме тому технологія WebSocket є однією з кращих можливостей вирішити проблему затримки між запитом, відповіддю та взаємодією «в реальному часі».

#### **Література**

1. Документація WebSockets. URL: <https://websockets.spec.whatwg.org/>.

## **РОЗРОБКА СИСТЕМИ ОПОВІЩЕННЯ ДЛЯ ДИСТАНЦІЙНОГО НАВЧАННЯ ЗА ДОПОМОГОЮ TELEGRAM BOT API НА МОВІ ПРОГРАМУВАННЯ C#**

## **NOTIFICATION SYSTEM DEVELOPMENT FOR DISTANCE LEARNING USING TELEGRAM BOT API AND C# PROGRAMMING LANGUAGE**

ТНТУ надає можливість студентам навчатися дистанційно. Дана платформа має усе необхідне для навчання. Матеріали з дисциплін згруповані в курси. На курсах є можливість читати лекції із заданих дисциплін, проходити тестування для перевірки рівня знань студентів та місце для збереження студентських робіт. Курси адмініструють інструктори – викладачі дисциплін.

Доступ до інформації з платформи для дистанційного навчання можна отримати лише зайшовши в електронний кабінет. З цього випливає що переглянути наявність нових повідомлень та відповісти на них можна лише з кабінету користувача. Це створює незручності у вигляді затримки відповіді на лист. Тому слід періодично заходити в кабінет користувача. Таку роботу можна автоматизувати.

Основні вимоги для вирішення проблеми:

- своєчасне отримання сповіщень;
- швидкий доступ до відповіді на повідомлення;
- швидкий доступ для написання нового листа;
- система повинна бути доступною для користування.

Своєчасне отримання сповіщень забезпечить прискорення спілкування між користувачами платформи для дистанційного навчання. Швидкий доступ до відповіді надасть можливість своєчасно реагувати на повідомлення. Щоб попередні пункти виконались система повинна бути максимальною доступною для користувачів. Критерій доступності задовольняють додатки для смартфонів, а саме додатки для спілкування – месенджери. Вони є надзвичайно популярними саме через швидкий доступ до повідомлень. Один з таких месенджерів це

«Телеграм». Платформа «Телеграм» має наступні характеристики для вирішення проблеми:

- доступ здійснюється з усіх основних операційних систем;
- платформа є безкоштовною для використання;
- платформа має API для доступу через HTTP протокол;
- платформа має бібліотеки для використання API через мови різні
- мови програмування;
- доступні чат боти та API до них;
- детальна документація;
- усі дані зберігаються на веб серверах, а не на пристроях;
- одночасний доступ до платформи з кількох пристроїв.

Для логічної частини системи оповіщення буде використано мову програмування C# та платформу .NET. Мова програмування C# є загального призначення. З її допомогою можна створювати застосунки для різних платформ. У межах кваліфікаційної роботи буде використано консольний застосунок. Його можна запускати як на операційну систему «Windows» так і на «Linux». Крос-платформність мови дозволяє тримати одну кодову базу для усіх платформ. У випадку прив'язки до конкретної операційної системи кодову базу можна логічно розділити на бібліотеки. Це надасть можливість винести загальну кодову базу в крос-платформну бібліотеку та використовувати її модулями що прив'язані до певної операційної системи.

УДК 004.41

**О. Петрик, І. Стадник**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ОБЛІКУ ВНУТРІШНЬО ПЕРЕМІЩЕНИХ ОСІБ**

UDC 004.41

**O. Petryk, I. Stadnyk**

## **DEVELOPMENT OF AN AUTOMATED ACCOUNTING SYSTEM FOR INTERNALLY DISPLACED PERSONS**

У наш важкий час надзвичайно велика кількість українців, нажаль, отримала статус ВПО та були змушені перебратися до західної України в пошуках прихистку. Готелі та різні бази, які знаходяться на західній частині нашої держави можуть стати чудовим варіантом для цих людей. Тому у сьогоднішній нагальною проблемою стало встановлення порядку у веденні документальних різних готельних систем. На основі цього було вирішено створити умови для закладів, які зможуть з легкістю автоматизувати та вести облік таких людей у своїх системах. Умови полягають у створенні систем інформаційного управління.

Аналізуючи вимоги майбутньої системи дозволив отримати уявлення про те, як система повинна надавати послуги своїм користувачам. У процесі було переглянуто кілька продуктів, які зараз представлені на ринку, одна з них Fidelio Front Office. Вона дозволяє проводити автоматизацію бронювання, реєстрації і виселення відвідувачів із веденням документації. Її основні переваги це: легкість, зручність, безпека та гнучкість. Також зроблено аналіз браузерних версій деяких програм, які дозволяють бронювати номери, надають інформацію про готель. У результаті аналізу предметної області було визначено основні проблеми:

- Більшість систем на ринку нашої держави є зарубіжними, що також піднімає планку для початкового користувача, адже потрібно мати навички іноземної мови при роботі з нею.
- Однією з найбільш поширених систем на ринку є Fidelio.
- Впровадження нових систем розробленими закордонними фірмами вимагає від роботодавця робити додаткові видатки на підвищення кваліфікації.

Враховуючи усе вищесказане, було вирішено задля правильного функціонування місць розміщення ВПО, потрібно надати відповідне комп'ютерне забезпечення. Дане забезпечення локалізовано під наш ринок та дозволить бізнесу витратити менше коштів на підтримку програмного забезпечення.

Під час проектування програмного забезпечення було описано ключові сценарії використання системи, визначено кінцевого користувача та виконуванні дії. Початок планування складався з виявлення основи програми та її сутностей, після цього було побудовано схему бази даних, а саме приведення кожної таблиці до певної нормальної форми. Проведено проектування програми та проведено комплексне тестування усієї програми.

Отже, після реалізації проекту було проведено комплексне тестування усієї програми. Тестування – це дії, які націлені на перевірку правильності програми, тому цей процес є одним з ключових у життєвому циклі створення програмного забезпечення. На кінцевій стадії відбулася дослідна експлуатація системи, усі недоліки було виправлено. У результаті ми отримали конкурентноспроможну систему, яка вирішує нагальні питання, що виникли на ринку за останній час.

### **Література**

1. Диго С. М. Проектирование и использования баз данных. М.: Финансы и статистика, 1995. 208 с.
2. Кириллов В. В. Основы проектирования реляционных баз данных: учебное пособие. СПб.: ИТМО, 1994. 90 с.

УДК 004.41

**В. Простяк**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **РОЗРОБКА WEB-САЙТУ ІНТЕРНЕТ-МАГАЗИНУ ДЛЯ ПРОДАЖУ СПОРТИВНОГО ОДЯГУ НА ОСНОВІ MAGENTO 2**

UDC 004.41

**V. Prostiak**

## **WEB SITE DEVELOPMENT OF AN INTERNET STORE FOR THE SALE OF SPORTS CLOTHING BASED ON MAGENTO 2**

**Ключові слова:** Magento 2, eCommerce, Zend Framework, менеджер об'єктів, одинак, ін'єкція залежностей, фабрика, об'єктно-орієнтований підхід (ООП)

**Key word:** Magento 2, eCommerce, Zend Framework, object manager, singleton, dependency injection, factory, object-oriented approach

Останніми роками в усьому світі все більшого поширення набувають методи ведення бізнесу в Інтернеті, зокрема. інтернет-торгівля. Застосування Інтернету дозволяє швидко і з незначними витратами вивести і просувати продукцію на національний і міжнародні ринки. Торгівля через Інтернет дозволяє істотно знизити вартість продукції, оскільки відпадають потреби в утриманні торгових площ, придбанні торговельного обладнання тощо. Крім того, споживач може швидко знайти потрібний товар, вивчити його характеристики, ознайомитися з відгуками інших споживачів, обрати зручний спосіб і час доставки товару, провести платежі через Інтернеті і т. ін.

З роками розробка інтернет-магазинів зазнала значних змін. Через велику популярність цієї галузі почало з'являтися багато платформ на основі яких програмісти могли б легко створити унікальний інтернет-магазин. Великим проривом у цій сфері стала eCommerce платформа з відкритим кодом Magento.

Magento – це програмне забезпечення створене з використанням Zend Framework. Перша версія була випущена 31 березня 2008 року. Через велику кількість недоліків та вразливостей систему постійно оновлювали та у 2015 році відбувся випуск бета-версії Magento 2.

Magento 2 має багато нових і покращених функцій, інструментів розробника, а його архітектура значно відрізняється від усіх попередніх версій. Одною з основних відмінностей стало те, що у Magento 2 відмовилися від використання менеджера об'єктів та одинака, натомість у нових версіях використовується ін'єкція залежностей та фабрики для створення об'єктів. Для написання back-end частини сайту використано мову програмування php та об'єктно-орієнтований підхід (ООП) в програмуванні.

Платформа надає можливість для різного роду ведення продаж, кількість магазинів легко розширити, тематику магазину можна швидко змінити відповідно до вимог замовника, система працює швидко та надійно.

### **Література**

1. Документація Magento 2. URL: <https://devdocs.magento.com/>.

## ПІДВИЩЕНИЙ РІВЕНЬ ПРОКРАСТИНАЦІЇ В ЛЮДЕЙ ЧЕРЕЗ ЗБІЛЬШЕННЯ ІНФОРМАЦІЇ І ДОСТУПУ ДО НЕЇ

UDC 004.41

N. Svytak

## INCREASED LEVEL OF PROCRASTINATION IN PEOPLE DUE TO INCREASING INFORMATION AND ACCESS TO IT

**Ключові слова:** прокрастинація, матриця, планування.

**Key words:** procrastination, matrix, planning.

Прокрастинація зазвичай концептуалізується як ірраціональна тенденція відкладати виконання необхідних завдань або доручень, незважаючи на негативний вплив цього відкладення на окремих осіб та організації.

Вважається, що прокрастинація є невдачею саморегуляції, пов'язаною з різними особистими та ситуативними детермінантами. Зокрема, дослідження показують, що характеристики завдання (наприклад, нечіткі інструкції, час винагород і покарань, а також несприйнятливність до виконання завдання), особистісні аспекти (наприклад, п'ятифакторна модель, мотивація та пізнання) і фактори середовища (наприклад, спокуса, стимули та відповідальність) є основними детермінантами прокрастинації.

Рисунок 1 показує, розподіл літератури протягом 1900–2020 років. Тобто збільшення публікацій і досліджень пов'язано із збільшенням цієї проблеми в людей.

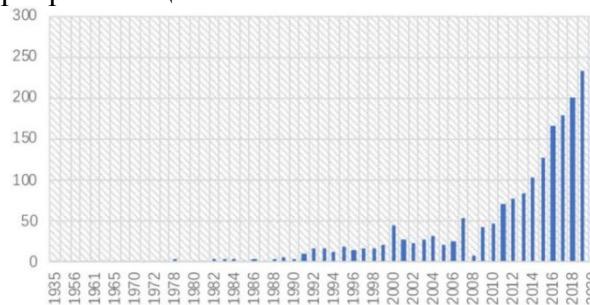
Сенс матриці Ейзенхауера полягає в оцінці конкретного завдання по параметрам важливості/терміновості і сортування списку справ за цим принципом. Мета цієї матриці полягає в плануванні, або оцінці прийняття рішень.

Цей метод фокусується на вирішенні питань управління часом і завданнями через неправильне визначення пріоритетів. Процес аналітичної ієрархії допомагає особі, яка приймає рішення максимально об'єктивно розставити пріоритети та прийняти найкраще рішення. Метод переважно вимагає певної логіки та математики від простого до складнішого. На жаль, ми не можемо просто сказати собі припинити відкладати. І незважаючи на поширеність різних методик підвищення продуктивності, зосередження на питанні про те, як зробити більше роботи, не усуває першопричину прокрастинації.

Підсумовуючи вище описане, можна зробити висновок, що ріст цієї проблеми присутній і ігнорування її не є її вирішенням. Тому використання цього або якщо буде знайдено кращого рішення є одним з кроків для його подолання.

### Література

1. Lay C. (1986). At last, my research article on procrastination. J.
2. Harris N. N., Sutton R. I. (1983). Task procrastination in organizations: a framework for research.



**Рисунок 1.** Розповсюдження публікацій на тему прокрастинації



УДК 004.4

**I. Сіжук, В. Бревус**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **РОЗРОБКА ЗВУКОВИХ ЕФЕКТІВ ДЛЯ ПЛАГІНІВ ВІРТУАЛЬНОЇ СТУДІЇ ЦИФРОВИХ ЗВУКОВИХ РОБОЧИХ СТАНЦІЙ**

UDC 004.4

**I. Sizhuk, V. Brevus**

## **DEVELOPMENT OF SOUND EFFECTS FOR VIRTUAL STUDIO PLUG-INS' DIGITAL AUDIO WORKSTATIONS**

Технологія VST (технологія віртуальної студії) стала помітною галуззю в розробці аудіопрограмного забезпечення, використовуючи цифрову обробку сигналу для імітації традиційного обладнання студії звукозапису за допомогою програмного забезпечення у вигляді зовнішніх плагінів. Вона виявилася надзвичайно практичною та надійною [1].

Окреме значення в галузі займає ефект (здебільшого як окремий плагін), що впливає на звуковий сигнал чином насичення або зміни його гармонічного, тембрального складу (distortion) шляхом синтезу деформації, при якому складні спектри утворюються з простих тонів шляхом зміни форми сигналів за допомогою формуючої кривої (Waveshaping). Своєї актуальності тип деформації (distortion) формування (waveshaping) набув з появою технології VST та цифрових звукових робочих станцій (DAW).

Ринок waveshaping distortion плагінів не насичений конкурентами. Однією з причин є «штучний», «цифровий» характер звучання таких плагінів, що далеко не завжди є метою кінцевого характеру звуку. Але коли ціллю є якраз описаний вище характер звучання, то waveshaping distortion ефекти підійдуть як ніякі інші. За допомогою формування хвилі досягаються самі оригінальні та багаті на гармоніки тембри та спектри. Часто це агресивні, масивні (базовані на низьких частотах) звуки. Представникам такого типу технічних рішень не вистачає параметрів, за допомогою котрих можна було б пом'якшити ефект дії заданої «агресивної» формуючої кривої.

Пропонується рішення waveshaping distortion алгоритму, формуючу криву, котрого можна змінити не тільки за допомогою графічного редактора, але й за допомогою типу деформації (warp), що використовується у тибличному синтезі: bend [2]. Цей тип деформації в залежності від обраного режиму роботи дозволяє: затискати або згинати форму хвилі всередину (до середини циклу хвилі), «витягувати» або згинати хвилю назовні (до країв циклу хвилі).

### **Література**

1. Virtual Studio Technology inside Music Production. URL: [https://sci-hub.se/https://link.springer.com/chapter/10.1007/978-3-319-01466-1\\_22](https://sci-hub.se/https://link.springer.com/chapter/10.1007/978-3-319-01466-1_22).
2. Serum: Advanced Wavetable Synthesizer. URL: <https://xferrecords.com/products/serum>.

