



Галицький фаховий коледж імені В'ячеслава Чорновола

# ЗБІРНИК ТЕЗ

за матеріалами студентських  
науково-практичних конференцій  
в рамках Днів Науки 2024

в Галицькому фаховому коледжі імені В'ячеслава  
Чорновола



секція «Комп'ютерних технологій» та  
секція «Фізико-математичних та природничих дисциплін»

Тернопіль 2024



Редакційна колегія:

*Глинська Марина Любомирівна* - заступник директора Галицького фахового коледжу імені В'ячеслава Чорновола з навчально-методичної роботи.

*Посвятовська Ольга Богданівна* - голова ЦК інформатики та комп'ютерних дисциплін Галицького фахового коледжу імені В'ячеслава Чорновола,

*Сиротюк Оксана Богданівна* - викладач комп'ютерних дисциплін Галицького фахового коледжу імені В'ячеслава Чорновола

*Чубей Олександра Орестівна* - голова ЦК фізико-математичних та природничих дисциплін Галицького фахового коледжу імені В'ячеслава Чорновола.

ЗБІРНИК тез за матеріалами студентських науково-практичних конференцій в рамках Днів Науки 2024 в Галицькому фаховому коледжі імені В'ячеслава Чорновола (секція «Комп'ютерних технологій» та секція «Фізико-математичних та природничих дисциплін» – Тернопіль: Галицький фаховий коледж імені В'ячеслава Чорновола, 2024. 114 с.

До збірника увійшли матеріали і тези доповідей, які подано учасниками студентських науково-практичних конференцій в рамках Днів Науки в Галицькому фаховому коледжі імені В'ячеслава Чорновола (секція «Комп'ютерних технологій» та секція «Фізико-математичних та природничих дисциплін»).

При публікації максимально збережено орфографію, пунктуацію та стилістику запропоновану авторами та підтриману науковими керівниками. За зміст праць та достовірність наведених фактів і статистичних даних відповідальність несуть автори та їх наукові керівники.

Рекомендовано до друку Педагогічною радою Галицького фахового коледжу імені В'ячеслава Чорновола

(протокол № 8 від 15.05.2024 року)

© Галицький фаховий коледж імені В'ячеслава Чорновола, 2024

Верстка: Музика М.- інженер-електронік

Галицького фахового коледжу імені В'ячеслава Чорновола

## ЗМІСТ

|  |    |
|--|----|
| СТВОРЕННЯ КОМП'ЮТЕРНОЇ ГРИ В ПОПУЛЯРНОМУ ЖАНРІ ШУТЕР<br>Біла Анастасія.....  | 5  |
| ТЕХНОЛОГІЇ РОЗРОБКИ ПРОГРАМНИХ ЗАСОБІВ АСИМЕТРИЧНОГО<br>ШИФРУВАННЯ ФАЙЛІВ НА ОСНОВІ ЕЛІПТИЧНИХ КРИВИХ<br>Борак Іван.....                   | 12 |
| СИСТЕМА КОНТРОЛЮ ЯКОСТІ ПОВІТРЯ В ТИМЧАСОВОМУ УКРИТТІ<br>Буката Анна.....  | 17 |
| ОЗЕЛЕНЕННЯ ДАХІВ<br>Довганик Марія.....  | 21 |
| ІНТЕГРАЛ В ПОВСЯКДЕННОМУ ЖИТТІ ЛЮДИНИ<br>Дудун Юрій.....   | 26 |
| РОЗРОБКА СИСТЕМИ АВТОМАТИЗАЦІЇ МАРКЕТИНГУ ЛОКАЛЬНОГО<br>БІЗНЕСУ В СФЕРІ SMM<br>Ковальчук Яна.....  | 30 |
| АНАЛІЗ МЕРЕЖЕВОГО ТРАФІКУ ДЛЯ ВИЯВЛЕННЯ АНОМАЛІЙ ТА<br>ЙОГО РОЛЬ В СФЕРІ КІБЕРБЕЗПЕКИ<br>Лобас Сергій.....                                 | 36 |
| ДОСЛІДЖЕННЯ ТА ВІЗУАЛІЗАЦІЯ ЕФЕКТИВНОСТІ АЛГОРИТМІВ<br>СОРТУВАННЯ НА ОСНОВІ РЕАЛІЗОВАНОГО ПРОГРАМНОГО<br>ЗАСТОСУНКУ<br>Музика Михайло..... | 43 |
| АРХІТЕКТУРА ТА ПРИНЦИПИ РОБОТИ СИСТЕМИ МІЖПРОЦЕСНОЇ<br>ВЗАЄМОДІЇ D-BUS В ОПЕРАЦІЙНИХ СИСТЕМАХ GNU/LINUX<br>Музика Михайло.....             | 49 |
| ТЕХНОЛОГІЇ РОЗРОБКИ ОНЛАЙН-ТЕСТІВ<br>Москалюк Назарій.....   | 55 |
| ТЕХНОЛОГІЇ РЕАЛІЗАЦІЇ КРОСПЛАТФОРМЕНОСТІ WEB-ЗАСТОСУНКІВ   |    |

|   |     |
|---|-----|
| Нищота Юрій.....  | 64  |
| ПРИСТРІЙ АВТОМАТИЗОВАНОГО КЕРУВАННЯ ВІДДАЛЕНОЮ СИСТЕМОЮ СПОСТЕРЕЖЕННЯ                             |     |
| Ольховецький Андрій.....  | 70  |
| SOFT SKILLS ТА HARD SKILLS РОЗРОБНИКА ІГРОВИХ ПРОГРАМ ТА ЇХ СПЕЦИФІКА В УКРАЇНІ                   |     |
| Пелех Мар'ян.....   | 73  |
| СПЕЦИФІКА ВИКОРИСТАННЯ TINKERCAD ДЛЯ РОЗРОБКИ ПРОЄКТІВ  |     |
| Підодвірний Констянтин.....   | 80  |
| РОЗРОБКА ТА РЕАЛІЗАЦІЯ ВЕБСАЙТУ ДЛЯ НАВЧАННЯ ХІМІЇ: ІНТЕРАКТИВНИЙ ПІДХІД ДО ОСВІТИ                |     |
| Поточняк Роман.....   | 84  |
| ТЕХНОЛОГІЇ РЕАЛІЗАЦІЇ ІГРОВИХ ДОДАТКІВ  |     |
| Рибчак Максим.....  | 88  |
| ВПЛИВ ЕКОНОМІЧНИХ ЗАВДАНЬ У ГРІ НА РОЗВИТОК ФІНАНСОВОЇ ГРАМОТНОСТІ ГРАВЦІВ                        |     |
| Романовська Анна.....   | 93  |
| СИСТЕМА СПОВІЩЕННЯ ПРО ПОВІТРЯНУ ТРИВОГУ  |     |
| Рудник Вадим.....   | 97  |
| ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В СУЧАСНОМУ СВІТІ   |     |
| Солтис Н.М, Стиран В.А.....   | 101 |
| ЗАХИСТ БАЗ ДАНИХ ВІД SQL ІН'ЄКЦІЙ   |     |
| Худоб'як Костянтин.....   | 106 |
| РОБОТОТЕХНІКА В АГРОНОМІЇ: АВТОМАТИЗАЦІЯ ПРОЦЕСІВ КОНТРОЛЮ ЗА ТЕХНІКОЮ ТА СІЛЬСЬКИМ ГОСПОДАРСТВОМ |     |
| Черних Діана, Процик Святослав.....   | 111 |

*Біла Анастасія, здобувач фахової передвищої освіти  
IV курсу спеціальності «Комп'ютерні науки»  
науковий керівник  
викладач комп'ютерних дисциплін Посвятовська О. Б.*

## **СТВОРЕННЯ КОМП'ЮТЕРНОЇ ГРИ В ПОПУЛЯРНОМУ ЖАНРІ ШУТЕР**

Індустрія комп'ютерних ігор – це велика частина економіки, яка створює робочі місця, залучає інвестиції та приносить значні доходи бізнесу. Це підтверджує щорічна статистика Steam, яку публікує американська корпорація Valve. Так, у 2023 році понад 500 ігрових проєктів отримали загальний прибуток понад 3 мільйонів доларів США [1].

Відповідно до аналізу статистичних даних від аналітичної компанії Newzoo [2] можна зробити наступні висновки: залежно від віку, різні покоління мають різні вподобання щодо розваг. Згідно з дослідженням, 94% представників покоління Альфа (народжені після 2010 року) захоплюються іграми. Покоління Z (2000 – 2010 роки) насолоджуються іграми нарівні з активністю в соціальних мережах, і це становить 90%. Представники покоління Y або "міленіали" (1982– 1999 роки) виявляють зацікавленість у відсотку 82%. Покоління X (1965 – 1981 роки) захоплені іграми на 64%. У той час, як у представників покоління "бебі-бумерів" (1955 – 1964 роки) основною мотивацією для гри є відпочинок і розслаблення, тому вони віддають перевагу головоломкам або платформеним іграм, що складає 43% їх захопленості (рис. 1).

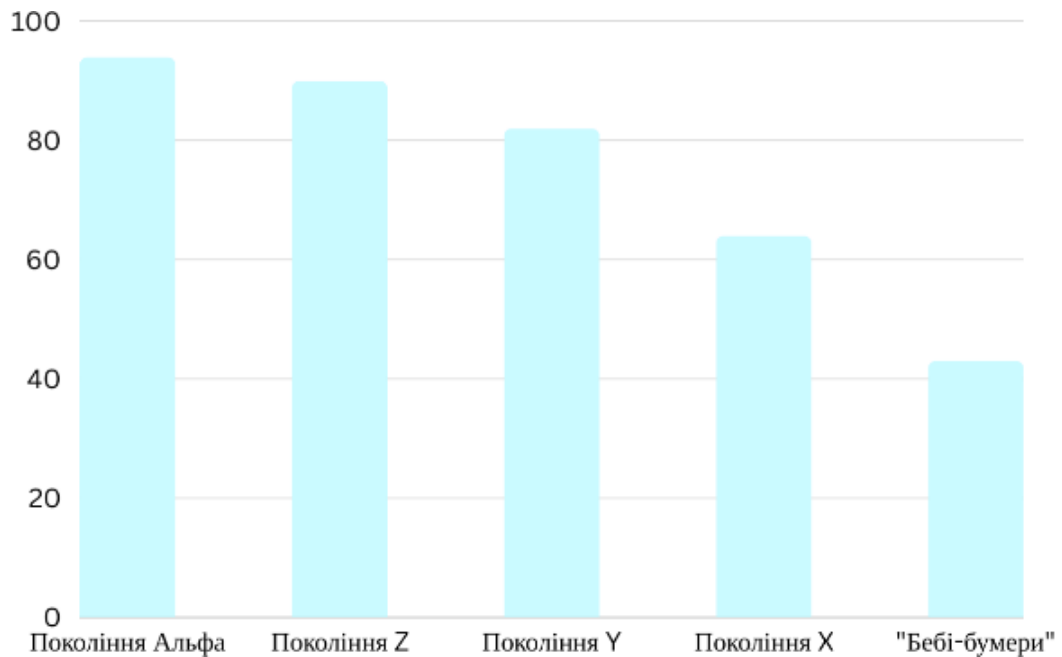


Рисунок 1 — Відсоток захопленості іграми відносно поколінь

Кожне покоління любить й різні ігрові жанри [3]. Для покоління Z найпопулярнішими жанрами ігор є шутери (66%), бойовики/пригодницькі ігри (62%), симулятори (42%), і спортивні ігри (41%). "Міленіали" також віддають перевагу шутерам (62%), а потім екшен/пригодам (58%), наступними йдуть перегони та спортивні ігри (44%) та симулятори (42%). А серед покоління X найбільш популярні екшен та шутери (46%), головоломки та платформери (35%), а також спортивні ігри (34%) та гонки (33%).

Американська компанія NielsenIQ у співпраці з Wargaming опублікувала результати комплексного дослідження про геймерів та відеоігри в Україні. [4] Відповідно до наведеного звіту, найбільш популярними ігровими жанрами є такі: Puzzle, Vehicle Shooter і Shooter, що відображено на рисунку 2.

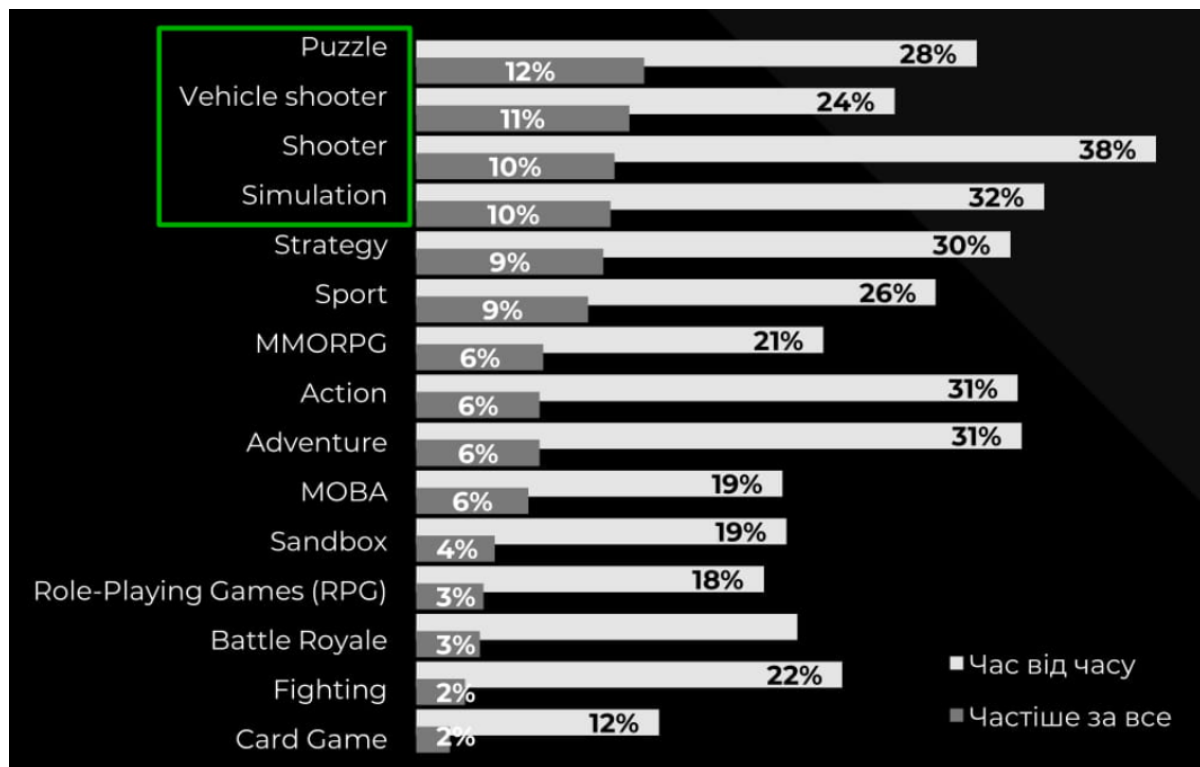


Рисунок 2 — Статистика найпопулярніших жанрів ігор в Україні

На основі аналізу останніх тенденцій в галузі комп'ютерних ігор популярність жанру шутер невинно зростає, що зумовлено наступними факторами:

- інновації в ігровому дизайні, що надають гравцям змогу знаходити новий та цікавий ігровий досвід;
- унікальність ігрового процесу завжди привертає увагу завдяки нестандартному геймплею та охоплює ширшу аудиторію, завдяки можливості поєднувати різні режими гри;
- потенціал для майбутнього розвитку, зокрема, через використання інтерактивних технологій, віртуальної реальності та штучного інтелекту.

У сучасному світі різко зросла і важливість соціальних функцій в іграх, що відкриває для розробників додаткові можливості розвитку бізнесу та монетизації продукту. У таблиці 1 систематизовано та прояснено сутність моделей монетизації проєктів ігрової індустрії. [5]

Таблиця 1 – Моделі монетизації проєктів ігрової індустрії

| Модель                         | Суть   | Приклад застосування                                   |
|--------------------------------|--|--|
| Роздрібна торгівля             | Купівля фізичної копії гри в магазині або інтернет-магазині.   | Більша частина сучасних проєктів                       |
| Цифрове розповсюдження         | Придбання цифрової копії гри через інтернет.   |  |
| Підписка                       | Підписка на послуги, які надають можливість грати в онлайн-ігри, надають безкоштовно ігри або тимчасовий доступ до бібліотеки з різними проєктами. | PlayStation Plus, Origin Access                        |
| Мікротранзакції                | Купівля покупцем додаткових елементів гри. Спеціальні предмети, текстури зброї, розблоковані персонажі тощо.                                       | FIFA, Overwatch, Call of Duty, Warface, World of Tanks |
| Додаткові матеріали (DLC)      | Додатковий матеріал у вигляді додаткової сюжетної лінії, тобто гравці купують частини гри окремо.  | Bioshock Infinite, Half-Life 2, DOOM Eternal           |
| Коробка зі здобиччю (loot box) | Додаткові ігрові предмети, які випадковим чином з'являються при відкритті ящиків.  | CounterStrike: Global Offensive, Call of Duty          |
| Сезонний абонемент             | Купівля повної гри та доповнень. Кожну частину гри можна придбати зі знижкою порівняно з покупкою кожної частини окремо.                           | The Witcher 3: Wild Hunt, DOOM Eternal, Anno 1800      |
| Торгівля між гравцями          | Ігровими предметами та валютою можна вільно обмінюватись між гравцями на внутрішньо ігровому ринку.  | Сервіс Steam   |
| Реклама                        | Розміщення реклами в грі, в різних   | Більша частина   |



|  |  |                |
|--|--|----------------|
|  | формах (банерна реклама, рекламні паузи та розміщення рекламних продуктів у гру тощо). | мобільних ігор |
|--|--|----------------|

Для успішної реалізації ігрового застосунку в жанрі шутер при проєктуванні доцільно використовувати UML-діаграму варіантів використання, UML-діаграму класів, BPMN-діаграму проходження ігрового застосунку (рис. 3).

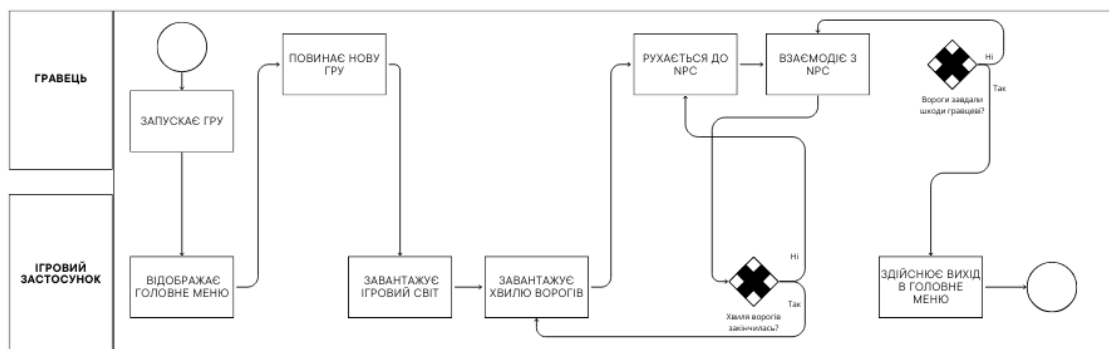


Рисунок 3 — Діаграма моделі та опис бізнес-процесів, які виконуються в ігровому додатку

Відповідно до рисунка діаграми вище, гра починається з того, що після запуску гри відображається головне меню. Натиснувши відповідну кнопку меню, гравець починає гру, і відображається в ігровому світі. Потім він дізнається про мету, сюжет і завдання гри. Далі розпочинає взаємодію з NPC і рухається до них. Якщо хвиля ворогів NPC завершиться, гравець переходить на новий рівень. Інакше гравець продовжує взаємодію з NPC та повторює дії, які були виконані раніше. Гра завершується, якщо гравець зазнає шкоди та вичерпає всі ресурси здоров'я, і він виходить у головне меню.

Реалізація ігрового застосунку передбачає створення продукту для ПК та ОС Windows у тривимірному просторі. Найбільш популярними рушіями для розробки, які відповідатимуть даним вимогам, є Unity та Unreal Engine. Аналізуючи переваги та недоліки кожного з них, було обрано для розробки застосунку рушієм Unity, завдяки його невисоким системним вимогам,

масштабному маркетплейсу Unity Asset Store та зрозумілому користувацькому інтерфейсу. Хоча Unreal Engine відомий можливістю створення ігор у високоякісній графіці та використанням блупринтів, проте не завжди можлива сумісність із процесорами AMD Ryzen. Результат реалізації гри у жанрі шутер за допомогою рушія Unity зображено на рисунку 4.