УДК 004.43

**Д. Сомін, А. Паламар, В. Волоський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ПЕРЕВАГИ WEBASSEMBLY ЯК ІНСТРУМЕНТА РЕАЛІЗАЦІЇ АЛГОРИТМІВ У РЕСУРСОМІСТКИХ ВЕБ-ДОДАТКАХ**

UDC 004.43

**D. Somin, A. Palamar, V. Voloskyi**

## **WEBASSEMBLY ADVANTAGES AS A TOOL FOR ALGORITHMS IMPLEMENTATION IN HIGH-LOAD WEB APPLICATIONS**

Сьогодні великої популярності набуло використання веб застосунків для різноманітних обчислень. Разом з тим, процес їх розробки супроводжується низкою проблем: складність масштабування, швидкість виконання, вартість хостингу, тощо. Значним недоліком більшості веб платформ є абсолютне домінування мови JavaScript (JS) та її похідних, як інструменту розробки додатків.

Попри наявність сильних сторін, JavaScript має значні недоліки: низька ефективність та великий об'єм коду, відсутність типу даних «ціле число», повільна швидкість виконання, неповна підтримка стандартів та різна поведінка браузерів під час інтерпретації. Тому, актуальною задачею є оптимізація складних алгоритмів за допомогою інших мов програмування.

Одним із методів оптимізації JavaScript алгоритмів є технологія WebAssembly (Wasm), яка містить опис незалежного від браузера, універсального, низькорівневого проміжного коду для запуску застосунків, скомпільованих на різних мовах програмування, що має на меті компенсувати недоліки JavaScript.

Перевагами WebAssembly є можливість написання та компіляції певного коду на низькорівневих мовах програмування, таких як C++, Go, Rust, та використання його у веб додатках. Це дозволяє підвищити швидкодію виконання окремих модулів веб додатків. Також, значною перевагою WebAssembly є можливість написання багатьох потокових програм.

В середньому Wasm є швидшим ніж JavaScript, але в кожному окремому випадку потрібно робити порівняння цих двох технологій, оскільки використання Wasm може дати як і набагато кращий результат так і навпаки. В значній мірі це залежить від браузера, який використовується для запуску веб додатка.

Як правило, Wasm добре показує себе там, де велика кількість операцій з пам'яттю, наприклад: шифрування, хешування, робота з масивами даних та інше. В серпні 2019 року була опублікована стаття [1], у якій автори реалізували можливість запуску під WebAssembly консольних утиліт Linux, для запуску бенчмарків. В результаті досліджень було виявлено, що швидкість виконання коду Wasm на 30 % вища, ніж JavaScript. Разом з тим, код, написаний на Wasm, на 50% повільніший ніж двійковий код. З моменту публікації, WebAssembly отримав популярність та кращу підтримку в браузерах, що збільшило його пікову і середню продуктивність.

Отже, для оптимізації роботи з даними у веб додатках доцільніше використовувати WebAssembly у порівнянні з JS, тому що приріст продуктивності може досягати 30 %.

### **Література**

1. Bearman D. Office of the Secretary: Evaluation of Email Records Management and Cybersecurity Requirements, ESP-16-03. The American Archivist. Vol. 80. No. 2. 2017. P. 459–462.

УДК 004.4

**І. Стефанишин, М. Петрик**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ІДЕНТИФІКАЦІЇ ПАРАМЕТРІВ АНОРМАЛЬНИХ НЕВРОЛОГІЧНИХ РУХІВ ЛЮДИНИ ПІД ДІЄЮ ТЕХНОГЕНИХ НАВАНТАЖЕНЬ**

УДК 004.4

**I. Stefanyshyn, M. Petryk**

## **DESIGN OF THE INFORMATION SYSTEM FOR THE IDENTIFICATION OF THE PARAMETERS OF ABNORMAL NEUROLOGICAL HUMAN MOVEMENTS UNDER THE EFFECT OF TECHNOLOGICAL LOADS**

На сьогодні існує багато захворювань на різні захворювання, які спричиняють дисфункцію моторно-рухових рухів людини. Особливу увагу приділяють захворюванням верхніх кінцівок людини. Відповідно до даних Всесвітньої організації охорони здоров'я в Україні в середньому 6% осіб у віці 65 років і більше, а також 3,8% осіб до 40 років скаржаться на різні типи тремору [1].

Область діагностування пацієнтів з ознаками тремору має ряд складностей, які заважають медичним працівникам поставити коректний діагноз. До основних перешкод оцінювання можна віднести біологічний, техногенний і людський фактори: різна природа походження тремору, пізні звернення в медичну установу для діагностування тремору, застаріле обладнання та програмне забезпечення. Застосування сучасних інформаційних технологій в медицині дозволяє підвищити якість діагностики захворювань за рахунок надання додаткової інформації про виникнення патологічних процесів.

Метою наукового дослідження є виявлення ознаків тремору під час проходження тест спіралі Архімеда. Базуючись на отриманих даних від цифрового планшету та електроміографа розроблялася математична модель. Ця модель розроблена для оцінювання ступеня захворювання пацієнта.

Проектування та розробка такого роду системи є не із простих. Потрібно бути уважним до кожних дрібниць, бо від результатів у майбутньому залежатиме здоров'я людини. Велика кількість процесів обробки та аналізу даних, отриманих від пристроїв діагностування, додавали свої перешкоди для коректної розробки застосунку [2].

### **Література**

1. Петрик М. Р., Михалик Д. М., Мудрик І. Я. Спосіб цифрового вимірювання параметрів аномальних неврологічних рухів верхніх кінцівок у пацієнтів із проявами тремору. Патент на корисну модель № 130247, Бюл. № 22 від 26.11.2018.
2. Haubenberger D, Kalowitz D, Nahab F B, Toro C, Ippolito D, Luckenbaugh DA, Wittevrongel L, Hallett M. Validation of Digital Spiral Analysis as Outcome Parameter for Clinical Trials in Essential Tremor. *Movement Disorders* 26 (11), 2073–2080, (2011).

УДК 004.41

**О. Сторожук**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## ТЕХНОЛОГІЯ БЛОКЧЕЙН В ОСНОВІ NFT

UDC 004.41

**O. Storozhuk**

## NFT TECHNOLOGY BASED ON BLOCKCHAIN

**Ключові слова:** блокчейн, крипто валюта, блок транзакцій, хеш, хеш-дерево, транзакція.

**Key words:** blockchain, crypto currency, transaction block, hash, hash tree, transaction.

Блокчейн – це розподілена база даних, що зберігає впорядкований ланцюжок записів, так званих блоків, що постійно довшає. Кожен блок містить часову позначку, хеш попереднього блоку та дані транзакцій, подані як хеш-дерево. Тому внесення змін в один з блоків вимагає відповідних змін в усіх блоках після нього, що зазвичай виявляється або дуже складно, або дуже коштовно.

Блок транзакцій – спеціальна структура для запису нових транзакцій в системі Біткоїн та аналогічних їй. Блок містить відомості про транзакції, дерево їхніх хешів, а також заголовок зі службовими даними, де зокрема наведено і хеш попереднього блоку, тож кожен наступний блок є також підтвердженням попереднього.

Щоб транзакція вважалася достовірною («підтвердженою»), її формат та підписи мусять перевірити й записати (разом з іншими транзакціями) в новий блок. Кожен наступний блок посиляється на попередній, тож усі блоки можна вишикувати в один ланцюжок, що являтиме собою історію транзакцій за весь час існування системи. Перший блок ланцюжка – первинний блок (англ. genesis block) – то окремий випадок, бо в нього відсутній материнський блок.

Блок складається із заголовка та списку транзакцій. Заголовок блоку містить свій хеш, хеш попереднього блоку, хеші транзакцій та додаткову службову інформацію. Першою транзакцією в блоку завжди вказується отримання комісії, яка стане нагородою користувачеві за створений блок.

### Література

1. В. Пилявець. Технологія блокчейн. URL: <http://blockchaininsociety.in.ua/index.php/journal/>.

УДК 004.41

**Є. Тимченко, Г. Цуприк**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **РОЗРОБКА РОЗУМНОЇ СИСТЕМИ ОБЛІКУ РОБОТИ КОМП'ЮТЕРНОЇ МЕРЕЖІ ПІДПРИЄМСТВА В СЕРЕДОВИЩІ EMBARCADERO RAD STUDIO XE**

УДК 004.41

**Ye. Tymchenko, H. Tsupryk**

## **DEVELOPMENT OF A ACCOUNTING SMART SYSTEM OF COMPUTER NETWORK SERVICE OF THE ENTERPRISE IN THE EMBARCADERO RAD STUDIO XE ENVIRONMENT**

З необхідністю мати можливість віддалено контролювати та вести діяльність будь-якого напрямку стикнувся сьогодні, вже без перебільшення, кожен керівник малого, середнього та великого бізнесу, а також структур державного підпорядкування. Це реальна можливість пережити скрутні часи, оскільки навіть при прикладанні над зусиль, та скороченні видатків сьогоднішня статистика вражає: кожен третій українець, на жаль, сьогодні позбавився джерел доходів. В статистичному сенсі це страшно, а от якщо розглянути на прикладі реальної родини з двох дорослих та дитини, то, в перекладі з мови цифр, це означає, що весь тягар утримання знаходиться на комусь одному із дорослих членів родини.

Про предмет дослідження можна сказати, що він означається в межах об'єкту дослідження. Зокрема найчастіше моніторинг в організації, будь-якої форми власності все ще ведеться в тому числі письмово, що може призвести до суттєвого впливу так званого людського фактору та, відповідно, не коректного оформлення та не своєчасного заповнення журналів технічного стану персональних комп'ютерів у відділах. А ці проблеми можна легко вирішити використавши сучасні інформаційні можливості, шляхом повної якісної автоматизації процесу контролю та управління.

Така інформаційна система змогла б суттєво полегшити працю як завідувачів (керівників) так і працівників, які закріплені за певним відділом чи підрозділом, а також обслуговуючого персоналу в цілому. Система допоможе зберегти інформацію про технічні параметри та коректність роботи технічних засобів в одному місці, без необхідності ведення звітності в ручну чи напів автоматизовано.

Програмний засіб повинен бути високо надійним. У всьому програмному засобі повинно використовуватися обмеження по введенню даних у поля таблиць. Повинно бути реалізовано надання прав доступу, тобто в залежності від користувача, який працює з інформаційною системою, зокрема, якщо це адміністратор, то він володіє повними правами доступу, а якщо це працівник або завідувач відділом, то відповідно його права будуть дещо обмеженими. До того, як буде здійснюватись занесення нових даних, необхідно виконати перевірку на відповідність усієї внесеної інформації. Програмний засіб повинен характеризуватися достатньо високою швидкістю, тобто обробляти усі операції за відносно короткий відрізок часу.

Програмний засіб повинен коректно та швидко функціонувати під керуванням операційної системи Microsoft Windows XP/Vista. Програмний засіб реалізовано в середовищі Embarcadero RAD Studio XE мовою програмування Delphi/Object Pascal.

### **Література**

1. Бойко І., Петрик М., Цуприк Г. Інформаційні технології видобутку даних (Data mining, високопродуктивні обчислення у складних системах): навчальний посібник. 2020.
2. Петрик М., Михалик Д., Петрик О., Цуприк Г. Методичні вказівки до виконання атестаційної роботи магістра за спеціальністю 121 «Інженерія програмного забезпечення» для усіх форм навчання. Тернопіль: Тернопільський національний технічний університет імені Івана Пулюя, 2020. 27 с.

## СЕКЦІЯ 5. НОВІТНІ ФІЗИКО-ТЕХНІЧНІ ТА ОСВІТНІ ТЕХНОЛОГІЇ

УДК 004.4

**Я. Войтович, А. Лупенко**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

### МЕТОДИ ПОБУДОВИ ШИРОКОСМУГОВИХ МЕРЕЖ ДОСТУПУ

УДК 004.4

**Y. Voitovych, A. Lupenko**

### METHODS OF BUILDING BROADBAND ACCESS NETWORKS

Телекомунікаційні технології відіграють величезну роль у всіх без винятку сферах сучасного суспільства, коли від швидкості, якості та своєчасної передачі інформації залежить правильність прийняття стратегічно важливих рішень. Крім того, телекомунікаційні системи виключно важливі в передачі і доведенні до кожного громадянина політичної, громадської, культурної, освітньої та іншої інформації. Галузь зв'язку виконує найважливішу функцію передачі інформації для забезпечення стратегічної та економічної безпеки країни, життєдіяльності людей, суспільного виробництва, управління на всіх ієрархічних і територіальних рівнях.

Мережі доступу останнім часом викликають зростаючий інтерес у фахівців у зв'язку з їх сервісними можливостями. Вони постійно розширюються, виходять на якісно новий рівень і покривають практично всі типи сервісу: від передачі голосу і даних до мультимедіа та відео.

У роботі розглянуті методи розвитку широкосмугових мереж доступу: мобільний та фіксований зв'язок. Мобільний бездротовий зв'язок представлений технологіями WCDMA та WiMax. Фіксований наземний зв'язок представлений оптичними мережами зв'язку PON та на базі мідних кабелів xDSL.

За допомогою оптичних технологій, особливо технології PON, потоки даних можуть бути організовані для абонента зі швидкістю від 1 до 2,4 Гбіт/с. Оптичні технології постійно розвиваються і дешевшають. Однак, мідна вита пара з покриттям на короткі відстані для підключення центральних вузлів розподілу до абонентів все ще широко застосовується.

Також розглянуто питання оцінки загасання в пасивних оптичних мережах при різних неоднорідностях, а також оцінено загасання в абонентській мережі з мідним кабелем для розгортання систем xDSL.

Здійснено розрахунок втрат в оптичних з'єднувачах, розрахунок параметрів надійності волокон в оптичному кабелі.

#### Література

1. Вишне夫斯基 В., Портной С., Шахнович И. Энциклопедия WiMax. Путь 4G. М.: Техносфера, 2009. С. 472.
2. 3GPP Technical Report 25.942, RF System Scenarios.
3. Бірюков М. Л., Стеклов В. К., Костік Б. Я. Транспортні мережі телекомунікацій: Системи мультиплексування: підручник для студентів вищ. техн. закладів; під ред. В. К. Стеклова. К.: Техніка, 2005. С. 312.
4. Гребенніков В. О., Хиленко В. В. Про стратегію і тактику інформаційного розвитку України. Частина 4. Розвиток мереж радіодоступу. Зв'язок. 2008. № 3, 4. С. 13–21.
5. Corning Cable Systems Evolant Solutions for FTTx Networks. URL: [www.corning.com/cablesystems](http://www.corning.com/cablesystems).

УДК 551.46.077:629.584

**Ю. Попович**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **РОЗРОБЛЕННЯ АВТОМАТИЗОВАНОГО МЕТОДУ КОНТРОЛЮ ПОВЕРХОНЬ МЕТАЛОКОНСТРУКЦІЙ, ВІДНОВЛЕНИХ РОБОТИЗОВАНИМ НАПЛАВЛЕННЯМ**

УДК 551.46.077:629.584

**Yu. Popovych**

## **DEVELOPMENT OF AN AUTOMATED METHOD OF CONTROLLING THE SURFACES OF METAL STRUCTURES RESTORED BY ROBOTIC SURFACING**

В наш час використання роботів в галузі машинобудування України, знаходиться у стані стагнації, ця сфера все повністю монополізована іноземними державами [1]. В даний час вони розробили роботів для зварювання кабін автомобілів, морських суден та інших металоконструкцій.

Зварювальні роботи широко використовуються в автомобільній промисловості, але загалом, у машинобудуванні ситуація інша. Виходячи з виробничих характеристик автомобілебудування, можна було використовувати роботів на певних технологічних операціях, проте для зварювання довгомірних металоконструкцій роботам доводиться рухатися всередині або зовні конструкцій, щоб виконувати зварювальні роботи, що ускладнює застосування роботизованого зварювання. Апаратне забезпечення зварювального робота для машинобудівної промисловості можна розділити на три частини, які включають режим руху, захоплювач робота та зварювальні пристрої. Всі ці складові впливають на якість зварного шва.

Відомі методи опису поверхонь руйнування статистичними параметрами відповідно до підходів, що використовуються в трибології та механіці руйнування, матеріалознавстві [2]. Використовуючи підхід автоматизованого аналізу зображень, встановлено характеристики пор за допомогою попередньої бінаризації та досліджуємо зв'язок між мікроскопічними характеристиками пор і параметрами макроскопічної деформаційної поведінки. Цей метод усуває обмеження традиційних методів розпізнавання зображень, наприклад неможливість отримати повні багатомасштабні характеристики.

Для цього з зони зварного шва вирізали зразки, досліджували їх на статичний розтяг, одержали зображення центральної зони руйнування зразка методом скануючої мікроскопії. Після цього ці зображення бінаризували та на чорно-білих поверхнях обчислювали фрактальну розмірність.

Цей метод можна використовувати для забезпечення всіх переваг точного автоматизованого розпізнавання зображень, і він дозволяє комп'ютеризовано обробляти мікроскопічні зображення, щоб значно підвищити точність ідентифікації пор у зразках зварних швів за роботизованого зварювання. Єдиним недоліком методу є те, що він руйнівний та потребує вирізання зразків з досліджуваної конструкції.

### **Література**

1. Попович О. Аналіз сучасного стану розвитку робототехніки у світі. Техніки Всеукраїнської студентської науково – технічної конференції «Природничі та гуманітарні науки. актуальні питання»: Тернопіль, ТНТУ ім. Івана Пулюя, 2008. С. 62.
2. Попович П. В., Марущак П. О., Дзюра В. О., Шевчук О. С. Оцінка довговічності засобів транспорту в АПК з урахуванням впливу агресивних середовищ. Тернопіль, ТНТУ ім. Івана Пулюя, 2018. 280 с.

УДК 004.031.42

**Г. Осухівська, А. Волощук**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ТЕХНОЛОГІЇ ПЕРЕДАВАННЯ ТА ОПРАЦЮВАННЯ ДАНИХ В КОМП'ЮТЕРИЗОВАНИХ СИСТЕМАХ ОБЛІКУ ЕЛЕКТРОЕНЕРГІЇ**

UDC 004.031.42

**H. Osukhivska, A. Voloshchuk**

### **DATA TRANSMISSION AND PROCESSING TECHNOLOGIES IN COMPUTERIZED ELECTRICITY ACCOUNTING SYSTEMS**

У зв'язку із проблемами в енергетичній інфраструктурі України через військові дії з боку росії, а також із зростанням використання електроенергії споживачами в мирний час постає питання ефективного передавання і опрацювання отриманої інформації в енергосистемі. При цьому важливим є передача даних з генеруючих об'єктів у системи контролю і моніторингу, в тому числі щоденного та щогодинного відстеження використання електроенергії кінцевими споживачами.

Водночас однією з головних проблем є вимірювання спожитої електроенергії, передавання результатів вимірів на центральний сервер через канали зв'язку, зберігання та аналіз даних. Для розв'язання даної проблеми використовується передача даних з електролічильників комп'ютеризованій системі обліку електроенергії за допомогою системи АСКОЕ, яка використовує послідовні інтерфейси RS 485/422/232 або через мережі TCP/IP. Це дозволяє забезпечити передачу інформації про електроспоживання головних вузлів області.

Автоматизована система комерційного обліку електроенергії (АСКОЕ) становить собою обчислювальну систему з центральним вузлом та розподіленою системою вимірювань, проектується як багаторівнева ієрархічна автоматизована система. Завдяки АСКОЕ появляється можливість самостійно переглядати дані щодо енергоспоживання в будь-який час.

Впровадження АСКОЕ забезпечує значну економію коштів на оплату енергоресурсів. Система дає можливість автоматизувати розрахунки споживачів що своєю чергою створює порядок розрахунків та використання енергоресурсів прозорими. Підготовка та передача інформації у відповідні операторські пункти та системні центри АСКОЕ учасників енергетичного ринку відповідно до діючих нормативних актів і договорів, створення єдиного інформаційного простору для захисту комерційних інтересів всіх зацікавлених сторін.

Сучасне АСКОЕ потрібно розглядати як єдину систему, що містить декілька пов'язаних між собою рівнів обладнання, а також програмного забезпечення, що працюють одночасно.

Розвиток даної технології в електромережах викликане не лише збільшенням заощадження енергії та енергоефективності, а й потребою регулювання режимами роботи енергосистем, що на сьогодні є дуже актуальним, оскільки одним із ключових завдань, які стоять перед енергетикою України, є організація надійного та чіткого обліку електричної енергії та її потужності. Потреба швидкого запровадження АСКОЕ в області обґрунтована важливим небалансом обліку спожитої та здійснюваної електричної енергії, яка тягне за собою допоміжні витрати електронної енергії.

На даний момент часу впровадження АСКОЕ є доцільним кроком у вирішенні багатьох проблем, насамперед пов'язаних з обліком та контролем витрат електроенергії та забезпечення контролю за відхиленням режимів споживання і розподілу електричної енергії.



УДК 004

**Ю. Дзюбак, Я. Коненко, Н. Луцик**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ ХОСТИНГУ РЕСУРСІВ З ВИКОРИСТАННЯМ ВІРТУАЛІЗАЦІЇ**

UDC 004

**Yu. Dziubak, Ya. Konenko, N. Lutsyk**

## **ANALYSIS OF METHODS AND MEANS RESOURCE HOSTING USING VIRTUALIZATION**

Віртуалізація – це створення гнучкої заміни реальних ресурсів – з тими самими функціями та зовнішніми інтерфейсами, що й у фізичних прототипів, але з різними атрибутами, такими як розмір, продуктивність та вартість. Така заміна називається віртуальними ресурсами.

Віртуалізація зазвичай застосовується до фізичних апаратних ресурсів шляхом об'єднання кількох фізичних ресурсів у загальні пули, у тому числі користувачі отримують віртуальні ресурси. За допомогою віртуалізації з одного фізичного ресурсу можна зробити кілька віртуальних.

Розглянуто можливості, процеси розгортання та питання управління віртуальної машини для п'яти гіпервізорів, які допомагають забезпечити віртуалізацію систем у хмарі – PowerVM™, VMware ESX Server, Xen, KVM та z/VM®. Досліджено переваги та недоліки продуктів і платформ віртуалізації, що надаються, від таких компанії як Citrix, Microsoft, Oracle, Parallels, Red Hat, VMware.

Для технологій віртуалізації та панелей управління вивчено мінімальні та рекомендовані системні вимоги. Розглянуті рішення використовують повну віртуалізацію, тому для експериментів необхідна апаратна віртуалізація. Апаратна віртуалізація – віртуалізація за допомогою спеціальної процесорної архітектури. На відміну від програмної віртуалізації, за допомогою техніки можливе використання ізольованих гостьових систем, керованих гіпервізором безпосередньо. Гостьова система не залежить від архітектури хост платформи та реалізації платформи віртуалізації.

Апаратна віртуалізація також забезпечує продуктивність близьку до продуктивності невіртуалізованої машини, що дає віртуалізації можливість практичного використання та тягне за собою її широке поширення. Найбільш поширені технології віртуалізації Intel-VT та AMD-V.

Дослідження показали, що лідером ринку залишається VMware з гіпервізором ESXi, інші комерційні пропозиції, а саме Citrix (Xen), Oracle VM (Xen), Red Hat (KVM) та Parallels (OpenVZ). ESXi, Xen, KVM та OpenVZ використовують гіпервізори з відкритим вихідним кодом і є основними технологіями віртуалізації. Наймасовішим гіпервізором буде ESXi, тому що рішення VMware контролюють та розвивають основну частину ринку віртуалізації. Гіпервізор Xen розвивається за рахунок просування великими корпораціями як Citrix, Oracle, Amazon і т.д., але він може поступитися місцем KVM, що активно розвивається.

Ринок інфраструктури віртуалізації серверів є основою для двох надзвичайно важливих тенденцій на ринку, які мають зв'язок: модернізація інфраструктури та «хмарні» обчислення. Для модернізації інфраструктури використовується віртуалізація, щоб покращити використання ресурсу, зменшити витрати, покращити ефективність використання енергії, покращити швидкість доставки ресурсу та інкапсулювати образ робочого навантаження, що включає автоматизацію. У цьому відношенні віртуалізація являє собою горизонтальний тренд для переважної більшості підприємств і робочих навантажень, які зрештою теж стають віртуалізованими.

### **Література**

3. John L. Hennessy, David A. Patterson Computer Architecture, Sixth Edition, Morgan Kaufman Publishers 2019. P. 118.
2. Mike Mayers CompTIA+ Certification (exams 220-1001&220-1002) McGraw-Hill Education 2019. P. 206–322.

УДК 004

**Ю. Дзюбак, Я. Коненко, Я. Войтович**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ТЕСТУВАННЯ ТА ПОРІВНЯЛЬНИЙ АНАЛІЗ ПЛАТФОРМ ВІРТУАЛІЗАЦІЇ**

UDC 004

**Yu. Dziubak, Ya. Konenko, Ya. Voytovych**

### **TESTING AND COMPARATIVE ANALYSIS OF VIRTUALIZATION PLATFORM**

Встановлено і використано KVM для створення і запуску віртуальних машин на сервері з CentOS. Описано методику створення заснованих на образах віртуальних машин і віртуальні машини, які використовують логічний том (LVM). KVM використовує апаратну віртуалізацію, для його роботи потрібен процесор, який підтримує апаратну віртуалізацію, наприклад, Технологія Intel VT або AMD-V.

Розглянуто технології віртуалізації на рівні операційної системи (VirtualBox) і повної віртуалізації (KVM і Xen). Як обладнання для експерименту обрали рішення на базі процесора Intel Haswell Core i7 4770K. В якості операційної системи обрано Fedora 32. Продуктивність при використанні технологій віртуалізації тестувалися за допомогою Phoronix Test Suite та результати опубліковувалися на [openbenchmarking.org](http://openbenchmarking.org).

Для керування віртуальними серверами достатньо використовувати virsh - один із головних інструментів керування libvirt. Але для автоматизації VPS/VDS та адміністрування хостмашин потрібна панель управління. Є безліч панелей управління для KVM. Обрано панель управління для комерційного використання – Solus Virtual Manager (SolusVM). Розробник - британська компанія Soluslabs Ltd. SolusVM керується через вебінтерфейс, можна підключити до білінгу. Для управління всіма нодами потрібна лише одна майстер-нода, встановлена на VPS. Панель працює під операційними системами, як CentOS 6+ та подібних.

При тестуванні гіпервізорів з'ясувалося, що віртуалізація лише на рівні операційної системи підходить не для всіх сфер застосування. Зокрема, не підходить для алгоритмів високоточних обчислень. Для високонавантажених проєктів і науково-дослідних робіт найкраще підходить KVM, оскільки ресурси, виділені конкуруючим віртуальним машинам, добре розподілялися за великих навантажень. Xen найкраще використовувати для декодування/кодування відео, 3D візуалізації та рендерингу. Компанія Citrix Systems використовує Xen не тільки для віртуалізації серверів, але і робочих столів, ця технологія найкраще підходить для таких рішень.

Результати порівняльного тестування показали, що серед рішень повної віртуалізації KVM також найкраще підходить для хостингу. Технологія OpenVZ, яка спочатку створювалася як технологія для віртуального хостингу, випереджає за показниками щільності, гнучкості та масштабованості. Хоча більшість компаній, які впроваджують високонадійні проєкти, вибирають VMware. KVM та Xen мають менше поширення.

Неможливо керувати великою кількістю віртуальних машин та кількома хост машинами з консолі. В основному панель керування oVirt призначена для керування хостмашинами на базі KVM, але її можна налаштувати для роботи з Xen та VirtualBox. SolusVM надає можливість підключення хост машин на базі KVM, Xen (у режимі PV та HVM). Ліцензія для SolusVM коштує \$10 і тому популярна серед хостингових компаній. Ліцензії VMware коштують дуже дорого, але вони пропонують кращу панель управління та пропонують великі можливості адміністрування.

Ринок інфраструктури віртуалізації серверів є основою двох надзвичайно важливих тенденцій на ринку, які мають зв'язок: модернізація інфраструктури та «хмарні» обчислення. Для модернізації інфраструктури використовується віртуалізація, щоб покращити використання ресурсу, зменшити витрати, покращити ефективність використання енергії, покращити швидкість доставки ресурсу та інкапсулювати образ робочого навантаження, що включає автоматизацію. У цьому відношенні віртуалізація являє собою горизонтальний тренд з переважною більшістю підприємств і робочих навантажень, які зрештою теж стають віртуалізованими.

## ОПТИМІЗАЦІЯ РОБОТИ ВИПАРНОЇ СТАНЦІЇ ПРИ ВИГОТОВЛЕННІ СОКУ

UDC 681.5

A. Kurko, M. Khomiak, K. Ursta, A. Shelelo

## OPTIMIZATION OF EVAPORATION STATION DURING JUICE PRODUCTION

Для раціонального використання теплових ресурсів за допомогою оптимального керування випарною станцією на цукровому заводі необхідно здійснювати контроль роботи контурів випарної станції, протокол управління яких визначається на основі поточних оцінок технологічних параметрів, таких як: рівні і температура в корпусах станції, споживання соку і сиропу, теплофізичні характеристики пари, а також рівень її споживання технологічними установками фабрики.

Метою роботи було оптимізувати роботу випарної станції випарювання соку.

Було розроблено алгоритм визначення стану випарної станції як об'єкта керування. Для дослідження застосовували метод кластеризації на основі самоорганізованих карт Кохонена, описаний в [1]. Він дозволив визначити набір можливих станів об'єкта на основі інформації, прихованої в часових рядах технологічних параметрів випарних станцій. Застосування методу нечіткої класифікації дозволило визначити стан випарної станції в поточний момент часу на основі значень поточних параметрів випарної станції та отриманої множини можливих станів об'єкта.

Для набору дослідних даних було розроблено систему керування з можливістю збору всіх параметрів виробничого процесу в реальному часі. Розроблена система керування забезпечує якісний контроль процесу випарювання.