Рисунок 4 — Реалізація ігрового застосунку за допомогою рушія Unity

На основі розробленого ігрового застосунку у жанрі шутер можна зробити наступні висновки. Розроблені UML-діаграми сприяли кращій структуризації коду та логіки гри, а використані бібліотеки мови C# і компоненти рушія Unity Engine допомогли створити якісну візуальну частину гри. Актуальність жанру, висока якість геймплею, звукових ефектів та графіки дають можливість конкурувати поміж великої кількості схожих продуктів. Ігровий застосунок відповідає сучасним вимогам ринку та потребам цільової аудиторії, забезпечуючи стабільний дохід у майбутньому.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. У 2023 році понад 500 ігор заробили більш ніж \$3 мільйони доходу в Steam. *Українська спільнота GameDev DOU*. URL: <https://gamedev.dou.ua/news/steam-games-revenue/> (дата звернення: 08.05.2024).
2. Newzoo Report Reveals Gaming Trends Across Generations in 2023 | GAM3S.GG. GAM3S.GG. URL: <https://gam3s.gg/news/newzoo-report-generational-gamers/> (дата звернення: 08.05.2024).
3. Shooters or sports: What's the most popular gaming genre? - GWI. GWI. URL: <https://blog.gwi.com/marketing/most-popular-gaming-genre/> (дата звернення: 08.05.2024).
4. NielsenIQ и Wargaming назвали популярні відеоігри в Україні. Overclockers.ua. URL: <https://www.overclockers.ua/news/games/2021-02-18/128629/> (дата звернення: 08.05.2024).
5. Гречко А. В., Захаров Н. В., Фалько М. О. АНАЛІЗ ДИНАМІКИ РОЗВИТКУ РИНКУ ВІДЕОІГОР, ДЖЕРЕЛ ЙОГО ФІНАНСУВАННЯ ТА ОСОБЛИВОСТЕЙ МОНЕТИЗАЦІЇ ПРОДУКЦІЇ В ДАНІЙ СФЕРІ. Ефективна економіка. (дата звернення: 09.05.2024).

*Борак Іван, здобувач фахової передвищої освіти IV курсу  
спеціальності «Комп'ютерні науки»  
науковий керівник Івасьєв С. В.*

## **ТЕХНОЛОГІЇ РОЗРОБКИ ПРОГРАМНИХ ЗАСОБІВ АСИМЕТРИЧНОГО ШИФРУВАННЯ ФАЙЛІВ НА ОСНОВІ ЕЛІПТИЧНИХ КРИВИХ**

У цифровому світі, де велика кількість особистих даних перебуває в постійній загрозі мережових атак, важливою стає не лише технологічна складова безпеки, але і свідомість користувачів щодо забезпечення конфіденційності своїх даних. Популярність асиметричного шифрування та еліптичних кривих зростає у зв'язку з високою крипостійкістю та простотою реалізації.

Асиметричне шифрування, що використовує різні ключі для кодування та декодування, є ефективним засобом забезпечення безпеки. На сьогоднішній день, найпоширеніші алгоритми шифрування включають AES, RSA, ECC та схему обміну ключами Diffie-Hellman Key Exchange. Еліптичні криві в цьому контексті виділяються як перспективний підхід, оскільки вони забезпечують високий рівень захисту при невеликих розмірах ключів[1].

Таким чином, розвиток технологій шифрування йде поруч із поширенням культури цифрової безпеки.

На жаль, на сьогоднішній день не існує багатьох готових програм для асиметричного шифрування файлів на основі еліптичних кривих, які були б широко відомі або легко доступні. Проте існують різноманітні криптографічні бібліотеки, які можуть бути використані для реалізації такого функціоналу. Наприклад, бібліотеки, такі як OpenSSL, Bouncy Castle та Microsoft CryptoAPI (CNG), можуть надати необхідний інструментарій для створення власних програм або скриптів, які здатні виконувати асиметричне

шифрування файлів за допомогою еліптичних кривих. Ці бібліотеки забезпечують широкий функціонал криптографічних операцій і можуть бути використані для створення різноманітних криптографічних застосунків, включаючи шифрування файлів з використанням еліптичних кривих.

Загальноприйнятою бібліотекою для криптографічних операцій є OpenSSL. Серед її переваг можна виділити високий рівень надійності та безпеки, а також широку підтримку криптографічних алгоритмів, включаючи асиметричне шифрування на основі еліптичних кривих. OpenSSL є відкритим програмним забезпеченням і має активну спільноту розробників, що забезпечує постійне оновлення та виправлення потенційних вразливостей.

Однак в OpenSSL можуть бути деякі недоліки. Перш за все, інтерфейс програмування додатків (API) OpenSSL не завжди є дружнім до користувача, і для його використання може знадобитися значна експертиза у криптографії. Крім того, OpenSSL не завжди надає оптимальну продуктивність та ефективність для певних завдань, іноді виникають проблеми з витратою ресурсів системи.

Bouncy Castle - популярна бібліотека для криптографічних операцій. Вона має широкий функціонал і підтримує різні алгоритми шифрування, включаючи асиметричне шифрування на основі еліптичних кривих. Однією з переваг Bouncy Castle є те, що вона має простий інтерфейс програмування додатків (API), який робить її використання більш доступним для широкого кола розробників.

Серед недоліків Bouncy Castle можна виділити меншу поширеність порівняно з OpenSSL, що може вплинути на доступність підтримки та рівень експертизи серед спеціалістів. Крім того, деякі користувачі вказують на меншу швидкодію Bouncy Castle порівняно з аналогічними бібліотеками[2].

Microsoft CryptoAPI (CNG) - це набір інтерфейсів програмування додатків для криптографічних операцій у середовищі Windows. Однією з переваг цієї бібліотеки є її інтеграція з операційною системою Windows, що дозволяє розробникам легше та ефективніше використовувати її для

створення безпечних програм. Проте, деякі розробники вказують на обмежену підтримку алгоритмів та функцій у Microsoft CryptoAPI порівняно з іншими бібліотеками криптографії. Крім того, використання CryptoAPI може бути обмеженим у крос-платформеному програмуванні, оскільки використовується в більшості випадків для середовища Windows.

Для виконання такого завдання можна використовувати різні мови програмування, а також, інші бібліотеки. Для прикладу реалізуємо таке завдання мовою програмування Python, що є однією з найпопулярніших мов програмування, яка використовується для розробки різноманітних програм, включаючи криптографічні застосунки. Однією з його основних переваг є простота та зручність у використанні, що робить його відмінним вибором для широкого кола розробників, від початківців до професіоналів.

У Python існує безліч бібліотек для криптографії, які дозволяють виконувати різноманітні криптографічні операції, включаючи асиметричне шифрування на основі еліптичних кривих. Однією з переваг Python у контексті криптографії є його широке використання та підтримка в різних галузях індустрії, що забезпечує наявність великої кількості ресурсів та документації для розробників. Крім того, Python має активну спільноту розробників, яка постійно розвивається і вносить удосконалення до існуючих бібліотек. Однак, важливо також зазначити, що Python може бути не найшвидшою мовою програмування, особливо для виконання великих обчислювальних завдань у криптографії. Для деяких додатків швидкодія може бути критичною, і в таких випадках розробники можуть шукати альтернативи з використанням інших мов програмування або оптимізації виконання коду Python.

Для прикладу реалізації такого завдання підійде наступний програмний код, який шифрує текст:

```
from tinyec import registry
from Crypto.Cipher import AES
import hashlib, secrets, binascii
```

```

def encrypt_AES_GCM(msg, secretKey):
    aesCipher = AES.new(secretKey, AES.MODE_GCM)
    ciphertext, authTag = aesCipher.encrypt_and_digest(msg)
    return (ciphertext, aesCipher.nonce, authTag)
def ecc_point_to_256_bit_key(point):
    sha = hashlib.sha256(int.to_bytes(point.x, 32, 'big'))
    sha.update(int.to_bytes(point.y, 32, 'big'))
    return sha.digest()
curve = registry.get_curve('brainpoolP256r1')
def encrypt_ECC(msg, pubKey):
    ciphertextPrivKey = secrets.randbelow(curve.field.n)
    sharedECCKey = ciphertextPrivKey * pubKey
    secretKey = ecc_point_to_256_bit_key(sharedECCKey)
    ciphertext, nonce, authTag = encrypt_AES_GCM(msg,
secretKey)
    ciphertextPubKey = ciphertextPrivKey * curve.g
    return (ciphertext, nonce, authTag, ciphertextPubKey)
return plaintext

```

Розроблені функції використовують асиметричне шифрування з використанням алгоритму AES-GCM та еліптичних кривих (ECC). Під час шифрування, спочатку генерується випадковий приватний ключ, який використовується для обчислення загального секрету з публічним ключем. Цей загальний секрет перетворюється в симетричний ключ за допомогою функції хешування. Потім за допомогою отриманого симетричного ключа та алгоритму AES-GCM виконується шифрування повідомлення. В результаті отримується зашифрований текст разом із непередбачуваним вектором та аутентичним тегом, які використовуються для забезпечення цілісності та автентичності повідомлення. Під час розшифрування, використовуючи приватний ключ, відбувається обчислення того самого загального секрету, який використовувався при шифруванні, та отриманий симетричний ключ. За допомогою цього ключа та вхідних параметрів розшифровується зашифрований текст, щоб отримати оригінальне повідомлення (Рисунок 1).

Результат виконання реалізованого алгоритму зображено на рисунку 1.

```
original msg: b'Text to be encrypted by ECC public key and decrypted by its corresponding ECC private key'  
encrypted msg: {'ciphertext': b'bb30ce21d7c50318878881cf1a848f9ca74d26864929c2036807d2c9e921e6a83290ab71e930c7b89b002dd9649e4823da129e82166ece3ed2e5aef9e2ec3d6f2deeb603c2b32825f52b7795ad5e689085c05b56fb9578d31a', 'nonce': b'ae333391b86775a574f9a26505eae897', 'authTag': b'91e32b2d7e0d6f13376b3b7e67b849e2', 'ciphertextPubKey': '0x7c13c9c808dbd20d81ff78fb27184a964d890adf4139a8fc3adb09b43b14f7431'}  
decrypted msg: b'Text to be encrypted by ECC public key and decrypted by its corresponding ECC private key'
```

Рисунок 1 – Результат виконання демонстративного коду програмного засобу

Після дослідження різних підходів до розробки програмних засобів асиметричного шифрування файлів на основі еліптичних кривих можна зробити висновок, що всі перелічені технології є досить ефективними та зручними для цієї мети. Отже, для створення програмних засобів для асиметричного шифрування файлів можна використовувати комбінації підходів та технологій, залежно від вимог до безпеки, функціональних можливостей та особливостей системи. При цьому важливо враховувати основні принципи криптографії, а також специфіку роботи з еліптичними кривими в контексті шифрування файлів.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. X. Zuxiu, "A Modified Security Teach-lab Management Model Based on ECC Digital Certificate," *2012 Fifth International Conference on Intelligent Computation Technology and Automation*, Zhangjiajie, China, 2012, pp. 18-21, doi: 10.1109/ICICTA.2012.12.
2. E. M. Hernández, G. M. Chávez and J. V. Hernández, "Roundness Estimation of Sedimentary Rocks Using Eiptic Fourier and Deep Neural Networks," *2020 IEEE International Autumn Meeting on Power, Electronics and Computing (ROPEC)*, Ixtapa, Mexico, 2020, pp. 1-5, doi: 10.1109/ROPEC50909.2020.9258735.