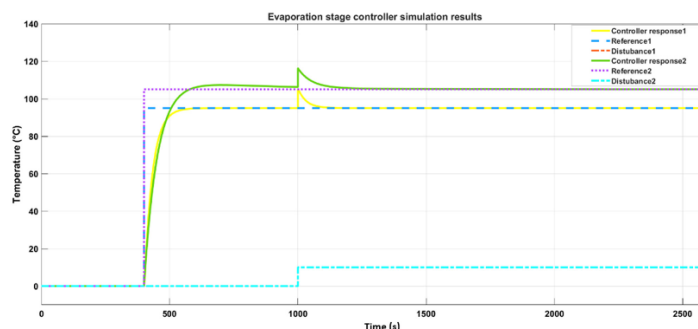
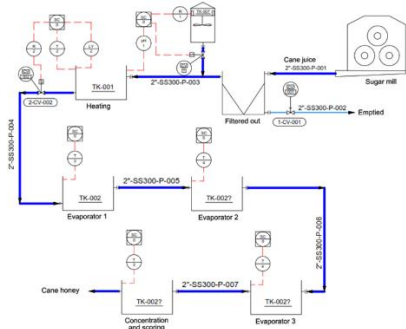


Рисунок 1. Структурна схема автоматизації та перехідний графік ПІД регулювання контуру температури

### Література

1. Ladanyuk A., Kyshenko V., Shkolna E., Sych M. Development of the algorithm of determining the state of evaporation station using neural networks. Eastern-European Journal of Enterprise Technologies, 5 (2 (83)), 54–62.
2. Mejía V. C., Mejía G. C., Rubio O. G., Cortes I. B., Yordi E. G., Tenezaca B. O., Miño Valdés, J., Suárez E. G., Martínez A. P. Automatic Control System for Cane Honey Factories in Developing Country Conditions. Processes. 2022. 10 (5), 915.

**МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ  
КОНТРОЛЮ ПРОЦЕСУ ФЕРМЕНТАЦІЇ****MATHEMATICAL SUPPORT OF THE COMPUTER SYSTEM FOR  
CONTROL OF THE FERMENTATION PROCESS**

При розробці комп'ютерної системи контролю процесу ферментації винних продуктів важливим є вибір та обґрунтування відповідного математичного забезпечення. У літературі можна знайти багато різних математичних систем, що моделюють процес виробництва вина на основі різної кінетики. Зазвичай враховують зміни стану таких показників, як дріжджі, цукор, спирт та концентрація азоту, що засвоївся. Класична модель яка використовується для математичного представлення запропонована David et al. [1], що описується диференціальними рівняннями. Ця модель сформульована з кінетикою Міхаеліса-Ментена [2].

Загалом, для моделювання процесу бродіння вина існує безліч моделей, заснованих на звичайних диференціальних рівняннях. За основу було взято модель, описану в [3], яка представляє еволюцію біомаси дріжджів та інших концентрацій субстратів, таких як азот, цукор та етанол. Цукор перетворюється на етанол, а дріжджі ростуть шляхом метаболізму цукру та інших поживних речовин. З цієї відомої моделі було виведено модель, засновану на звичайних диференціальних рівняннях, яка також враховує "смерть" дріжджів, пов'язану з киснем та етанолом.

Для опису процесу контролю за ферментацією винних продуктів доцільно в моделях враховувати показники температури та CO<sub>2</sub> з метою прогнозування проведення технологічних операцій (наприклад, мінімізація старіння дріжджів). Модель, яка пропонується, описується:

$$\int F(t, y(t), u(t)) dt + \tau c$$

З урахуванням:

$$y(t) = f(t, y(t), u(t), d, p), \forall t \in [t_c, t_c + \tau] \quad y(t_c) = \hat{y}(t_c) \\ c(t, y(t), u(t), d, p) \geq 0, \forall t \in [t_c, t_c + \tau],$$

де  $t_c$  означає поточний момент часу, а точне формулювання  $f(t, y(t), u(t))$  залежить від конкретного розглянутого випадку. Для моделі формулювання  $f(t, y(t), u(t))$  залежить від цільового стану та контрольних даних, що надходять від попереднього обчислення.

Таким чином, на основі запропонованої моделі розроблено математичне забезпечення комп'ютерної системи контролю процесу ферментації винних продуктів.

**Література**

1. Shea A. J. Comfort. Oxygen and fermentation. More Wine – Absolutely Everything! for Wine-Making, 2009.
2. R. David, D. Dochain, J.-R. Mouret, A. Vande Wouwer, and J.-M. Sablayrolles. Dynamical modeling of alcoholic fermentation and its link with nitrogen consumption. In Proceedings of the 11th International Symposium on Computer Applications in Biotechnology (CAB 2010), 2010, P. 496–501.
3. H. H. Dittrich and M. Großmann. Mikrobiologie des Weines. Ulmer, fourth edition, 2011, 288 p.

УДК 004.4

**О. Прокопюк, Н. Ромашевська, Я. Войтович, Р. Жаровський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **АВТОМАТИЗАЦІЯ ТА ОПТИМІЗАЦІЯ НАВЧАЛЬНОГО ПРОЦЕСУ В ВИЩИХ НАВЧАЛЬНИХ ЗАКЛАДАХ**

UDC 004.4

**O. Prokopyuk, N. Romashevskya, Ya. Voytovych, R. Zharovskyi**

## **AUTOMATION AND OPTIMIZATION OF THE EDUCATIONAL PROCESS IN HIGHER EDUCATIONAL INSTITUTIONS**

На сьогоднішній день проекти з віртуалізації ІТ-інфраструктури активно впроваджуються багатьма провідними компаніями. У процесі віртуалізації ІТ-інфраструктури створюється віртуальна інфраструктура, що має багато нових можливостей при збереженні існуючої схеми діяльності ІТ-ресурсів. Компанії-постачальники різних платформ віртуалізації готові надати інформацію про успішні проекти щодо впровадження віртуальної інфраструктури у великих банках, промислових компаніях, лікарнях, освітніх установах. Безліч переваг віртуалізації операційних систем дозволяють компаніям економити на обслуговуванні, персоналі, апаратному забезпеченні, забезпеченні безперебійної роботи.

Суть нашої концепції передбачає, що у кожному університеті чи коледжі має існувати розширюваний і масштабований центр обробки даних (ЦОД).

Переваги концепції:

- ЦОД є центром управління навчальним процесом університету;
- оновлення (upgrade) Hardware & Software проходить лише на ЦОДі, що суттєво економить витрати на комп'ютерне обладнання;
- у комп'ютерних класах використовуються тонкі (thin) та нульові (zero) клієнти;
- обчислювальна потужність ЦОД може використовуватися для наукових досліджень;
- використання віртуалізації збільшує ефективність використання комп'ютерної техніки з 10% до 70–80%;

Заняття відбуваються у комп'ютерних класах, ці класи обладнані тонкими чи нульовими клієнтами. Усі класи підключені до ЦОДу. Для кожної дисципліни створюються спеціальні «образи» для проведення занять з усіх дисциплін, що вивчаються. На перервах між заняттями оператор ЦОД завантажує необхідні «образи» для проведення заняття. Дані образи містять лише необхідне програмне забезпечення та додаткове програмне забезпечення, яке відстежує всі дії студента. Усі студенти отримують однотипні завдання. Відстеження програмного забезпечення після закінчення заняття створить звіт про всі дії студента, за допомогою якого викладач може об'єктивно оцінити студента.

Система для віртуалізації програм Microsoft SoftGrid забезпечує аналогічне «пісочниці» середовище під назвою SystemGuard, в якому програми можуть виконуватися на комп'ютерах користувачів без локальної установки операційної системи.

SystemGuard – віртуальне прикладне середовище, яке містить усі необхідні додатку елементи, такі як файли, дані реєстру, СОМ-об'єкти та інформація про середовище. Сумісні з SoftGrid програми не потрібно встановлювати в основній операційній системі, але вони взаємодіють з операційною системою відповідно до певних правил, щоб уникнути зайвого дублювання даних у віртуальному просторі.

УДК 004.4

**Н. Ромашевська, О. Прокопюк, Р. Жаровський**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ВИКОРИСТАННЯ ТЕХНОЛОГІЇ ВІРТУАЛІЗАЦІЇ У ПРОЦЕСІ НАВЧАННЯ СТУДЕНТІВ**

UDC 004.4

**N. Romashevskya, O. Prokopyuk, R. Zharovskyi**

### **AUTOMATION AND OPTIMIZATION OF THE EDUCATIONAL PROCESS IN HIGHER EDUCATIONAL INSTITUTIONS**

Метою даної роботи є визначити працездатність, стабільність роботи та час витрачений на відновлення працездатності сервера з технологією віртуалізації порівняно із звичайним використанням ПК у процесі навчання.

Проводився експеримент у двох аудиторіях для підготовки фахівців у галузі розробки та створення бази даних у SQL з усім необхідним програмним забезпеченням. В обох аудиторіях встановлюється стандартне програмне забезпечення для навчання фахівців: Мови програмування, Mathcad, AutoCAD, MS Office та інше. Єдиною відмінністю є саме те, що в одній аудиторії встановлюється сервер, налаштовується Active Directory, налаштування домену. Далі - розгортається VMware з 12 віртуальними машинами, на одній встановлюється SQL Server - для навчання розробників, після закінчення установки та налаштування - знімається образ для перекидання на решту 11 віртуальних машин та резервного відновлення на випадок непередбачених проблем.

В іншій аудиторії для навчання розробників – на кожному комп'ютері встановлюється VMware і на одній віртуальній машині встановлюється OS Linux, на якому ставиться SQL Server, налаштовується, після цього зберігається образ однієї віртуальної машини, що розгортається на інші 11 комп'ютерів.

Весь експеримент складається із трьох частин метою яких є:

- підготовка аудиторій, тобто встановлення та налаштування ПЗ.
- експериментальна перевірка працездатності та стабільної роботи двох аудиторій, а також фіксування технічних проблем та фіксування часу на їх усунення.
- підрахунок та обробка всіх отриманих даних.

В результаті проведених досліджень отримано наступні дані та результати.

Допрацьована та повністю налаштована доменна система разом з технологією віртуалізації на базі VMware дозволяє в 3 рази скоротити випадки технічних збоїв, а час усунення цих збоїв – у 5 разів.

Система проста у користуванні та зрозуміла. Проте політики безпеки та винятків потребують ретельного та уважного налаштування.

Система Windows Server 2008 R2 з технологією віртуалізації дозволяє дати доступ до свого власного віртуального сервера і дати повноцінну можливість налаштувати, зіпсувати і відновити її самостійно, без шкоди для інших учасників – викладач або системний адміністратор здатний відновити працездатність SQL Server протягом 5–10 хвилин (на відміну від звичайного методу 40 хвилин або більше).

Віртуалізація програм дозволила спростити розгортання, адміністрування, оновлення та видалення програм, що використовуються під час роботи та навчання.

Серверна віртуалізація дозволяє значно скоротити кількість фізичних серверів шляхом їхньої консолідації: розміщення декількох серверів на одному фізичному обладнанні. Консолідація досягається шляхом розміщення серверів в окремих віртуальних машинах, розташованих на одному фізичному сервері, та керованих гіпервізором.

УДК 621.317.39:578.087

**В. Тимошук, В. Карташов, Р. Королюк, Т. Рубен**

(Тернопільський національний технічний університет імені Івана Пулюя, Україна)

## **ОГЛЯД ПРОТОКОЛІВ КЕРУВАННЯ ДЛЯ ПОБУДОВИ АВТОМАТИЗОВАНИХ СИСТЕМ ВІДДАЛЕНОГО УПРАВЛІННЯ**

UDC 621.317.39:578.087

**V. Tymoshchuk, V. Kartashov, R. Koroliuk, T. Ruben**

## **OVERVIEW OF CONTROL PROTOCOLS FOR BUILDING AUTOMATED REMOTE CONTROL SYSTEMS**

Протокол мережевого обміну інформацією – це перелік типів і форматів переданих блоків даних, правила їх обробки і взаємодії термінальних комп'ютерів на одному рівні. Інтерфейсом називають набір правил, що визначають взаємодію сервісів сусідніх рівнів в одному терміналі.

CoAP – Constrained Application Protocol – це спеціалізований протокол додатків до Інтернету для обмежених пристроїв. Це дозволяє тим обмеженим пристроям, званим «вузлами», спілкуватися з більш широким Інтернетом, використовуючи подібні протоколи. CoAP призначений для використання між пристроями в одній і тій же обмеженій мережі (наприклад, малопотужними, втратними мережами), між пристроями та загальними вузлами в Інтернеті, і між пристроями в різних обмежених мережах, приєднаних до Інтернету. CoAP також використовується за допомогою інших механізмів, таких як SMS у мережах мобільного зв'язку.

Протокол CoAP призначений для взаємодії простих пристроїв, такими як клапани, вимикачі, датчики якими можна керувати або контролювати їхні покази віддалено через мережу Інтернет.

Протокол AMQP: Advanced Message Queuing Protocol (AMQP) – це відкритий протокол, для передачі повідомлень між компонентами системи з низькою затримкою і на досить високій швидкості. AMQP може налаштовується під потреби конкретного проекту і передбачає надійний транспортний протокол, такий як TCP.

Ідея цього протоколу полягає в обміні довільним чином повідомленнями через AMQP-брокер, який здійснює маршрутизацію, гарантує доставку, розподіл потоків даних, підписку на потрібні типи повідомлень.

Головною перевагою AMQP є функція зберігання, а також передачі даних, яка забезпечує надійність навіть якщо в роботі мережі стався збій. Недоліком протоколу AMQP є низький рівень успішності доставки при низькій пропускну здатності та збільшується при збільшенні пропускну здатності. А перевагою, порівнюючи AMQP з REST, протокол AMQP може відправляти більшу кількість повідомлень у секунду.

MQTT або Message Queue Telemetry Transport – це легкий, компактний і відкритий протокол обміну даними створений для передачі даних на віддалених локаціях, де потрібно невеликий розмір коду і є обмеження по пропускну здатності каналу.

Також існує версія протоколу MQTT-SN (MQTT for Sensor Networks), раніше відома як MQTT-S, яка призначена для вбудованих бездротових пристроїв без підтримки TCP / IP мереж, наприклад, Zigbee.

Основними особливостями протоколу повідомлень MQTT є:

- асинхронний протокол
- компактні повідомлення
- Робота в умовах нестабільного зв'язку на лінії передачі даних
- Підтримка декількох рівнів якості обслуговування (QoS)
- Легка інтеграція нових пристроїв

Протокол MQTT працює на прикладному рівні поверх TCP/IP і використовує за замовчуванням 1883 порт.

Обмін повідомленнями в протоколі MQTT здійснюється між клієнтом (client), який може бути видавцем або підписником (publisher/subscriber) повідомлень, і брокером (broker) повідомлень (наприклад, Mosquitto MQTT).

Видавець відправляє дані на MQTT брокер, вказуючи в повідомленні певну тему, топик (topic). Підписники можуть отримувати різні дані від безлічі видавців залежно від підписки на відповідні топіки.

Пристрої MQTT використовують певні типи повідомлень для взаємодії з брокером, нижче представлені основні:

Connect – встановити з'єднання з брокером

Disconnect – розірвати з'єднання з брокером

Publish – опублікувати дані в топик на брокера

Subscribe – підписатися на топик на брокера

Unsubscribe – відписатися від топіка

Схему взаємодії між підписником, видавцем і брокером можна переглянути на рисунку 1.



**Рисунок 1.** Схema взаємодії між підписником, видавцем і брокером

Видавець відправляє дані на MQTT брокер, вказуючи в повідомленні певну тему, топик (topic). Підписники можуть отримувати різні дані від безлічі видавців залежно від підписки на відповідні топіки.

Аналізуючи всі три види протоколів для використання їх для розробки навчальним стендів віддаленого керування на нашу думку необхідно обрати протокол MQTT, як найбільш гнучкий в даному плані.

### Література

1. Що таке MQTT і для чого він потрібен в IoT? Опис протоколу MQTT. URL: <http://edu.asu.in.ua/mod/book/tool/print/index.php?id=116>.
2. Протокол обмеженого застосування. URL: [https://en.wikipedia.org/wiki/Constrained\\_Application\\_Protocol](https://en.wikipedia.org/wiki/Constrained_Application_Protocol).
3. Сервоприводи. URL: <http://www.princeton.edu/~mae412/TEXT/NTRAK2002/292-302.pdf>.
4. Ардуіно. URL: <https://uk.wikipedia.org/wiki/Arduino>.



## ВИКОРИСТАННЯ ПНЕВМО-СТРУМЕНЕВИХ ЗАХОПЛЮВАЧІВ В АВТОМАТИЧНИХ СЕПАРАТОРАХ ЛИСТОВОГО МАТЕРІАЛУ

UDC 621.865

P. Fedoriv, I. Fedoriv

### THE USE OF PNEUMATIC-JET ATTACHERS IN AUTOMATIC SEPARATORS OF SHEET MATERIAL

Автоматичні сепаратори листового матеріалу найчастіше застосовуються в поліграфії при подачі заготовок. Поштучне відділення заготовки від стопи за допомогою вакуумних захоплювачів, які набули широкого поширення, потребує додаткових механізмів для здійснення зворотно-поступальних рухів для захоплення заготовок і подачі їх на подальше транспортування. Загальний час спрацювання таких механізмів визначатиметься часом спрацювання механізмів подачі захоплювачів до листового матеріалу, часом спрацювання пневматичного клапана та часом створення вакууму в камері захоплювача.

Для збільшення продуктивності роботи автоматичних сепараторів пропонується використовувати пневмо-струменеві захоплювачі (рис. 1), які не потребують механізмів підйому-опускання й здатні притягувати заготовки на певній відстані  $h$ . В основу розроблених конструкцій струменевих пристроїв подачі листового матеріалу покладений ефект виникнення присмоктуючої дії, що виражається у взаємодії витікаючого з отвору малого діаметра струменя стиснутого повітря з обтічною плоскою поверхнею листа [1].

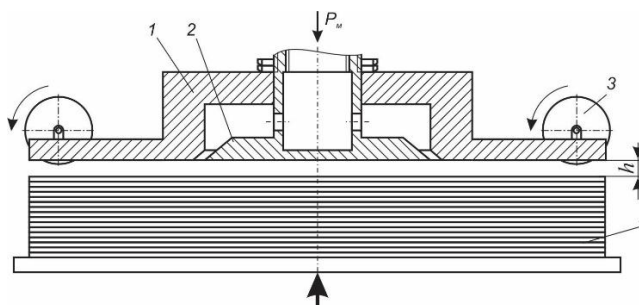


Рисунок 1. Сепаратор листового матеріалу

Час циклу струменево-фрикційних живильників неперервної подачі заготовок визначається сумою часу  $t_1$  руху верхньої заготовки від стопи до периферії фрикційних дисків 2 під дією присмоктуючої сили струменя, часу  $t_2$  транспортування заготовки фрикційними дисками до приймальних роликів і часу  $t_3$  транспортування роликками.

Неперервна поштучна видача листових заготовок забезпечується за допомогою періодичного відключення подачі стиснутого повітря з використанням елементів контролю заготовки, що транспортується у власній площині або за допомогою виконання умови  $t_1 > t_2 + t_3$  а також зміщення в сторону приймальних роликів відносно центру ваги результативної сили присмоктування на величину  $1/6 \dots 1/8$  її довжини.

#### Література

1. Проць Я. І., Федорів П. С., Цяпуга Ю. О., Скочилас В. В. Дослідження статичних характеристик струменево-фрикційних пристроїв подачі листового матеріалу. Поліграфія і видавнича справа. Львів: Українська академія друкарства, 2011. С. 93–97.

## ОГЛЯД РЕЗУЛЬТАТІВ ВИКОРИСТАННЯ ДАВАЧІВ ВОЛОГОСТІ, ВИГОТОВЛЕНИХ З КАРТОННИХ ТА ПАПЕРОВИХ ВІДХОДІВ

UDC 621.317.39:578.087

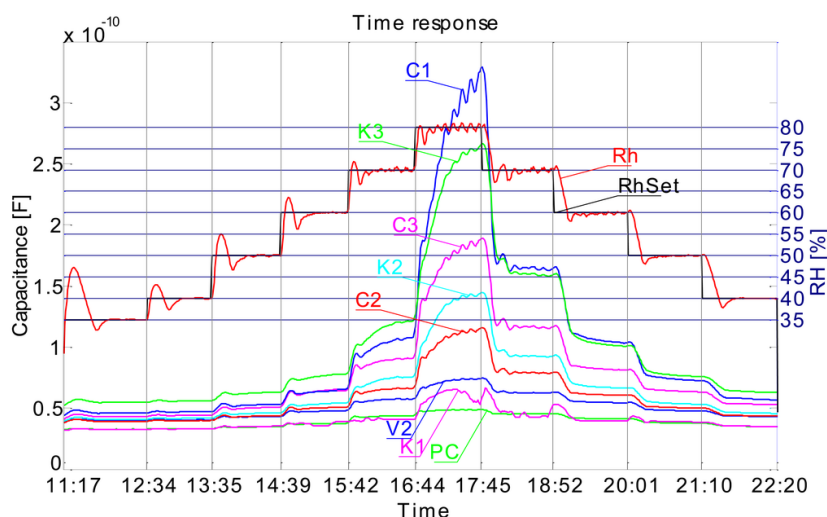
M. Shtyiuik, O. Yurchyk, O. Totosko, V. Levytskyi

## AN OVERVIEW OF THE RESULTS OF USING HUMIDITY SENSORS MADE FROM CARDBOARD AND PAPER WASTE

У роботі було проведено виготовлення, проектування та дослідження різних ємнісних датчиків вологості з трафаретним друком. Два типи ємнісних датчиків вологості були розроблені та виготовлені за допомогою трафаретного друку на переробленому папері та картоні. Такі давачі можуть використовуватись для оцінки вологості у різних технологічних процесах.

Як друкарську фарбу використовували комерційно доступні електропровідні фарби. Значна кількість робіт присвячена методам вимірювання вологості з використанням паперу як діелектрика. Ефективність різних конструкцій була перевірена у вологій камері. Відносну вологість у камері змінювали в діапазоні 35–80 % відносної вологості при постійній температурі 23°C. Цікавими параметрами були ємність і провідність кожного матеріалу датчика, а також довгострокова поведінка.

Отримані результати демонструють переважно логарифмічну реакцію паперових датчиків, за винятком датчиків на основі картону. Датчики на основі переробленого паперу демонструють зміну значення на три порядки величини, тоді як датчики на основі картону мають зміну значення на кілька 10 секунд у всьому діапазоні відносної вологості



**Рисунок 1.** Час відгуку гребінчастих давачів на різних матеріалах (перероблений папір C1–C3, картон K1–K3, PC – полікарбонат 2 × 2 см, V2 – пакувальний папір для харчових продуктів, RhSet – задане значення камери вологості, Rh – виміряне значення камери вологості); Суфікс давачів: 1–1 × 1 см, 2–2 × 2 см, 3–3 × 3 см.

### Література

1. Mraović M., Muck T., Pivar M., Trontelj J., Pleteršek A. Humidity Sensors Printed on Recycled Paper and Cardboard. *Sensors*. 2014, 14, 13628–13643.

## ЗМІСТ

### СЕКЦІЯ 1. МАТЕМАТИЧНЕ МОДЕЛЮВАННЯ

**А. Кашосі, О. Кишкевич, Н. Загородна**

УПРАВЛІННЯ ЯКІСТЮ ДАНИХ В ПРОЦЕСІ ЕТЛ В УМОВАХ РЕСУРСНИХ ОБМЕЖЕНЬ

**A. Kashosi, O. Kyshkevych, Zagorodna Nataliya**

DATA QUALITY MANAGEMENT IN ETL PROCESS UNDER RESOURCE CONSTRAINTS

3

**В. Пісьціо, І. Белякова, В. Медвідь**

ОПТИМІЗАЦІЯ ФОРМИ П'ЄЗОЕЛЕКТРИЧНОГО ТРАНСФОРМАТОРА

**Vadim Piscio, Iryna Belyakova, Volodymyr Medvid**

SHAPE OPTIMIZATION OF PIEZOELECTRIC TRANSFORMER

6

**М. Фриз, Б. Млинко**

БАГАТОВИМІРНІ УМОВНІ ЛІНІЙНІ ВИПАДКОВІ ПРОЦЕСИ

**Mykhailo Fryz, Bogdana Mlynko**

MULTIVARIATE CONDITIONAL LINEAR RANDOM PROCESSES

8

### СЕКЦІЯ 2. ІНФОРМАЦІЙНІ СИСТЕМИ ТА ТЕХНОЛОГІЇ, КІБЕРБЕЗПЕКА

**А. Анпілогов**

ДОДАТКОВІ ЗАСОБИ ЗАХИСТУ БАЗИ МЕТАДАНИХ РЕЄСТРУ ІНФОРМАЦІЙНИХ РЕСУРСІВ

**A. Anpilohov**

ADDITIONAL MEANS OF PROTECTION OF THE METADATA BASE OF THE REGISTER OF INFORMATION RESOURCES

9

**О. Багрій**

ОСОБЛИВОСТІ РЕАЛІЗАЦІЇ ЗБОРУ ТА ОПРАЦЮВАННЯ ДАНИХ ДЛЯ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ НА ОСНОВІ КЛАВАТУРНОГО ПОЧЕРКУ

**O. Bagriy**

FEATURES OF IMPLEMENTATION OF DATA COLLECTION AND PROCESSING FOR KEYBOARD-BASED USER AUTHENTICATION

10

**О. Багрій**

АНАЛІЗ ЗАСОБІВ АУТЕНТИФІКАЦІЇ КОРИСТУВАЧІВ НА ОСНОВІ КЛАВАТУРНОГО ПОЧЕРКУ

**O. Bagriy**

ANALYSIS OF KEYBOARD-BASED USER AUTHENTICATION MEANS

12

**Т. Базан**

АНАЛІЗ ВХІДНИХ ДАНИХ СИСТЕМИ ПРОГНОЗУВАННЯ ФІНАНСОВОЇ РЕНТАБЕЛЬНОСТІ ПІДПРИЄМСТВА

**T. Bazan**

ANALYSIS OF INPUT DATA OF THE SYSTEM FOR FORECASTING THE FINANCIAL PROFITABILITY OF THE ENTERPRISE

14

**К. Белоусов, Т. Маєвський**

РОЛЬ ТА ЗНАЧЕННЯ ВЕЛИКИХ ДАНИХ В СУЧАНИХ НАУКОВИХ ДОСЛІДЖЕННЯХ

**K. Bielousov, T. Maievskiy**

THE BIG DATA ROLE AND SIGNIFICANCE IN MODERN SCIENTIFIC RESEARCH

15

**К. Белоусов, Т. Маєвський**

ВЕЛИКІ ДАНІ ТА АНАЛІТИЧНЕ ОПРАЦЮВАННЯ В НАУКОВИХ ДОСЛІДЖЕННЯХ

**K. Bielousov, T. Maievskiy**

BIG DATA AND ANALYTICAL PROCESSING IN SCIENTIFIC RESEARCH

16

<b>А. Блавіцький, С. Мацюк, С. Криськова</b> ОЦІНКА РОЗВИТКУ БЕЗПЕКИ ОПЛАТИ ПЛАТІЖНИМИ КАРТКАМИ <b>A. Blavitskyi, S. Matsiuk, S. Kryskova</b> ASSESSMENT OF THE SECURITY DEVELOPMENT OF PAYMENT CARDS	17
<b>А. Буковська</b> ПАРАЛЕЛЬНЕ ТА РОЗПОДІЛЕНЕ ГЕНЕРУВАННЯ POWERSET З ВИКОРИСТАННЯМ ПЛАТФОРМИ ОБРОБКИ ВЕЛИКИХ ДАНИХ <b>A. Bukovska</b> PARALLEL AND DISTRIBUTED POWERSET GENERATION USING A BIG DATA PLATFORM	18
<b>В. Василенко, Н. Стадник</b> ВИКОРИСТАННЯ СТАКУ ELK ДЛЯ ДОСЛІДЖЕННЯ ПОДІЙ <b>V. Vasylenko, N. Stadnyk</b> USING ELK STACK TO RESEARCH OF EVENTS	20
<b>В. Василенко, Н. Стадник</b> ЛОГУВАННЯ – ЩО ЦЕ І В ЧОМУ ЙОГО КОРИСТЬ <b>V. Vasylenko, N. Stadnyk</b> LOGGING – WHAT IS IT AND WHAT IS ITS BENEFIT	21
<b>Р. Волошин</b> АУДИТ БЕЗПЕКИ AMAZON SELLING PARTNER API <b>R. Voloshyn</b> AMAZON SELLING PARTNER API CYBERSECURITY AUDIT	22
<b>І. Воробець</b> ПОРІВНЯННЯ МЕТОДІВ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ <b>I. Vorobets</b> COMPARISON OF TIME SERIES FORECASTING METHODS	23
<b>М. Гаврилов</b> ПОВТОРНА ІДЕНТИФІКАЦІЯ ЛЮДЕЙ ЗА ФОТО ТА ВІДЕО ЗАСОБАМИ COMPUTER VISION <b>M. Havrylov</b> RE-IDENTIFICATION OF PEOPLE FROM PHOTOS AND VIDEOS BY MEANS OF COMPUTER VISION	24
<b>О. Голинська, Я. Мудрик</b> РОЛЬ CRM-СИСТЕМИ У СУЧАСНИХ БІЗНЕС-ПРОЦЕСАХ <b>O. Holynska, Lecturer,</b> ROLE OF CRM SYSTEM IN MODERN BUSINESS PROCESSES	25
<b>В. Грицюк, М. Стадник</b> КЛАСТЕРИЗАЦІЯ СПАМ-ДОМЕНІВ МЕТОДАМИ МАШИННОГО НАВЧАННЯ <b>V. Hrytsiuk, M. Stadnyk</b> SPAM DOMAINS CLUSTERIZATION BY USING MACHINE LEARNING METHODS	26
<b>Н. Зарічний, Є. Тиш</b> АВТОМАТИЗАЦІЯ ТЕСТУВАННЯ МОБІЛЬНИХ ДОДАТКІВ ЗА ТЕХНОЛОГІЄЮ AGILE <b>N. Zarichnyi, Ye. Tysh, Ph.D.</b> AUTOMATION OF MOBILE APPLICATION TESTING USING AGILE TECHNOLOGY	27
<b>О. Кравчук</b> ВИЗНАЧЕННЯ ПОГОДНИХ УМОВ У TELEGRAM <b>O. Kravchuk</b> DETERMINATION OF WEATHER CONDITIONS IN TELEGRAM	28