*Букача Анна, здобувач ОКР фахового молодшого бакалавра  
IV курсу спеціальності «Комп'ютерна інженерія»,  
науковий керівник Посвятовська О.Б.*

## **СИСТЕМА КОНТРОЛЮ ЯКОСТІ ПОВІТРЯ В ТИМЧАСОВОМУ УКРИТТІ**

Забезпечення якості повітря у тимчасових укриттях є важливим завданням у контексті військових дій. Забруднене повітря може становити серйозну загрозу здоров'ю людей, що перебувають в укриттях. Система контролю якості повітря покликана вирішити ці проблеми шляхом використання сучасних технологій моніторингу та очищення повітря. Основною метою створення системи контролю якості повітря є забезпечення нешкідливих та комфортних умов для перебування людей у тимчасових укриттях. Важливість даної теми обґрунтована необхідністю захисту здоров'я людей під час військових дій та інших надзвичайних ситуацій.

На ринку представлені різні системи контролю якості повітря, що використовуються в різних умовах. Деякі з них розглянуто нижче:

1. TuYa Smart Air Quality Monitor
  - Функції: Моніторинг рівня CO<sub>2</sub>, температури, вологості.
  - Переваги: Інтеграція з іншими розумними пристроями, зручний мобільний додаток.
  - Недоліки: Обмежена функціональність порівняно з конкурентами, можливі проблеми зі стабільністю зв'язку.
2. Airthings Wave Mini
  - Функції: Моніторинг радону, вологості, температури.
  - Переваги: Компактний дизайн, інтерактивні сповіщення.

- Недоліки: Обмежені можливості моніторингу, можливі проблеми з точністю вимірювань.

### 3. Xiaomi ClearGrass Air Monitor

- Функції: Моніторинг якості повітря, температури, вологості.
- Переваги: Висока точність вимірювань, інтеграція з екосистемою Xiaomi.
- Недоліки: Обмежені можливості звітності та аналізу даних, можлива проблема з тривалістю життєвого циклу.

На основі аналізу існуючих систем було розроблено систему контролю якості повітря, яка має наступні переваги. Реалізована система забезпечує широкий спектр моніторингу: вона контролює більше параметрів повітря, включаючи CO<sub>2</sub>, формальдегід, пил, температуру, вологість та інші шкідливі речовини. Використання датчиків CO<sub>2</sub>, температури, формальдегіду, пилу та вологості забезпечує високу точність вимірювань та швидку реакцію на зміни у складі повітря, що робить нашу систему більш ефективною. Розроблена система орієнтована на інтеграцію з Інтернетом речей (IoT), що дозволяє здійснювати віддалений моніторинг та управління, підвищуючи зручність користування. Автоматизація процесів дозволяє автоматично керувати системою фільтрації та вентиляції на основі отриманих даних, що знижує потребу у втручанні людини та підвищує ефективність роботи системи. І, нарешті, наша система має конкурентоспроможну вартість при високій функціональності та надійності, що робить її доступною для широкого кола користувачів.

Ключові аспекти розробленої системи:

- Впровадження високочутливих сенсорів для постійного контролю за якістю повітря є основним компонентом системи. Ці сенсори здатні виявляти навіть незначні концентрації шкідливих речовин, таких як CO<sub>2</sub>, формальдегід, пил та інші.
- Розроблений інтуїтивно зрозумілий інтерфейс дозволяє адміністратору системи та людям котрі перебувають в укритті легко моніторити стан повітря,

керувати системою та отримувати повідомлення про її роботу. Це забезпечує зручність використання системи і підвищує її ефективність.

Важливим аспектом є також впровадження енергоефективних рішень, які дозволяють системі працювати ефективно з мінімальним споживанням енергії. Це критично в умовах тимчасових укриттів, оскільки дозволяє знизити експлуатаційні витрати і забезпечити безперебійну роботу системи.

В перспективі передбачено автоматичне регулювання параметрів якості повітря. Вона передає дані на екран та в додаток ThingView Free, що дозволяє користувачам моніторити якість повітря у реальному часі. На рисунку 1. показано, як виглядає інтерфейс даних у веб-додатку ThingSpeak, де відображаються графіки показників CO<sub>2</sub>, температури, концентрації пилу та якості повітря.

#### Channel Stats

Created: [about a month ago](#)  
Last entry: [less than a minute ago](#)  
Entries: 33

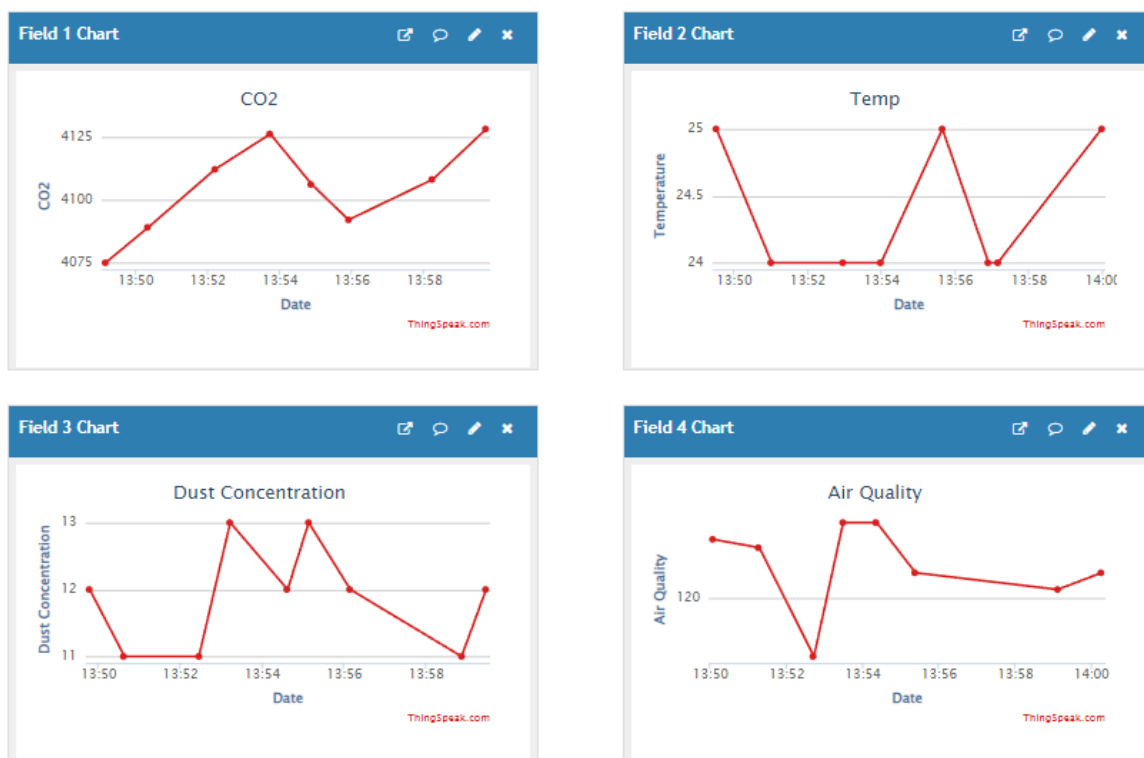


Рисунок 1 — Інтерфейс даних

Розроблена система контролю якості повітря у тимчасових укриттях є актуальною для забезпечення безпеки та комфорту людей у складних умовах. Використання сучасних технологій, таких як високочутливі сенсори, автоматизовані системи управління та ефективні фільтри, дозволяє створити надійну та ефективну систему, яка забезпечить чисте та безпечне повітря в укриттях.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. WiFi датчик якості повітря Moes (Tuuya smart). (n.d.). WiseHome. URL: <https://wisehome.com.ua/ua/wifi-datchik-jakosti-povitrja-moes-tuuya-smart.html> (дата звернення: 07.05.2024).
2. Chaikovska, K. (2023, June 9). [ua] 5 найкращих рішень забезпечення якості внутрішнього повітря в 2022 році і не тільки[:] - European Business Association. European Business Association. URL: <https://eba.com.ua/5-najkrashhyh-rishen-zabezpechennya-yakosti-vnutrishnogo-povitrya-v-2022-rotsi-i-ne-tilky/> (дата звернення: 09.05.2024).

*Довганик Марія, здобувачі фахової передвищої освіти II курсу  
спеціальності «Дизайн середовища»  
науковий керівник Черноока Л. М.*

## **ОЗЕЛЕНЕННЯ ДАХІВ**

В сучасному світі більш ніж половина людства мешкає в містах, тому зростає роль хорошої урбаністики, практичної архітектури й ефективних, екологічних технологій будівництва.

Озеленення дахів — одна з передових технологій, що стає дедалі популярнішою в сучасному будівництві.

Історія зелених дахів налічує кілька десятків століть. Вперше дахи почали озеленювати в країнах зі суворим кліматом — в Скандинавії й на Фарерських островах. Для цього використовували дерен — поверхневий шар ґрунту, вкритий травами та мохом, переплетений живим та відмерлим корінням рослин, кореневищами багаторічних трав. Дерен слугував доступним теплоізолятором [1].

В 60-тих роках ХХ століття в Німеччині технологія зелених дахів знову стала популярною. Основна причина — потреба зменшити навантаження на стічну систему міст. Власники зелених дахів платили менші податки, адже вважалося, що озеленена покрівля вбирає дощову воду й зменшує навантаження на міську каналізацію. Економічний стимул став поштовхом для архітекторів та дизайнерів розробляти варіанти озеленення покрівель

Порівняно зі звичайною кривлею, зелені дахи мають ряд переваг:

1. Зменшують стік дощової води з даху в каналізацію.
2. Є хорошим теплоізолятором.
3. Позитивно впливають на мікроклімат міста, знижуючи температуру повітря у ньому та регулюючи вологість[5].

4. Покращують міську екосистему, адже є місцем проживання для рідкісних видів рослин, комах й птахів, що ними харчуються.
5. Добре виконують роль звукоізолятора.
6. Подовжують термін служби покрівлі, оскільки захищають конструкцію даху від ультрафіолетового випромінювання[4].
7. Є додатковим корисним простором для міщан, зокрема мешканців будинку.