<b>О. Кравчук</b> РОЗРОБКА ТЕЛЕГРАМ БОТІВ НА PYTHON	
<b>O. Kravchuk</b> DEVELOPMENT OF TELEGRAM BOTS IN PYTHON	29
<b>Н. Лісовий, А. Ставицька, А. Гіжовський</b> АНАЛІТИЧНЕ ОПРАЦЮВАННЯ ВЕЛИКИХ ЗА ОБСЯГОМ ДАНИХ	
<b>N. Lisovyi, A. Stavytska, A. Hzhovskyi</b> LARGE DATA VOLUMES ANALYTICAL PROCESSING	30
<b>Н. Шаблій, П. Марценюк</b> СИСТЕМИ МОНИТОРИНГУ СТАНУ ДОВКІЛЛЯ	
<b>N. Shabliy, P. Martseniuk</b> ENVIRONMENTAL STATE MONITORING SYSTEMS	31
<b>Р. Маслій</b> СИСТЕМА БЕЗПЕКИ ДЛЯ ІОТ З ВИКОРИСТАННЯМ SIEM ТЕХНОЛОГІЙ	
<b>R. Maslii</b> SECURITY SYSTEM FOR IOT USING SIEM TECHNOLOGIES	32
<b>А. Блавицький, С. Мацюк, С. Криськова</b> ЖИТТЄВИЙ ЦИКЛ ПЛАТЕЖУ	
<b>A. Blavitskyi, S. Matsiuk, S. Kryskova</b> PAYMENT LIFE CYCLE	33
<b>М. Мокрицький, Ю. Скоренький</b> ДОСЛІДЖЕННЯ ВРАЗЛИВОСТЕЙ НЕЙРОІНТЕРФЕЙСІВ	
<b>M. Mokrytskyi, Yu. Skorenkyu</b> STUDY OF BRAIN-COMPUTER INTERFACES VULNERABILITY	34
<b>Г. Мушинська, Л. Дмитроца</b> АНАЛІТИКА ОПТИМІЗАЦІЇ ЧАТ-БОТА	
<b>H. Mushynska, L. Dmytrotsa</b> CHAT BOT OPTIMIZATION ANALYTICS	35
<b>К. Николин</b> РОЗВІДКА ВІДКРИТИХ ДЖЕРЕЛ ІНФОРМАЦІЇ ДЛЯ ВИЯВЛЕННЯ ЗАГРОЗ БЕЗПЕКИ БІЗНЕСУ	
<b>K. Nykolyn</b> OPEN SOURCE INTELLIGENCE FOR IDENTIFYING BUSINESS SECURITY THREATS	36
<b>Т. Патральський</b> ТРАНСФОРМАЦІЯ ДАНИХ У НАСТРОЮВАНІ ІНФОРМАЦІЙНІ ЗВІТИ ТА ІНФОРМАЦІЙНІ ПАНЕЛІ LOOKER STUDIO	
<b>T. Patralskyi</b> DATA TRANSFORMATION INTO CUSTOMIZABLE INFORMATION REPORTS AND INFORMATION PANELS LOOKER STUDIO	37
<b>Ю. Петришин</b> СИСТЕМИ МЕНЕДЖМЕНТУ, МОДЕЛЬ ISO 27001	
<b>Yu. Petryshyn</b> MANAGEMENT SYSTEMS, ISO 27001 MODEL	38
<b>П. Прийма, А. Зав'ялова, В. Дуда</b> ІНТЕРНЕТ РЕЧЕЙ, «ВЕЛИКІ ДАНІ» ТА АНАЛІТИКА. СТАН ТА ПЕРСПЕКТИВИ ДОСЛІДЖЕНЬ	
<b>P. Pryima, A. Zavialova, V. Duda</b> THE INTERNET OF THINGS, BIG DATA AND ANALYTICS. RESEARCH STATUS AND PROSPECTS	39
<b>П. Прийма, А. Зав'ялова, В. Дуда</b> ІНСТРУМЕНТИ АНАЛІТИЧНОГО ОПРАЦЮВАННЯ «ВЕЛИКИХ ДАНИХ»	
<b>P. Pryima, A. Zavialova, V. Duda</b> TOOLS FOR BIG DATA ANALYTICAL PROCESSING	40

<b>І. Ралік</b> ЗАДАЧА РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ОБЛІКУ РЕАЛІЗАЦІЇ ТОВАРІВ В ТОРГІВЛІ	
<b>I. Ralik</b> THE TASK OF SOFTWARE DEVELOPMENT FOR THE GOODS SALE ACCOUNTING IN RETAIL	41
<b>О. Ревнюк</b> ЯКІСТЬ УПРАВЛІННЯ ІНФОРМАЦІЙНОЮ БЕЗПЕКОЮ ОРГАНІЗАЦІЙ	
<b>Revnuk O.</b> THE QUALITY OF INFORMATION SECURITY MANAGEMENT OF ORGANIZATIONS	42
<b>А. Романець, Г. Козбур</b> ПРОБЛЕМИ АУТЕНТИФІКАЦІЇ АКАУНТІВ У СОЦМЕРЕЖАХ	
<b>A. Romanets, G. Kozbur</b> ACCOUNT AUTHENTICATION PROBLEMS IN SOCIAL NETWORKS	44
<b>А. Романець, Г. Козбур</b> БЕЗПЕКА СОЦМЕРЕЖІ ПІД ЧАС АУТЕНТИФІКАЦІЇ КОРИСТУВАЧА	
<b>A. Romanets, G. Kozbur</b> SOCIAL NETWORK SECURITY DURING USER AUTHENTICATION	45
<b>Ю. Северіна</b> ІНФОРМАЦІЙНІ СИСТЕМИ В ТУРИЗМІ	
<b>Yu. Severina</b> INFORMATION SYSTEMS IN TOURISM	46
<b>В. Семенюк</b> ПОЄДНАННЯ ТЕХНОЛОГІЇ ДОПОВНЕНОЇ РЕАЛЬНОСТІ ТА ФАКТОГРАФІЧНОГО ПОШУКУ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДЛЯ КОНСОЛІДАЦІЇ СОЦІО-КОМУНІКАЦІЙНИХ РЕСУРСІВ «РОЗУМНОГО МІСТА» В МУЗЕЙНІЙ ДІЯЛЬНОСТІ	
<b>V. Semeniuk</b> COMBINATION OF AUGMENTED REALITY TECHNOLOGY AND FACTOGRAPHICAL SEARCH OF THE INFORMATION SYSTEM FOR THE CONSOLIDATION OF SOCIO-COMMUNICATION RESOURCES OF THE SMART CITY IN MUSEUM ACTIVITIES	47
<b>С. Сербичанський</b> ДОСЛІДЖЕННЯ ВИМОГ ДО ФІЗИЧНОГО ТА ПРОГРАМНОГО ЗАХИСТУ ІНФОРМАЦІЇ НА ОБ'ЄКТАХ КРИТИЧНОЇ ІНФРАСТРУКТУРИ В УМОВАХ ЗАГРОЗ І ОБМЕЖЕНЬ	
<b>S. Serbychanskyi</b> STUDY OF REQUIREMENTS FOR PHYSICAL AND SOFTWARE PROTECTION OF INFORMATION AT CRITICAL INFRASTRUCTURE OBJECTS UNDER THE CONDITIONS OF THREATS AND LIMITATIONS	48
<b>В. Сербін</b> РЕАЛІЗАЦІЯ ПАРАЛЕЛЬНОЇ ОБРОБКИ ДАНИХ В МОВІ ПРОГРАМУВАННЯ JAVASCRIPT	
<b>V. Serbin</b> IMPLEMENTATION OF PARALLEL DATA PROCESSING IN JAVASCRIPT PROGRAMMING LANGUAGE	49
<b>О. Сороківський, Я. Литвиненко</b> ПОРІВНЯННЯ ПРЕТРЕНОВАНИХ МОДЕЛЕЙ ДЛЯ ДЕТЕКЦІЇ ОБ'ЄКТІВ	
<b>O. Sorokivskyi, I. Lytvynenko</b> COMPARATION OF THE PRETRAINED MODELS FOR OBJECT DETECTION	51

<b>А. Станько</b> АНАЛІЗ КОНЦЕПЦІЇ ВСЕОСЯЖНОГО ІНТЕРНЕТУ – ІоЕ	
<b>A. Stanko</b> ANALYSIS OF THE CONCEPT OF THE INTERNET OF EVERYTHING – ІоЕ	53
<b>М. Турчиняк</b> ТЕХНОЛОГІЇ ВПЛИВУ СОЦІАЛЬНИХ МЕРЕЖ НА ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ	
<b>M. Turchyniak</b> TECHNOLOGIES OF THE INFLUENCE OF SOCIAL NETWORKS ON ENSURING INFORMATION SECURITY	55
<b>Д. Урбан</b> АНАЛІЗ ЗАГРОЗ КОМП'ЮТЕРНИХ СИСТЕМ	
<b>D. Urban</b> ANALYSIS OF COMPUTER SYSTEM THREATS	57
<b>А. Хом'як</b> СИГНАЛИ ГОЛОВНОГО МОЗКУ, ЯКІ МОЖНА ОТРИМАТИ НЕІНВАЗИВНИМИ МЕТОДАМИ	
<b>A. Khomiak</b> BRAIN SIGNALS OBTAINABLE VIA NON-INVASIVE IMAGING	58
<b>Г. Шимчук, О. Голотенко, Р. Золотий</b> ОСНОВНІ ПРОБЛЕМИ ТА ЗАГРОЗИ ХМАРНОЇ БЕЗПЕКИ	
<b>G. Shymchuk, O. Holotenko, R. Zoloty</b> USE THE MAIN PROBLEMS AND THREATS OF CLOUD SECURITY	59
<b>А. Мачужак</b> АВТОМАТИЗАЦІЯ ЗАДАЧ ТЕСТУВАННЯ ТА РОЗГОРТАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	
<b>A. Machuzhak</b> AUTOMATION OF SOFTWARE TESTING AND DEPLOYMENT TASKS	61
<b>СЕКЦІЯ 3. КОМП'ЮТЕРНІ СИСТЕМИ ТА МЕРЕЖІ</b>	
<b>М. Домарецький</b> ОГЛЯД СИСТЕМ ДЛЯ РОЗПІЗНАВАННЯ ЖЕСТИВ	
<b>M. Domaretskyi</b> REVIEW OF GESTURE RECOGNITION SYSTEMS	62
<b>А. Луцків, С. Баран</b> ТЕХНОЛОГІЇ НЕІНВАЗИВНОГО ВИМІРЮВАННЯ РІВНЯ ГЛЮКОЗИ В КРОВІ	
<b>A. Lutskiv, S. Baran</b> TECHNOLOGIES OF NON-INVASIVE GLUCOSE LEVEL MEASUREMENT IN BLOOD	63
<b>А. Луцків, С. Баран</b> АЛГОРИТМИ МАШИННОГО НАВЧАННЯ ДЛЯ ПРОГНОЗУВАННЯ РІВНЯ ГЛЮКОЗИ В КРОВІ	
<b>A. Lutskiv, S. Baran</b> MACHINE LEARNING ALGORITHMS FOR PREDICTING THE LEVEL OF GLUCOSE IN THE BLOOD	64
<b>А. Луцків, М. Бондаренко</b> ОСОБЛИВОСТІ ЗАДАЧ І ФУНКЦІЙ DEVOPS ФАХІВЦІВ	
<b>A. Lutskiv, M. Bondarenko</b> FEATURES OF TASKS AND FUNCTIONS OF DEVOPS SPECIALISTS	65

<b>А. Луцків, І. Барна</b> АНАЛІЗ СЕРВІСНО-ОРІЄНТОВАНОЇ АРХІТЕКТУРИ В ПРОЦЕСІ ЗАСТОСУВАННЯ DEVOPS ПРАКТИК	
<b>A. Lutskiv, I. Barna</b> ANALYSIS OF SERVICE-ORIENTED ARCHITECTURE IN THE PROCESS OF APPLICATION OF DEVOPS PRACTICES	66
<b>А. Луцків, М. Бондаренко</b> ОСОБЛИВОСТІ ОПТИМІЗАЦІЇ СИСТЕМ ПІДТРИМКИ КОРИСТУВАЧІВ ІЗ ЗАСТОСУВАННЯМ ПІДХОДУ СИСТЕМ МАСОВОГО ОБСЛУГОВУВАННЯ	
<b>A. Lutskiv, M. Bondarenko</b> FEATURES OF USER SUPPORT SYSTEMS OPTIMIZATION USING THE APPROACH OF MASS SERVICE SYSTEMS	67
<b>А. Луцків, М. Бондаренко</b> АРХІТЕКТУРА СИСТЕМИ ПІДТРИМКИ З ОПТИМІЗАЦІЄЮ ПРОЦЕСУ УПРАВЛІННЯ ЗВЕРНЕННЯМИ КОРИСТУВАЧІВ	
<b>A. Lutskiv, M. Bondarenko</b> SUPPORT SYSTEM ARCHITECTURE WITH OPTIMIZATION OF THE USER COMPLAINT MANAGEMENT PROCESS	68
<b>В. Яцишин, Т. Кобець</b> ТЕХНОЛОГІЯ MESH В КОМП'ЮТЕРНИХ СИСТЕМАХ ПЕРЕДАЧІ ДАНИХ	
<b>V. Yatsyshyn, T. Kobets</b> TECHNOLOGIES OF NON-INVASIVE GLUCOSE LEVEL MEASUREMENT IN BLOOD	69
<b>В. Яцишин, Т. Кобець</b> МЕТОДИ ВИБОРУ ОПТИМАЛЬНИХ КОМПОНЕНТІВ КОМП'ЮТЕРНИХ СИСТЕМ НА ОСНОВІ АНАЛІЗУ ІЄРАРХІЇ	
<b>V. Yatsyshyn, T. Kobets</b> METHODS OF SELECTING OPTIMUM COMPUTER SYSTEM COMPONENTS BASED ON HIERARCHY ANALYSIS	70
<b>А. Луцків, М. Тимошук</b> ВАЖЛИВІСТЬ ДОКУМЕНТУВАННЯ В ПРОЦЕСІ ВДОСКОНАЛЕННЯ КОМП'ЮТЕРНИХ СИСТЕМ	
<b>A. Lutskiv, M. Tymoshchuk</b> THE IMPORTANCE OF THE DOCUMENTATION IN THE PROCESS OF IMPROVING COMPUTER SYSTEMS	71
<b>А. Луцків, М. Тимошук</b> МОДЕЛІ ПРЕДСТАВЛЕННЯ ДОКУМЕНТАЦІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПРИ ВДОСКОНАЛЕННІ КОМП'ЮТЕРНИХ СИСТЕМ	
<b>A. Lutskiv, M. Tymoshchuk</b> MODELS OF SOFTWARE DOCUMENTATION VIEW IN THE IMPROVEMENT OF COMPUTER SYSTEMS	72
<b>І. Головатий</b> ОБРОБКА ЗОБРАЖЕНЬ ІЗ ВИКОРИСТАННЯМ ГЕНЕТИЧНОГО АЛГОРИТМУ	
<b>I. Holovaty</b> IMAGE PROCESSING USING GENETIC ALGORITHM	73
<b>Ю. Гук, А. Паламар</b> МЕТОД АДАПТИВНОГО РЕГУЛЮВАННЯ ДОРОЖНЬОГО РУХУ НА ПЕРЕХРЕСТІ НА ОСНОВІ ІНТЕРНЕТУ РЕЧЕЙ	
<b>Y. Huk, A. Palamar</b> METHOD OF ADAPTIVE TRAFFIC CONTROL AT AN INTERSECTION BASED ON INTERNET OF THINGS	74



<b>О. Гуменюк</b> АНАЛІЗ РОБОТИ ІНСТРУМЕНТУ ДЛЯ УПРАВЛІННЯ ТА АНАЛІЗУ ЖУРНАЛІВ GRAYLOG	
<b>О. Humeniuk</b> PERFORMANCE ANALYSIS OF THE GRAYLOG LOG MANAGEMENT AND ANALYSIS TOOL	76
<b>О. Гуменюк</b> АНАЛІЗ РОБОТИ СТАНДАРТУ ЖУРНАЛЮВАННЯ SYSLOG	
<b>О. Humeniuk</b> ANALYSIS OF THE OPERATION OF THE SYSLOG JOURNALING STANDARD	77
<b>А. Лупенко, С. Куліков, Д. Денисов</b> КЛАСИФІКАЦІЯ ТА ОСОБЛИВОСТІ ЗАСТОСУВАННЯ ПРИКЛАДНИХ ПРОГРАМНИХ ІНТЕРФЕЙСІВ ПРИ РЕАЛІЗАЦІЇ КОМП'ЮТЕРНИХ СИСТЕМ	
<b>A. Lupenko, S. Kulikov, D. Denysov</b> CLASSIFICATION AND FEATURES OF THE APPLICATION PROGRAMMING INTERFACES IN COMPUTER SYSTEMS IMPLEMENTATION	79
<b>В. Яцишин, Н. Шаблій, Д. Денисов</b> ПРИЗНАЧЕННЯ І ДОЦІЛЬНІСТЬ ВИКОРИСТАННЯ API GATEWAY У КОМП'ЮТЕРНИХ СИСТЕМАХ	
<b>V. Yatsyshyn, N. Shabliy, D. Denysov</b> PURPOSE AND FEASIBILITY OF USING API GATEWAY IN COMPUTER SYSTEMS	80
<b>В. Яцишин, Н. Шаблій, І. Дишкант</b> ПРОЦЕС ФОРМУВАННЯ ПРОГРАМНИХ КОМПОНЕНТІВ ПОВТОРНОГО ВИКОРИСТАННЯ ПРИ РЕАЛІЗАЦІЇ КОМП'ЮТЕРНИХ СИСТЕМ	
<b>V. Yatsyshyn, N. Shabliy, I. Dyshkant</b> THE PROCESS OF FORMING REUSABLE SOFTWARE COMPONENTS IN THE IMPLEMENTATION OF COMPUTER SYSTEMS	81
<b>В. Яцишин, І. Дишкант</b> АРХІТЕКТУРА ЗАСОБУ ПІДТРИМКИ ПРОЦЕСУ ОЦІНЮВАННЯ ПОТЕНЦІЙНИХ КОМПОНЕНТІВ ПОВТОРНОГО ВИКОРИСТАННЯ	
<b>V. Yatsyshyn, PhD; Assoc. Prof., I. Dyshkant</b> ARCHITECTURE OF THE SUPPORT TOOL FOR THE EVALUATION OF POTENTIAL REUSE COMPONENTS	82
<b>А. Паламар, В. Дьомін, В. Волоський</b> ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ ДЛЯ МОНІТОРИНГУ СТАНУ ПРИБОРІВ БЕЗПЕРЕБІЙНОГО ЖИВЛЕННЯ	
<b>A. Palamar, V. Domin, V. Voloskyi</b> COMPUTER SYSTEM SOFTWARE FOR CONDITION MONITORING OF UNINTERRUPTIBLE POWER SUPPLY DEVICES	83
<b>А. Паламар, В. Дьомін</b> СТРУКТУРА МОДУЛЯ ДЛЯ МОНІТОРИНГУ СТАНУ ПРИБОРІВ БЕЗПЕРЕБІЙНОГО ЖИВЛЕННЯ	
<b>A. Palamar, V. Domin</b> MODULE STRUCTURE FOR CONDITION MONITORING OF UNINTERRUPTIBLE POWER SUPPLY DEVICE	84
<b>А. Паламар, І. Купратий</b> СИСТЕМА ДЛЯ ДИСТАНЦІЙНОГО МОНІТОРИНГУ СТАНУ ЗДОРОВ'Я ПАЦІЄНТІВ НА ОСНОВІ ІНТЕРНЕТУ МЕДИЧНИХ РЕЧЕЙ	
<b>A Palamar, I. Kupratyi</b> PATIENT HEALTH REMOTE MONITORING SYSTEM BASED ON INTERNET OF MEDICAL THINGS	85

<b>В. Ліщина, Р. Жаровський</b> МЕТОДИ ПІДВИЩЕННЯ ПРОПУСКНОЇ ЗДАТНОСТІ В МЕРЕЖАХ LTE <b>V. Lishchyna, R. Zharovskyi</b> METHODS OF INCREASE BANDWIDTH IN LTE NETWORKS	86
<b>О. Марчук</b> МЕТОД ІДЕНТИФІКАЦІЇ ДОРОЖНІХ ЗНАКІВ НА ОСНОВІ ЗГОРТКОВОЇ НЕЙРОМЕРЕЖІ <b>O. Marchuk</b> ROAD SIGN IDENTIFICATION METHOD BASED ON A CONVULSIONAL NEURAL NETWORK	87
<b>І. Мудрий</b> ЛОКАЛІЗАЦІЯ ТА КЛАСИФІКАЦІЯ ОБ'ЄКТІВ НА ЗОБРАЖЕННІ <b>I. Mudryi</b> LOCATION AND CLASSIFICATION OF IMAGE OBJECTS	88
<b>Т. Патральський</b> ЗБЕРІГАННЯ ТА ТРАНСФОРМАЦІЯ ДАНИХ У ХМАРНОМУ СЕРЕДОВИЩІ GOOGLE CLOUD BIGQUERY <b>T. Patralskyi</b> DATA STORAGE AND TRANSFORMATION IN THE CLOUD ENVIRONMENT GOOGLE CLOUD BIGQUERY	89
<b>В. Савчук, Н. Луцик</b> АНАЛІЗ ІСНУЮЧИХ СИСТЕМ КЛІМАТ-КОНТРОЛЮ <b>V. Savchuk, N. Lutsyk</b> ANALYSIS OF EXISTING CLIMATE CONTROL SYSTEMS	90
<b>В. Савчук, Н. Луцик</b> РОЗРОБКА СИСТЕМИ КЛІМАТ-КОНТРОЛЮ НА БАЗІ МІКРОКОНТРОЛЕРА ТА СЕНСОРІВ <b>V. Savchuk, N. Lutsyk</b> DEVELOPMENT OF THE CLIMATE CONTROL SYSTEM BASED ON THE MICROCONTROLLER AND SENSORS	91
<b>С. Свергун, Р. Жаровський</b> ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ПОБУДОВАНОГО НА МІКРОСЕРВІСНІЙ АРХІТЕКТУРІ <b>S. Svergun, R. Zharovskyi</b> TESTING OF SOFTWARE BUILT ON MICROSERVICE ARCHITECTURE	92
<b>С. Свергун, Р. Жаровський</b> ТЕСТУВАННЯ ПРОГРАМНОГО ПРОДУКТУ, ПОБУДОВАНОГО НА МІКРОСЕРВІСНІЙ АРХІТЕКТУРІ НА ОСНОВІ BDD <b>S. Svergun, R. Zharovskyi</b> TESTING OF SOFTWARE PRODUCT BUILT ON MICROSERVICE ARCHITECTURE BASED ON BDD	93
<b>І. Слюз, Р. Жаровський</b> ПРИНЦИПИ ТА ОСНОВНІ ЕТАПИ КОМПЛЕКСНОГО ТЕСТУВАННЯ КОМП'ЮТЕРНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ <b>I. Slyuz, R. Zharovskyi</b> PRINCIPLES AND MAIN STAGES OF COMPLEX TESTING OF A COMPUTER INFORMATION SYSTEM	94

<b>В. Тимошук, Д. Тимошук</b> ВІРТУАЛІЗАЦІЯ В ЦЕНТРАХ ОБРОБКИ ДАНИХ - АСПЕКТИ ВІДМОВСТІЙКОСТІ	
<b>V. Tymoshchuk, D. Tymoshchuk</b> VIRTUALIZATION IN DATA CENTERS – ASPECTS OF FAILURE TOLERANCE	95
<b>В. Яцишин, Б. Харитон</b> АРХІТЕКТУРА СИСТЕМИ ПІДТРИМКИ ПРОЦЕСІВ ВІЯВЛЕННЯ ТА ОЦІНЮВАННЯ ВРАЗЛИВОСТЕЙ ВЕБ-СЕРВЕРІВ У КОМП'ЮТЕРНИХ СИСТЕМАХ	
<b>V. Yatsyshyn, B. Kharyton</b> ARCHITECTURE OF THE SUPPORT SYSTEM FOR THE DETECTION AND ASSESSMENT OF WEB SERVER VULNERABILITY PROCESSES IN COMPUTER SYSTEMS	96
<b>В. Яцишин, В. Цимбалістий, Вік. Яцишин</b> КОМП'ЮТЕРНІ ІГРИ ЯК СПОСІБ МОДЕЛЮВАННЯ ПОВЕДІНКИ РЕАЛЬНИХ КОМП'ЮТЕРНИХ СИСТЕМ	
<b>Vas. Yatsyshyn, V. Tsymbalistyi, Vik. Yatsyshyn</b> COMPUTER GAMES AS A WAY OF REAL COMPUTER SYSTEMS BEHAVIOUR MODELLING	97
<b>В. Шаварський, Є. Тиш</b> ОСНОВНІ ПОНЯТТЯ СИСТЕМ ПЕРЕТВОРЮВАЧІВ СОНЯЧНОЇ ЕНЕРГІЇ	
<b>V. Shavarskiy, Ie. Tysh</b> BASIC CONCEPTS OF SOLAR ENERGY CONVERTER SYSTEMS	98
<b>В. Шаварський, Є.Тиш</b> ОСОБЛИВОСТІ РОЗРОБКИ ОДНОВІСНОГО СОНЯЧНОГО ТРЕКЕРА	
<b>V. Shavarskiy, Ie. Tysh</b> FEATURES OF THE DEVELOPMENT OF A SINGLE-AXIS SOLAR TRACKER	99
<b>В. Яцишин, Б. Харитон</b> СХЕМА РЕЛЯЦІЙНОЇ БАЗИ ДАНИХ ДЛЯ ЗБЕРІГАННЯ ТА ОПРАЦЮВАННЯ ВРАЗЛИВОСТЕЙ ВЕБ-СЕРВЕРІВ У РОЗУМНИХ КОМП'ЮТЕРНИХ СИСТЕМАХ	
<b>V. Yatsyshyn, B. Kharyton</b> A RELATIONAL DATABASE SCHEME FOR STORING AND PROCESSING WEB SERVER VULNERABILITIES IN SMART COMPUTER SYSTEMS	101
<b>Р. Ясінський, Г. Осухівська, А. Паламар</b> АПАРАТНО-ПРОГРАМНА СИСТЕМА ДЛЯ РЕГУЛЮВАННЯ МІКРОКЛІМАТУ ТЕПЛИЦЬ	
<b>R. Yasinskyi, H. Osukhivska, A. Palamar</b> HARDWARE AND SOFTWARE SYSTEM FOR GREENHOUSES MICROCLIMATE REGULATING	102
<b>СЕКЦІЯ 4. ПРОГРАМНА ІНЖЕНЕРІЯ ТА МОДЕЛЮВАННЯ СКЛАДНИХ РОЗПОДІЛЕНИХ СИСТЕМ</b>	
<b>А. Буй</b> ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ ВИРІШЕННЯ ПРОБЛЕМ ІЗ РЕАЛІЗАЦІЄЮ СІЛЬСЬКОГОСПОДАРСЬКОЇ ПРОДУКЦІЇ	
<b>A. Bui</b> INFORMATION SYSTEM FOR SOLVING PROBLEMS WITH SALE OF AGRICULTURAL PRODUCTS	103
<b>В. Волович, Б. Береженко, І. Боднарчук</b> ЗАДАЧА ПРОЄКТУВАННЯ ПРОГРАМНОЇ АРХІТЕКТУРИ В ПРОЦЕСАХ ЗАБЕЗПЕЧЕННЯ ЯКОСТІ	
<b>V. Volovych, B. Berezhenko, I. Bodnarchuk</b> THE PROBLEM OF SOFTWARE ARCHITECTURE DESIGN IN THE PROCESSES OF QUALITY ASSURANCE	104

<b>С. Глинянчук, І. Стадник</b> РОЗРОБКА WEB-ДОДАТКУ ДЛЯ РЕКОМЕНДАЦІЇ ІГОР	
<b>S. Hlynianchuk, I. Stadnyk</b> DEVELOPING WEB-APPLICATION FOR GAME RECOMMENDATION	107
<b>В. Гречаник</b> СИСТЕМАТИЗАЦІЯ ЛОГІСТИЧНИХ ПОСЛУГ	
<b>V. Hrechanyk</b> SYSTEMATIZATION OF LOGISTICS SERVICES	108
<b>О. Гузеляк, Ю. Шевчук, Б. Береженко, І. Боднарчук</b> ПРОГРАМНА АРХІТЕКТУРА В РОЗПОДІЛЕНИХ КОМАНДАХ ГНУЧКИХ ПРОЄКТІВ	
<b>O. Huzeliak, Yu. Shevchuk, B. Berezenko, I. Bodnarchuk</b> SOFTWARE ARCHITECTURE DESIGN IN DISTRIBUTED TEAMS OF AGILE PROJECTS	109
<b>О. Дзюма, І. Мудрик</b> ДОСЛІДЖЕННЯ СИСТЕМ ТЕСТУВАННЯ НА ОСНОВІ РОЗРОБЛЕНОГО ІНТЕРНЕТ СЕРВІСУ ПОТОКОВОГО АУДІО	
<b>O. Dziuma, I. Mudryk</b> RESEARCH OF TESTING SYSTEMS BASED ON A DEVELOPED INTERNET AUDIO STREAMING SERVICE	113
<b>Н. Доскоч, Г. Цуприк</b> РОЗРОБКА СИСТЕМИ ЗАБЕЗПЕЧЕННЯ БЕЗПЕЧНОГО ОБМІНУ ІНФОРМАЦІЄЮ З ВИКОРИСТАННЯМ ТЕХНОЛОГІЙ ВЕБ ПРОГРАМУВАННЯ	
<b>N. Doskoch, H. Tsupryk</b> DEVELOPMENT OF SAFE INFORMATION EXCHANGE SYSTEM USING WEB PROGRAMMING TECHNOLOGIES	114
<b>О. Кишкевич, А. Кашосі, Д. Михалик</b> ПРОЄКТУВАННЯ ТА РОЗРОБКА ПЛАТФОРМИ КЕРУВАННЯ МАРКЕТИНГОВИМИ ДАНИМИ НА ОСНОВІ AIRFLOW ТА HADOOP	
<b>O. Kyshkevych, A. Kashosi D. Mykhalyk</b> DESIGN AND DEVELOPMENT OF MARKETING DATA MANAGEMENT PLATFORM BASED ON AIRFLOW AND HADOOP	115
<b>А. Коваль, М. Петрик</b> МІКРОСЕРВІСНА АРХІТЕКТУРА В РОЗРОБЦІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	
<b>A. Koval, M. Petryk</b> MICROSERVICE ARCHITECTURE IN SOFTWARE DEVELOPMENT	117
<b>Р. Ковальчук</b> АДАПТАЦІЯ ТЕХНОЛОГІЙ КОНТРОЛЮ ТА УПРАВЛІННЯ ПРОЄКТАМИ В УМОВАХ НЕВИЗНАЧЕНОСТІ НА ОСНОВІ AGILE МЕТОДОЛОГІЇ РОЗРОБКИ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ	
<b>R. Kovalchuk</b> ADAPTATION OF PROJECT CONTROL AND MANAGEMENT TECHNOLOGIES IN CONDITIONS OF UNCERTAINTY BASED ON AGILE SOFTWARE DEVELOPMENT METHODOLOGY	118
<b>В. Масловський</b> ВЕБ-ЗАСТОСУНОК МЕРЕЖІ ДОСТАВОК І АСОРТИМЕНТУ ЗАКЛАДІВ ХАРЧУВАННЯ	
<b>V. Maslovskiy</b> WEB APPLICATION OF THE DELIVERY NETWORK AND ASSORTMENT OF FOOD INSTITUTIONS	119