За принципом функціонування зелені покрівлі поділяють на інтенсивні й екстенсивні. Екстенсивні дахи майже не потребують догляду, їх засаджують стійкими до посухи однорічними та багаторічними рослинами з неглибоким корінням. Залежно від місцевого клімату такі дахи матимуть різну висоту та масу крівельного килима, різнитимуться видовим складом рослин [2]. Інтенсивні покрівлі виконують функції повноцінного громадського простору - парку, саду чи дитячого майданчика. На них вирощують будь - які види рослин, включаючи дерева до 4-ох метрів у висоту, тож такі системи потребують регулярного професійного догляду. Висота й маса крівельного килима більша ніж в екстенсивних дахах, тут часто використовують системи поливу та водовідведення[3].

Сучасні технології дозволяють озеленювати не лише плоскі дахи, а й покрівлі з нахилом 20-30°. Будівельні компанії розробляють та постійно вдосконалюють власні протиерозійні системи, що розраховані на різні кути нахилу даху. До прикладу, є система, що може використовуватись при максимальному куті нахилу 50°. Існують також вертикальні системи озеленення - "Зелені стіни". Яскравим доказом того, що озелененню підлягають практично будь-які площини, є Каліфорнійська академія наук. Її дах є найбільшим у світі зеленим дахом, що також містить кулясті елементи[6].

Щоб озеленити дах вже збудованого дому, потрібно провести певні розрахунки, щоб визначити приблизне навантаження, яке може витримати дах. Це навантаження має дві складові: власна вага вже існуючої конструкції

даху та снігове навантаження для конкретного району. Власна вага даху на 1 м<sup>2</sup> дорівнює масі кожного шару крівельного килима на 1 м<sup>2</sup>. В середньому навантаження на дахи житлових будинків не перевищує 50 кг/м<sup>2</sup>. Це навантаження потрібно помножити на коефіцієнт 1.1 коефіцієнт запасу, що враховує різні непередбачувані ситуації.

Снігове навантаження є в нормах будівництва, воно різне для кожної області України. Для Тернопільської області — це 139 кг/м<sup>2</sup>. Коефіцієнт для снігового навантаження залежить від кута нахилу даху. Для односкатного даху цей коефіцієнт = 1.0. Для двоскатного він залежить від кута нахилу даху. Якщо кут нахилу менший, ніж кут 25 градусів, коефіцієнт = 1. Коефіцієнт при куті нахилу від 25 до 60 градусів = 1.25. При більшому куті нахилу снігове навантаження не враховується.

Повне навантаження на дах дорівнює сумі снігового навантаження й масі крівельного килима. Вирахувавши значення повного навантаження на дах у приватному будинку, збудованому згідно з цими нормами я дійшла висновку, що такий будинок не підлягатиме озелененню без додаткового укріплення даху, оскільки його дах не розрахований на додаткове навантаження від шару зі зеленим килимом.

Найкраще заселяти дах тими рослинами, які притаманні для конкретного регіону, адже вони вже звикли до місцевого клімату й не потребуватимуть особливого догляду, адже перебувають в природному для себе середовищі. Такий підбір рослин також може частково відновити вплив споруди на навколишнє середовище.

#### Підбір рослин для Тернопільської області

Багаторічні та дворічні рослини: чебрець, живучка, полин гіркий, обрієта, бадан, калюжниця, дзвоник, деревій, лаванда, гвоздика турецька, гвоздика трав'янка, крупка, гісоп, еріка, стокротки багаторічні, фіалки багаторічні, очиток, безсмертник, іберис вербейник, котовник (котяча м'ята), м'ята кучерява, лапчатка, первоцвіт, молодило, смолівка, суниці, безстебельна, вероніка.

Однорічні: мак, волошки, айстри, матіоли, ромашка лікарська, роман собачий.

Злакові: костриця, трясучка, тонконіг лучний, лисохвіст лучний.

Переваги зеленого даху у приватному будинку — хороша тепло- та шумоізоляція. В міському будівництві зелені дахи допомагають врегулювати стік дощових вод, завдяки добре підбраної рослинності допомагають зберегти підтримувати природні біоценози, оскільки стають місцем життя корисних видів комах, а також птахів, земноводних тощо. Зелений дах досить складна система, що включає шар гідроізоляції, кількшаровий субстрат для кореневої системи рослин і власне висаджені рослини. Підбір рослин, можливість вирощування дерев та кущів залежить від конструкції зеленого даху. Вибір рослин для озеленення даху повинен опиратися на види, що розповсюджені у конкретному регіоні. Основу даху мають складати багаторічні злаки та посухостійкі квіткові рослини, при бажанні їх можна доповнити однорічниками. Основний догляд за зеленими дахами це мінімальний полив під час посухи. Викосувати рослини на таких дахах не потрібно.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Вікіпедія: Озеленення дахів. URL: [https://uk.wikipedia.org/wiki/%D0%9E%D0%B7%D0%B5%D0%BB%D0%B5%D0%BD%D0%B5%D0%BD%D0%BD%D1%8F\\_%D0%B4%D0%B0%D1%85%D1%96%D0%B2](https://uk.wikipedia.org/wiki/%D0%9E%D0%B7%D0%B5%D0%BB%D0%B5%D0%BD%D0%B5%D0%BD%D0%BD%D1%8F_%D0%B4%D0%B0%D1%85%D1%96%D0%B2) (Дата звернення: 07.05.2024).
2. Green Roofs Technology: Extensive Green Roofs. URL: <https://greenrooftechnology.com/green-roof-finder/extensive-green-roof/> (Дата звернення: 14.05.2024).
3. Green Roofs Technology: Intensive Green Roofs. URL: <https://greenrooftechnology.com/green-roof-finder/intensive-green-roof/> (Дата звернення: 14.05.2024).



4. Wikipedia:Green roofs. [https://en.wikipedia.org/wiki/Green\\_roof](https://en.wikipedia.org/wiki/Green_roof) (Дата звернення: 13.05.2024).
5. Wikipedia:Urban heat island. URL: [https://en.wikipedia.org/wiki/Urban\\_heat\\_island](https://en.wikipedia.org/wiki/Urban_heat_island) (Дата звернення: 13.05.2024).
6. ZinCo: Сад на даху: як озеленюють покрівлі в Україні. URL: <https://zinco.com.ua/uk/blog/green-roof-ukraine-world> (Дата звернення: 07.05.2024).

*Дудун Юрій*

*здобувач фахової передвищої освіти I курсу*

*спеціальності «Комп'ютерна інженерія»*

*науковий керівник Чубей О.О.*

## **ІНТЕГРАЛ В ПОВСЯКДЕННОМУ ЖИТТІ ЛЮДИНИ**

Інтеграл – це математична операція, яка є оберненою до диференціювання. Інтеграл описує накопичену суму або об'єм певної величини, використовується для знаходження площ криволінійних фігур. Це означає, що зазначена математична операція дозволяє «збирати» величини, знаючи їх «швидкість зміни», тобто похідну.

Інтегралі тісно пов'язані з похідними, які описують швидкість зміни величин. Існує два основних типи інтегралів: визначений та невизначений. Визначений інтеграл використовується для обчислення площ, об'ємів та інших числових значень. Невизначений інтеграл знаходить первісну функцію, тобто функцію, похідна якої дорівнює даній.

Інтеграл є потужним інструментом, який використовується в різних сферах життя людини, від фізики та інженерії до фінансів та медицини. Їх можна використовувати для обчислення площ фігур під графіком функцій, об'ємів тіл, що мають складну форму, а також довжин, центрів мас, моментів інерції та багатьох інших характеристик фізичних об'єктів та математичних функцій.

Застосування інтегралів в житті людини надзвичайно широке. Так, у сфері фінансів інтеграл використовують для розрахунку відсоткових ставок, інвестицій, продуктивності праці та інших фінансових показників. У сфері економіки використання інтегралу доцільне для моделювання економічних

систем та аналізу ринків, а в економетриці – для оцінки ефективності економічних політик.

У сфері архітектури та дизайну інтеграл широко застосовують для визначення площі та об'ємів у процесі створення меблів, проєктування будівель, споруд та інших предметів дизайну. У фізиці інтеграл є важливим інструментом для обчислення швидкості, прискорення, роботи, енергії та інших фізичних величин. Активно використовується інтеграл в інженерії та комп'ютерних науках, зокрема, у процесі проєктування мостів, літальних апаратів, електронних пристроїв, розробки алгоритмів обчислення інтегралів, створення комп'ютерної графіки тощо.

Географія та геологія також є сферами застосування інтегралу, зокрема для визначення площ та об'ємів географічних об'єктів, включаючи профілі річок та форму місцевості. У біології таку математичну операцію як інтеграл використовують в ході моделювання росту популяцій, вивчення динаміки біологічних процесів, у медицині – для аналізу даних про поширення різноманітних хвороб, в процесі розробки методів діагностики та лікування, для розрахунку площі під графіком кривої температури або концентрації лікарського засобу в організмі з часом.

Запропонуємо наступні приклади задач із застосуванням інтегралів.

Задача зі сфери фінансів.

Компанія заготовляє та продає овочі. Швидкість отримання доходу (грн/год) від продажу продукції протягом дня змінюється і описується функцією  $f(t)$ , де  $t$  — час в годинах, починаючи з 0 годин.

Нехай функція швидкості отримання доходу виглядає так:  $f(t)=300+15t$ .

Задача: знайти загальний дохід компанії за 8 годин.

Використавши цю формулу  $P = \int_0^t f(t) dt$

Ми можемо дізнатись прибуток

$$P = \int_0^t f(t) dt = \int_0^8 f(300+15t) dt = \left( 300t + \frac{15t^2}{2} \right) \Big|_0^8 = 300 \cdot 8 + \frac{15 \cdot 8^2}{2} - \left( 300 \cdot 0 + \frac{15 \cdot 0^2}{2} \right) = 2880 \text{ грн}$$

Задача з архітектури та дизайну.

Знайти площу вітражного вікна, обмеженого параболою  $y=4-x^2$  та прямими  $y=-\frac{1}{2}x-1$ ,  $y=\frac{1}{2}x-1$  (рис. 1)

Для цього можемо знайти точки перетину даних ліній ( $x=-2$  та  $x=2$ ) і обчислити суму двох інтегралів:

$$\int_{-2}^2 (4-x^2) dx + \int_{-2}^2 \left( \frac{1}{2}x - 1 \right) dx = 10,67 + (-4) = 6,67$$

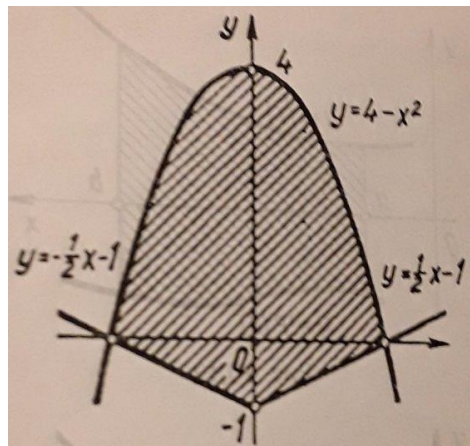


Рисунок 1 — Вітражне вікно

Задача з фізики.

Тіло рухається прямолінійно зі швидкістю  $v(t) = 20t - 3t^2$  (м/с). Обчисліть шлях, який пройшло тіло за інтервал часу від 2 до 5 с.

$$S = \int_{t_1}^{t_2} v(t) dt = \int_2^5 (20t - 3t^2) dt = \left( 20 \frac{t^2}{2} - 3 \frac{t^3}{3} \right) \Big|_2^5 = \dots$$

Отже, процес інтегрування – це фундаментальний інструмент математики, який здобув широке застосування в різних сферах нашого життя. Завдяки інтегралам ми можемо моделювати, аналізувати та розв'язувати складні задачі, що робить їх незамінними в науці, техніці, економіці та багатьох інших галузях. Їх вивчення є важливою частиною математичної освіти, адже вони дають можливість глибше зрозуміти світ навколо нас.

## СПИСОК ДЖЕРЕЛ ПОСИЛАННЯ

1. Застосування інтегралу URL: [buki.com.ua](http://buki.com.ua) (дата звернення: 12.06.2024р.)
2. Інтеграл URL: <http://surl.li/uoqum> (дата звернення: 11.06.2024р.)
3. Інтеграл у житті URL: <http://surl.li/uoqvк> (дата звернення: 12.06.2024р.)

*Ковальчук Яна, здобувач ОКР фахового молодшого бакалавра  
IV курсу спеціальності «Комп'ютерні науки»,  
науковий керівник Посвятовська О.Б.*

## **РОЗРОБКА СИСТЕМИ АВТОМАТИЗАЦІЇ МАРКЕТИНГУ ЛОКАЛЬНОГО БІЗНЕСУ В СФЕРІ SMM**

Social Media Marketing (SMM) – це різновид інтернет-маркетингу, який використовує соціальні медіа для просування брендів, залучення клієнтів та збільшення продажів. Основна мета SMM – створити віртуальну спільноту навколо бренду, залучити цільову аудиторію до взаємодії з контентом компанії та перетворити цю взаємодію на комерційні результати. SMM використовує соціальні мережі, щоб досягти цих цілей через публікації, рекламу, взаємодію з користувачами та аналітику даних.

Основні переваги SMM для автоматизації локального бізнесу наступні:

- широке охоплення мільярдів користувачів по всьому світу, дозволяючи бізнесам звертатися до великої аудиторії;
- можливість швидкої взаємодії з клієнтами дозволяє безпосередньо спілкуватися з клієнтами, що допомагає реагувати на їхні потреби та проблеми;
- наявність інструментів аналітики в соціальних медіа, які дозволяють бізнесу відстежувати ефективність своїх кампаній і вносити необхідні корективи.

Соціальні мережі стали невіддільною частиною сучасного життя. За даними Statista, станом на квітень 2024 року у світі найбільш популярними соціальними мережами є Facebook, YouTube, Instagram, WhatsApp та TikTok, що демонструє гістограма на рисунку 1, а їхня сумарна кількість зареєстрованих акаунтів складає близько 11,151 млрд користувачів [1].

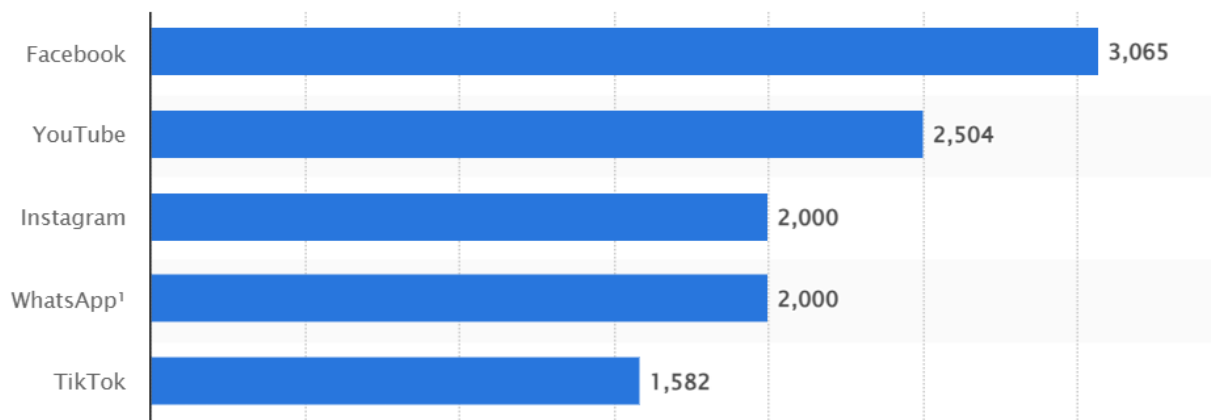


Рисунок 1 — Кількість активних користувачів соціальних мереж станом на квітень 2024 року (в мільйонах)

Ці дані підкреслюють важливість соціальних мереж як інструменту для бізнесу. Вони дозволяють бізнесу залучати нових клієнтів, збільшувати впізнаваність бренду, підтримувати зв'язок з клієнтами та швидко отримувати зворотний зв'язок про надані продукти чи послуги.

Одним з ключових аспектів використання соціальних мереж в маркетингу є автоматизація процесів, що дозволяє бізнесам оптимізувати свої зусилля та досягати кращих результатів. Прикладом таких мереж є Instagram та LinkedIn, які можуть бути використані для диджиталізації маркетингу локального бізнесу.

Instagram є однією з найпопулярніших соціальних мереж у світі з понад мільярдом активних користувачів. Серед переваг цієї платформи можна виділити те, що вона дозволяє бізнесам охоплювати широку аудиторію та залучати клієнтів за допомогою візуального контенту. Головним недоліком є необхідність постійного оновлення контенту. Це вимагає значних ресурсів та часу, особливо для невеликих бізнесів.

LinkedIn є професійною мережею, яка спеціалізується на бізнес-зв'язках та кар'єрному розвитку. Вона ідеально підходить для пошуку професійних

партнерів та клієнтів, а також активність на LinkedIn, включаючи публікації та взаємодію з іншими користувачами, сприяє зростанню репутації та впізнаваності бренду. Серед недоліків можна виділити те, що LinkedIn має суворі правила, і їх недотримання може призвести до видалення акаунта. Це вимагає від бізнесів уважного підходу до автоматизації, щоб уникнути порушень.

Автоматизація маркетингу в Інтернеті підвищує ефективність бізнесу, дозволяючи зосередитися на стратегічних завданнях та взаємодії з клієнтами. Такі соціальні мережі, як Instagram і LinkedIn пропонують унікальні можливості для автоматизації, кожна з яких має свої переваги та недоліки, залежно від специфіки бізнесу та його цілей. Важливо зберігати баланс між автоматизацією та персоналізованим підходом, щоб забезпечити успішну взаємодію та зростання бізнесу.

Для успішної автоматизації маркетингових процесів у соціальних мережах важливо врахувати специфічні вимоги бізнесу. Локальні SMM-компанії потребують платформ, які можуть задовольнити їх унікальні потреби та сприяти ефективній взаємодії з аудиторією. Основні вимоги до таких систем наступні:

- забезпечувати автоматичний обмін даними між користувачами на різних рівнях доступу;
- надати можливість автоматизованого планування та відстеження виконання завдань, що допомагає організувати робочі процеси та підвищити продуктивність;
- забезпечувати зручний процес реєстрації користувачів, наприклад, на основі їх Google акаунтів;
- надавати можливості для завантаження зображень, відео, написання постів та їх редагування, а також управління контентом;
- забезпечувати підтримку активного контакту з клієнтами за допомогою необхідних інструментів для взаємодії, такі як коментарі та повідомлення;
- забезпечення захисту персональних даних користувачів системи;



- наявність інтуїтивно зрозумілого інтерфейсу;
- сумісність з різними операційними системами.

Розробка соціальної мережі для локального бізнесу у сфері SMM повинна враховувати ці вимоги, щоб забезпечити ефективну автоматизацію маркетингових процесів. Важливо створити платформу, яка не тільки задовольняє потреби бізнесу, але й забезпечує зручність та безпеку для користувачів. Усі ці вимоги можуть бути виконані при застосуванні чистої архітектури для розробки системи.

Чиста архітектура – це підхід до розробки програмного забезпечення, який передбачає чітку модульну структуру системи, де кожен модуль відповідає за певну функцію і має чіткі межі взаємодії з іншими модулями. Така структура забезпечує кілька ключових переваг. По-перше, розділення бізнес-процесів на модулі дозволяє ефективно організувати роботу системи: кожен аспект бізнес-процесу, наприклад, автоматизація публікації контенту, може бути виділений в окремий модуль. Це сприяє легкості змін, оскільки кожен модуль може бути змінений або оновлений без впливу на інші частини системи, що дозволяє швидко адаптувати систему до нових вимог.

Крім того, чітка структура системи сприяє швидкому розвитку нових функцій. Завдяки модульності, нові функції можна розробляти та впроваджувати швидше, що підвищує загальну ефективність розробки. Також чиста архітектура зменшує ризики виникнення помилок та спрощує процес їх виправлення. Кожен модуль має свої чіткі межі та взаємодіє з іншими модулями через визначені інтерфейси, що зменшує ймовірність помилок і полегшує їх пошук та виправлення. Завдяки цим перевагам, чиста архітектура є ефективним підходом для розробки складних систем, включаючи соціальні мережі.

При розробці системи, також варто брати до уваги проблеми основних користувачів та методи їх вирішення, які зображено в таблиці 1.

Таблиця 1 – Таблиця формування задач для користувачів системи

| Потенційний користувач системи | Задачі, які потрібно вирішити   | Проблеми, які виникають   | Формування вирішення проблеми  |
|--------------------------------|---|---|--|
| Smm-спеціаліст                 | Пошук потенційних клієнтів, ведення комунікації, публікація власних проєктів та контенту, що притаманний соціальним мережам | Де знаходити потенційних клієнтів?<br>Який контент викладати?<br>Як налагодити комунікацію? | Знаходження потенційних клієнтів через публікацію контенту, комунікація із клієнтами через чат |
| Потенційний клієнт             | Знаходження smm-спеціаліста, перегляд контенту smm-спеціаліста  | Як знаходити smm-спеціаліста?<br>Яким чином переглядати контент smm-спеціаліста?            | Знаходження smm-спеціаліста через перегляд його контенту                                       |

Формулювання задач для користувачів системи автоматизації маркетингу визначає ключові функціональності, необхідні для забезпечення їх ефективної роботи під час використання системи. Врахування цих вимог дозволить створити інструмент, який відповідає потребам користувачів та сприяє успішному розвитку їх бізнесу.