<b>Т. Матвієнко, Г. Цуприк</b> РОЗРОБКА СИСТЕМИ КООРДИНАЦІЇ ТА ВИЗНАЧЕННЯ КРИТИЧНОГО ШЛЯХУ ПЕРЕМІЩЕННЯ РУХОМОГО СКЛАДУ НА МОВІ JAVA І ФРЕЙМВО- РКІВ SPRING, REACT	
<b>Т. Matviienko, H. Tsupryk</b> DEVELOPMENT OF SYSTEM FOR TRAFFIC COORDINATION OF TRACKS MOVING USING JAVA LANGUAGE AND SPRING, REACT FRAMEWORKS	120
<b>О. Мельник, М. Петрик</b> РОЗРОБКА АРХІТЕКТУРИ ІНФОРМАЦІЙНОЇ СИСТЕМИ ДІЛОВИХ КОМУНІКАЦІЙ НА ОСНОВІ ОПТИМІЗАЦІЙ СТРІМІНГОВОЇ ПЛАТФОРМИ	
<b>О. Melnyk, M. Petryk</b> DEVELOPMENT OF ARCHITECTURE FOR BUSINESS COMMUNICATIONS INFORMATION SYSTEM BASED ON STREAMING PLATFORM OPTIMIZATIONS	121
<b>Є. Музиченко, Ю. Стоянов</b> СИСТЕМА МОНИТОРИНГУ ТРАФІКУ В ІЄРАРХІЧНИХ КОМП'ЮТЕРНИХ МЕРЕЖАХ	
<b>Е. Muzychenko, Y. Stoyanov</b> TRAFFIC MONITORING SYSTEM IN HIERARCHICAL COMPUTER NETWORKS	122
<b>Н. Новак</b> ПОКРАЩЕННЯ ЯКОСТІ РОЗРОБЛЕНОГО ПРОДУКТУ ЗА ДОПОМОГОЮ СИСТЕМ АВТОМАТИЗОВАНОГО ТЕСТУВАННЯ	
<b>N. Novak</b> IMPROVEMENT OF THE QUALITY OF THE DEVELOPED PRODUCT WITH THE HELP OF THE AUTOMATED TESTING SYSTEM	123
<b>О. Остапчук, Г. Цуприк</b> ТЕХНІЧНІ ОСОБЛИВОСТІ ВЗАЄМОДІЇ МІЖ КЛІЄНТОМ ТА СЕРВЕРОМ У РЕАЛЬНОМУ ЧАСІ	
<b>О. Ostarchuk, H. Tsuprik</b> TECHNOLOGICAL FEATURES OF REAL-TIME CLIENT-SERVER COMMUNICATION	124
<b>В. Пацула</b> РОЗРОБКА СИСТЕМИ ОПОВІЩЕННЯ ДЛЯ ДИСТАНЦІЙНОГО НАВЧАННЯ ЗА ДОПОМОГОЮ TELEGRAM BOT API НА МОВІ ПРОГРАМУВАННЯ C#	
<b>V. Patsula</b> NOTIFICATION SYSTEM DEVELOPMENT FOR DISTANCE LEARNING USING TELEGRAM BOT API AND C# PROGRAMMING LANGUAGE	125
<b>О. Петрик, І. Стадник</b> РОЗРОБКА АВТОМАТИЗОВАНОЇ СИСТЕМИ ОБЛІКУ ВНУТРІШНЬО ПЕРЕМІЩЕНИХ ОСІБ	
<b>О. Petryk, I. Stadnyk</b> DEVELOPMENT OF AN AUTOMATED ACCOUNTING SYSTEM FOR INTERNALLY DISPLACED PERSONS	126
<b>В. Простяк</b> РОЗРОБКА WEB-САЙТУ ІНТЕРНЕТ-МАГАЗИНУ ДЛЯ ПРОДАЖУ СПОРТИВНОГО ОДЯГУ НА ОСНОВІ MAGENTO 2	
<b>V. Prostiak</b> WEB SITE DEVELOPMENT OF AN INTERNET STORE FOR THE SALE OF SPORTS CLOTHING BASED ON MAGENTO 2	127

<b>Н. Свитак</b> ПІДВИЩЕНИЙ РІВЕНЬ ПРОКРАСТИНАЦІЇ В ЛЮДЕЙ ЧЕРЕЗ ЗБІЛЬШЕННЯ ІНФОРМАЦІЇ І ДОСТУПУ ДО НЕЇ	
<b>N. Svytak</b> INCREASED LEVEL OF PROCRASTINATION IN PEOPLE DUE TO INCREASING INFORMATION AND ACCESS TO IT	128
<b>І. Сіжук, В. Бревус</b> РОЗРОБКА ЗВУКОВИХ ЕФЕКТІВ ДЛЯ ПЛАГІНІВ ВІРТУАЛЬНОЇ СТУДІЇ ЦИФРОВИХ ЗВУКОВИХ РОБОЧИХ СТАНЦІЙ	
<b>I. Sizhuk, V. Brevus</b> DEVELOPMENT OF SOUND EFFECTS FOR VIRTUAL STUDIO PLUG-INS` DIGITAL AUDIO WORKSTATIONS	128
<b>Д. Сомін, А. Паламар, В. Волоський</b> ПЕРЕВАГИ WEBASSEMBLY ЯК ІНСТРУМЕНТА РЕАЛІЗАЦІЇ АЛГОРИТМІВ У РЕСУРСОМІСТКИХ ВЕБ-ДОДАТКАХ	
<b>D. Somin, A. Palamar, V. Voloskyi</b> WEBASSEMBLY ADVANTAGES AS A TOOL FOR ALGORITHMS IMPLEMENTATION IN HIGH-LOAD WEB APPLICATIONS	130
<b>І. Стефанишин, М. Петрик</b> ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ ІДЕНТИФІКАЦІЇ ПАРАМЕТРІВ АНОРМАЛЬНИХ НЕРВОЛОГІЧНИХ РУХІВ ЛЮДИНИ ПІД ДІЄЮ ТЕХНОГЕНИХ НАВАНТАЖЕНЬ	
<b>I. Stefanyshyn, M. Petryk</b> DESIGN OF THE INFORMATION SYSTEM FOR THE IDENTIFICATION OF THE PARAMETERS OF ABNORMAL NEUROLOGICAL HUMAN MOVEMENTS UNDER THE EFFECT OF TECHNOLOGICAL LOADS	131
<b>О. Сторожук</b> ТЕХНОЛОГІЯ БЛОКЧЕЙН В ОСНОВІ NFT	
<b>O. Storozhuk</b> NFT TECHNOLOGY BASED ON BLOCKCHAIN	132
<b>Є. Тимченко, Г. Цуприк</b> РОЗРОБКА РОЗУМНОЇ СИСТЕМИ ОБЛІКУ РОБОТИ КОМП'ЮТЕРНОЇ МЕРЕЖІ ПІДПРИЄМСТВА В СЕРЕДОВИЩІ EMBARCADERO RAD STUDIO XE	
<b>Ye. Tymchenko, H. Tsupryk</b> DEVELOPMENT OF A ACCOUNTING SMART SYSTEM OF COMPUTER NETWORK SERVICE OF THE ENTERPRISE IN THE EMBARCADERO RAD STUDIO XE ENVIRONMENT	133
<b>СЕКЦІЯ 5. НОВІТНІ ФІЗИКО-ТЕХНІЧНІ ТА ОСВІТНІ ТЕХНОЛОГІЇ</b>	
<b>Я. Войтович, А. Лупенко</b> МЕТОДИ ПОБУДОВИ ШИРОКОСМУГОВИХ МЕРЕЖ ДОСТУПУ	
<b>Y. Voitovych, A. Lupenko</b> METHODS OF BUILDING BROADBAND ACCESS NETWORKS	134
<b>Ю. Попович</b> РОЗРОБЛЕННЯ АВТОМАТИЗОВАНОГО МЕТОДУ КОНТРОЛЮ ПОВЕРХОНЬ МЕТАЛОКОНСТРУКЦІЙ, ВІДНОВЛЕНИХ РОБОТИЗОВАНИМ НАПЛАВЛЕННЯМ	
<b>Yu. Popovych</b> DEVELOPMENT OF AN AUTOMATED METHOD OF CONTROLLING THE SURFACES OF METAL STRUCTURES RESTORED BY ROBOTIC SURFACING	135

<b>Г. Осухівська, А. Волошук</b> ТЕХНОЛОГІЇ ПЕРЕДАВАННЯ ТА ОПРАЦЮВАННЯ ДАНИХ В КОМП'ЮТЕРИЗОВАНИХ СИСТЕМАХ ОБЛІКУ ЕЛЕКТРОЕНЕРГІЇ	
<b>Н. Osukhivska, A. Voloshchuk</b> DATA TRANSMISSION AND PROCESSING TECHNOLOGIES IN COMPUTERIZED ELECTRICITY ACCOUNTING SYSTEMS	136
<b>Ю. Дзюбак, Я. Коненко, Н. Луцик</b> АНАЛІЗ МЕТОДІВ ТА ЗАСОБІВ ХОСТИНГУ РЕСУРСІВ З ВИКОРИСТАННЯМ ВІРТУАЛІЗАЦІЇ	
<b>Yu. Dziubak, Ya. Konenko, N. Lutsyk</b> ANALYSIS OF METHODS AND MEANS RESOURCE HOSTING USING VIRTUALIZATION	137
<b>Ю. Дзюбак, Я. Коненко, Я. Войтович</b> ТЕСТУВАННЯ ТА ПОРІВНЯЛЬНИЙ АНАЛІЗ ПЛАТФОРМ ВІРТУАЛІЗАЦІЇ	
<b>Yu. Dziubak, Ya. Konenko, Ya. Voytovych</b> TESTING AND COMPARATIVE ANALYSIS OF VIRTUALIZATION PLATFORM	138
<b>А. Курко, М. Хом'як, К. Урста, А. Шелельо</b> ОПТИМІЗАЦІЯ РОБОТИ ВИПАРНОЇ СТАНЦІЇ ПРИ ВИГОТОВЛЕННІ СОКУ	
<b>A. Kurko, M. Khomiak, K. Ursta, A. Shelelo</b> OPTIMIZATION OF EVAPORATION STATION DURING JUICE PRODUCTION	139
<b>Х. Ольховецька, Г. Осухівська</b> МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНОЇ СИСТЕМИ КОНТРОЛЮ ПРОЦЕСУ ФЕРМЕНТАЦІЇ	
<b>K. Olkhovetska, H. Osukhivska</b> MATHEMATICAL SUPPORT OF THE COMPUTER SYSTEM FOR CONTROL OF THE FERMENTATION PROCESS	140
<b>О. Прокопюк, Н. Ромашевська, Я. Войтович, Р. Жаровський</b> АВТОМАТИЗАЦІЯ ТА ОПТИМІЗАЦІЯ НАВЧАЛЬНОГО ПРОЦЕСУ В ВИЩИХ НАВЧАЛЬНИХ ЗАКЛАДАХ	
<b>O. Prokopyuk, N. Romashevskya, Ya. Voytovych, R. Zharovskyi</b> AUTOMATION AND OPTIMIZATION OF THE EDUCATIONAL PROCESS IN HIGHER EDUCATIONAL INSTITUTIONS	141
<b>Н. Ромашевська, О. Прокопюк, Р. Жаровський</b> ВИКОРИСТАННЯ ТЕХНОЛОГІЇ ВІРТУАЛІЗАЦІЇ У ПРОЦЕСІ НАВЧАННЯ СТУДЕНТІВ	
<b>N. Romashevskya, O. Prokopyuk, R. Zharovskyi</b> AUTOMATION AND OPTIMIZATION OF THE EDUCATIONAL PROCESS IN HIGHER EDUCATIONAL INSTITUTIONS	142
<b>В. Тимошук, В. Карташов, Р. Королюк, Т. Рубен</b> ОГЛЯД ПРОТОКОЛІВ КЕРУВАННЯ ДЛЯ ПОБУДОВИ АВТОМАТИЗОВАНИХ СИСТЕМ ВІДДАЛЕНОГО УПРАВЛІННЯ	
<b>V. Tymoshchuk, V. Kartashov, R. Koroliuk, T. Ruben</b> OVERVIEW OF CONTROL PROTOCOLS FOR BUILDING AUTOMATED REMOTE CONTROL SYSTEMS	143
<b>П. Федорів, І. Федорів</b> ВИКОРИСТАННЯ ПНЕВМО-СТРУМЕНЕВИХ ЗАХОПЛЮВАЧІВ В АВТОМАТИЧНИХ СЕПАРАТОРАХ ЛИСТОВОГО МАТЕРІАЛУ	
<b>P. Fedoriv, I. Fedoriv</b> THE USE OF PNEUMATIC-JET ATTACHERS IN AUTOMATIC SEPARATORS OF SHEET MATERIAL	145

**М. Штиук, О. Юрчик, О. Тотосько, В. Левицький**

**ОГЛЯД РЕЗУЛЬТАТІВ ВИКОРИСТАННЯ ДАВАЧІВ ВОЛОГОСТІ,  
ВИГОТОВЛЕНИХ З КАРТОННИХ ТА ПАПЕРОВИХ ВІДХОДІВ**

**M. Shtyuk, O. Yurchyk, O. Totosko, V. Levytskyi,**

**AN OVERVIEW OF THE RESULTS OF USING HUMIDITY SENSORS MADE  
FROM CARDBOARD AND PAPER WASTE**

146

**ЗМІСТ**

147





# **ІНФОРМАЦІЙНІ МОДЕЛІ, СИСТЕМИ ТА ТЕХНОЛОГІЇ**

## **Матеріали тез доповідей X науково-технічної конференції 7–8 грудня 2022 року**

Формат 60x90, папір ксероксний.  
Обл. вид. арк. 12,79  
Тираж 300 прим.

Видавництво Тернопільського національного технічного університету імені Івана Пулюя  
вул. Руська, 56, м. Тернопіль, 46001  
Тел. 52-21-99, 42-79-65

Свідоцтво про внесення суб'єкта видавничої справи до державного реєстру видавців,  
виготівників і розповсюджувачів видавничої продукції ДК № 4226 від 08.12.2011 р.