На основі вимог щодо розробки та функціональної складової було створено прототип соціальної мережі для локального бізнесу у сфері SMM, інтерфейс якої можна побачити на рисунку 2.

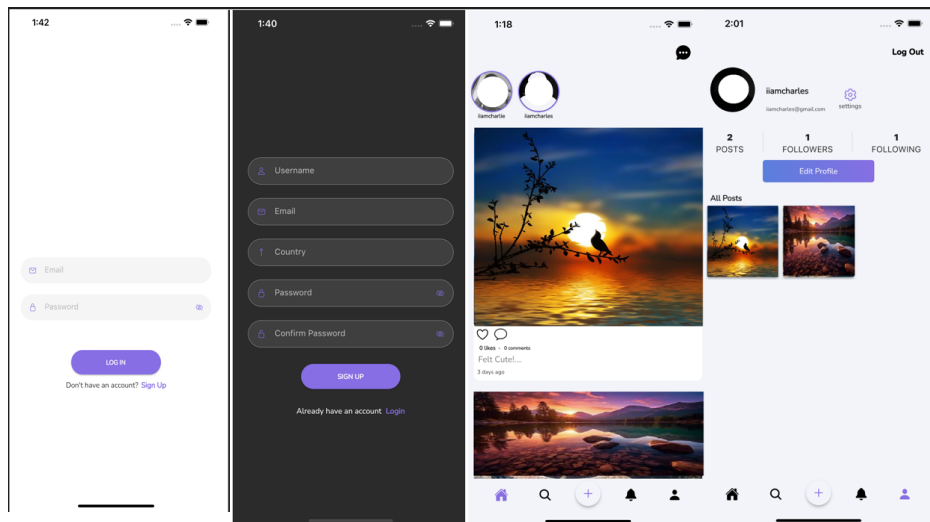


Рисунок 2 — Інтерфейс розробленої соцмережі

Розробка соціальної мережі є складним, але важливим завданням, яке сприятиме підвищенню ефективності та конкурентоспроможності локального бізнесу в сфері SMM.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Biggest social media platforms 2024 | Statista. *Statista*. URL: <https://www.statista.com/statistics/272014/global-social-networks-ranked-by-number-of-users/> (дата звернення: 02.05.2024).

*Лобас Сергій, здобувач ОКР Фаховий молодший бакалавр*

IV курсу спеціальності «Комп'ютерні науки»

науковий керівник

Івасьєв С.В.

## **АНАЛІЗ МЕРЕЖЕВОГО ТРАФІКУ ДЛЯ ВИЯВЛЕННЯ АНОМАЛІЙ ТА ЙОГО РОЛЬ В СФЕРІ КІБЕРБЕЗПЕКИ**

На сьогоднішній день спостерігається швидкий ріст обсягу мережевого трафіку, що призводить до ускладнення його структури. Аналіз мережевого трафіку стає все більш актуальним у зв'язку з розвитком мережевих технологій, впровадженням багатьох нових мережевих протоколів та збільшенням обсягу мережевих даних. Практичне застосування моніторингу мережі включає в себе виявлення проблем у роботі мережі, запобігання мережевим атакам і класифікацію трафіку. Моніторинг трафіку також необхідний для більш ефективної діагностики, оптимізації пропускнуої здатності каналів передачі даних, контролю якості зв'язку та розв'язання проблем у сфері кібербезпеки.

Аномалії в трафіку, такі як відмови[1], атаки, збої та інші види аномалій (рисунок 1), стали нормою в сучасних комп'ютерних мережах. Виявлення, діагностика та вирішення аномалій вчасно - це ключовий елемент повсякденної роботи мережі. Без такої можливості мережі не можуть працювати ефективно чи надійно. Точне виявлення та діагностика аномалій залежить від надійних та своєчасних даних, а також від встановлених методів ізоляції та ідентифікації аномалій на основі попереднього досвіду. Хоча було створено різноманітні інструменти для автоматичного сповіщення про відмови, виявлення аномалій автоматично загалом виявляється складним завданням. Важливим кроком у поліпшенні можливостей виявлення аномалій є ізоляція та характеристика їх ключових ознак.

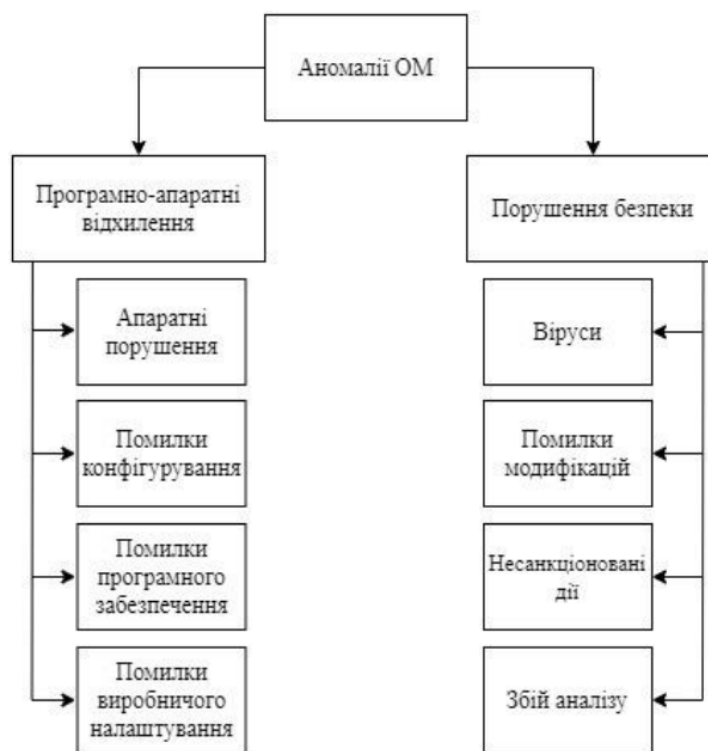


Рисунок 1 — Аномалії в мережі

Основні методи аналізу мережевого трафіку включають в себе використання систем виявлення вторгнень (IDS) та систем запобігання вторгнень (IPS). IDS моніторять мережу на наявність незвичайних або підозрілих активностей і сповіщають адміністраторів про можливі вторгнення або загрози безпеки. IPS, у свою чергу, виявляють шкідливі пакети або дії в мережі та блокують їх.

Крім того, аналіз пакетів мережевого трафіку на рівні даних є важливим інструментом для виявлення аномалій. Цей метод включає в себе моніторинг та аналіз самого трафіку для виявлення незвичайних зразків, які можуть вказувати на потенційні загрози або атаки. При аналізі такого трафіку звертають увагу на обсяг і частоту пакетів, незвичайні протоколи або порти, а також аномальні комбінації пакетів[2].

Нарешті, спеціалізовані інструменти та платформи для аналізу мережевого трафіку надають можливість адміністраторам моніторити, візуалізувати та аналізувати дані для виявлення аномалій та вразливостей.

Вони можуть включати функції автоматизації та інтелектуального аналізу для підвищення ефективності аналізу та виявлення потенційних загроз.

Розуміння та використання цих методів дозволяє адміністраторам мережі ефективно виявляти та реагувати на аномалії та потенційні загрози для забезпечення безпеки мережевого середовища.

Методи виявлення аномалій у мережевому трафіку виконують критичну роль у забезпеченні кібербезпеки, сприяючи вчасному виявленню та запобіганню потенційним загрозам для мережевих систем. Вони можуть бути розділені на кілька основних категорій, таких як статистичні методи, методи машинного навчання та гібридні підходи. (рисунок 2)



Рисунок 2 — Методи виявлення аномалій

Статистичні методи аналізу в основному спираються на порівняння фактичних даних з попередньо визначеними нормами або статистикою попередніх періодів для виявлення аномалій. Ці методи можуть виявити незвичайні зміни у шаблонах трафіку або вибої обсягу даних.

Статистичні методи виявлення аномалій у мережевому трафіку використовуються за допомогою тестів, особливо ефективно для точкових аномалій. Один із таких тестів - Z-оцінка, яка дозволяє визначити кількість стандартних відхилень від середнього значення для даного числа вибірки.

Наприклад, розглянемо ситуацію, коли сервер отримує запити від користувачів і час отримання кожного запиту може варіюватися. Для цього ми збираємо вибірку з п'яти часових міток отримання запитів:

10 мс, 12 мс, 14 мс, 11 мс, 13 мс.

Спочатку ми знаходимо середнє значення цих п'яти часових міток:

$$(10 + 12 + 14 + 11 + 13) / 5 = 12 \text{ мс.}$$

Потім ми розраховуємо дисперсію, що відображає ступінь відхилення значень від середнього:

$$D = [(10-12)^2 + (12-12)^2 + (14-12)^2 + (11-12)^2 + (13-12)^2] / 5 = (4 + 0 + 4 + 1 + 1) / 5 = 2 \text{ мс}^2.$$

Далі ми розраховуємо стандартне відхилення, яке є квадратним коренем з дисперсії:

$$\sigma = \sqrt{2} = 1.41 \text{ мс.}$$

Тепер ми можемо використати Z-оцінку для оцінки, наскільки віддалений кожен час отримання запиту від середнього значення:

$$Z = (X - \mu) / \sigma, \text{ де}$$

X - час отримання запиту,

$\mu$  - середнє значення,

$\sigma$  - стандартне відхилення.

Розглянемо, наприклад, час отримання запиту, який дорівнює 14 мс:

$$Z = (14 - 12) / 1.41 = 1.41.$$

Це означає, що час отримання цього запиту віддалений від середнього значення на 1.41 стандартних відхилень. Таким чином, ми можемо виявити, які запити прийшли з незвичайно великою затримкою в порівнянні зі звичайними запитами.

Методи машинного навчання використовують алгоритми, які навчаються на історичних даних для виявлення аномальних патернів. Ці алгоритми можуть розпізнавати аномалії, які не підпадають під звичайні шаблони поведінки мережі, та виявляти складні аномалії. Прикладом такого алгоритму є алгоритм Isolation Forest, що базується на ідеї випадкового

розбиття простору ознак для виявлення аномалій (рис. 3). Він визначається за допомогою відсічки ізольованих точок від нормальних даних, проводячи декілька стохастичних розбиттів. Після цього результати усереднюються для отримання остаточного результату. Цей алгоритм може розпізнавати аномалії різних типів, включаючи ізольовані точки з низькою локальною щільністю та малі кластери аномалій. Під час поділу даних виконується кілька рівнів розбиття, причому кожен рівень розділяється певну кількість разів. Позитивні розбиття продовжуються далі, тоді як негативні розбиття залишаються незмінними.

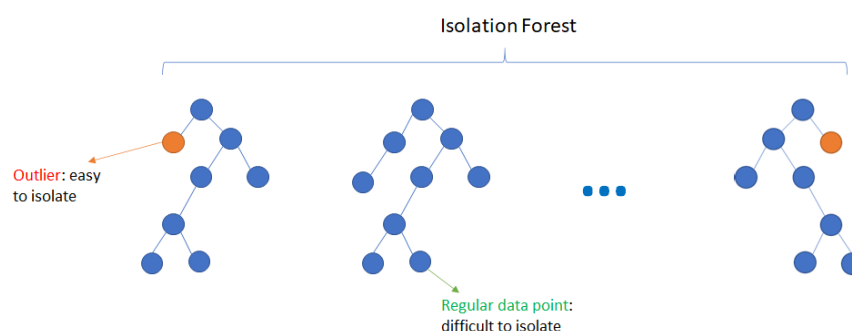


Рисунок 3 — Модель алгоритму машинного навчання для виявлення аномалій Isolation Forest

Гібридні підходи поєднують переваги статистичних методів та методів машинного навчання для поліпшення точності виявлення аномалій та зменшення кількості помилкових сигналів. Ці системи можуть використовувати статистичні методи для передварних фільтрацій даних перед тим, як подавати їх на вхід алгоритмам машинного навчання. Для демонстрації, обраний метод ітерацій.

Метод ітерації є чисельним методом для наближеного розв'язання систем рівнянь. Основна ідея полягає у визначенні послідовності наближених значень, які збігаються до точного розв'язку системи. Цей метод дозволяє знаходити значення коренів системи з заданою точністю шляхом ітеративного процесу. Одним з таких методів є метод Зейделя, що є одним із чисельних



методів ітераційного типу для розв'язання систем лінійних алгебраїчних рівнянь. Він використовується для знаходження наближених розв'язків систем рівнянь шляхом послідовного покращення наближених значень. У методі Зейделя нове значення кожної змінної обчислюється на основі найбільш актуальних значень інших змінних у поточному наближенні. Цей процес повторюється до тих пір, поки досягається задана точність або досягається задана кількість ітерацій. Характер збіжності методу Зейделя може змінюватися залежно від властивостей системи рівнянь та початкового наближення.(рисунок 4).

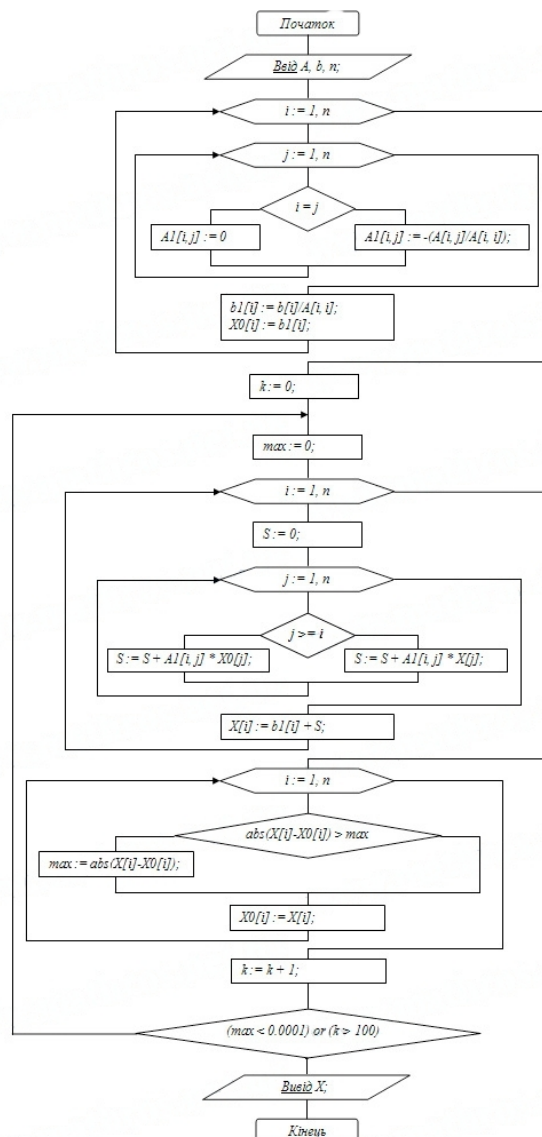


Рисунок 4 — Блок-схема ітераційного методу Зейделя

Аналіз мережевого трафіку в кібербезпеці є ключовим елементом виявлення потенційних загроз і атак на мережеві системи. Він допомагає виявляти аномальні патерни трафіку та надає можливість приймати відповідні заходи для запобігання кібератак. Методи аналізу мережевого трафіку використовуються для виявлення несправностей, незвичайних активностей та вразливостей в мережевій інфраструктурі. Ці методи можуть включати системи виявлення вторгнень (IDS), системи запобігання вторгнень (IPS), аналіз аномалій та виявлення зловмисного програмного забезпечення і атак з використанням мережі. Моніторинг мережевого трафіку важливий для ефективної діагностики, оптимізації пропускну здатності каналів передачі даних, контролю якості зв'язку та вирішення проблем у сфері кібербезпеки.

У роботі було проведено аналіз мережевого трафіку з метою виявлення аномалій та їх ролі у сфері кібербезпеки. Результати дослідження підтвердили важливість використання методів аналізу мережевого трафіку для виявлення потенційних загроз та кібератак.

Методи аналізу мережевого трафіку є важливим інструментом для моніторингу, виявлення та запобігання кіберзагрозам. Використання систем виявлення вторгнень, систем запобігання вторгнень, аналізу аномалій та інших спеціалізованих інструментів дозволяє ефективно реагувати на потенційні загрози та забезпечувати безпеку мережі.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. D. Jiang, J. Liu, Z. Xu and W. Qin, "Network traffic anomaly detection based on sliding window," *2011 International Conference on Electrical and Control Engineering*, Yichang, China, 2011, pp. 4830-4833, doi: 10.1109/ICCEENG.2011.6057677.
2. K. Limthong and T. Tawsook, "Network traffic anomaly detection using machine learning approaches," *2012 IEEE Network Operations and Management Symposium*, Maui, HI, USA, 2012, pp. 542-545, doi: 10.1109/NOMS.2012.6211951.

*Музика Михайло, здобувач ОКР фаховий молодший бакалавр  
III курсу спеціальності «Комп'ютерні науки»  
науковий керівник Сиротюк О. Б., Кульчинська Н. З.*

## ДОСЛІДЖЕННЯ ТА ВІЗУАЛІЗАЦІЯ ЕФЕКТИВНОСТІ АЛГОРИТМІВ СОРТУВАННЯ НА ОСНОВІ РЕАЛІЗОВАНОГО ПРОГРАМНОГО ЗАСТОСУНКУ

Алгоритми сортування є важливою деталлю любого програмного забезпечення чи комплексу, саме від ефективності цих алгоритмів залежить зручність та плавність користувацького досвіду.

Існує багато різних алгоритмів сортування, але зазвичай використовуються лише декілька, чому так? Все, тому що вони мають різну ефективність для різних наборів даних та задач. Одні ефективні при роботі з текстом, інші при роботі з числами. А інші алгоритми сортування відриваються від інших на певній кількості вхідних даних. Наприклад сортування «Insertion» буде досить ефективним при малій кількості елементів або коли дані вже частково відсортовані. Коли ж елементів багато то слід задуматись про використання ефективнішого алгоритму як от «Quick» який є лідером при великому наборі числових даних. На рис. 1 зображено порівняння алгоритмів сортування «Quick» та «Insertion».

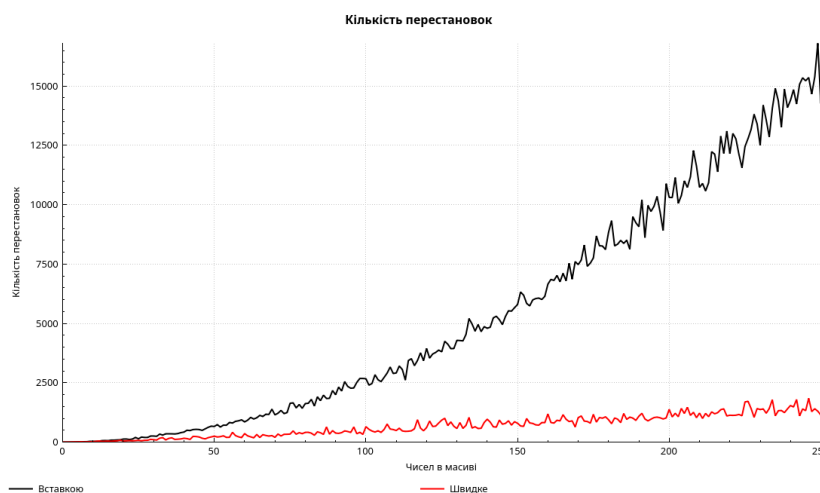


Рисунок 1 — Quick sort та Insertion sort

Для порівняння різних алгоритмів сортування було розроблено програмне забезпечення в якому можна тестувати ефективність різних алгоритмів сортування за часом та кількістю перестановок. Результати відображаються у вигляді інтерактивного графіка та таблиці.

Алгоритми сортування можна поділити на два типи:

- Ті що розділяють масив на менші підмасиви, сортують їх, а потім об'єднують, наприклад: Quick, Merge, Timsort;
- Прості які оперуються безпосередньо самим масивом, наприклад: Bubble, Insertion, Selection.

Декомпозиційні алгоритми, такі як Merge sort і Quicksort, зазвичай мають кращу асимптотичну складність ( $O(n \log n)$ ), але можуть мати більші накладні витрати через необхідність розбиття та об'єднання підмасивів.

Прості алгоритми, такі як Bubble sort і Insertion sort, мають гірші асимптотичні характеристики ( $O(n^2)$ ), але можуть бути ефективнішими для невеликих наборів даних або в певних ситуаціях.

Для тестування було вибрано та реалізовано декілька популярних алгоритмів:

- Bubble;
- Insertion;
- Selection;
- Cycle;
- Quick;
- Merge;
- Heap;
- Gnome.

З вибраних алгоритмів є як прості (Bubble, Insertion, ...) так і ті що використовують декомпозицію (Quick, Merge).

Для реалізації алгоритмів було обрано мову програмування C++ з компілятором, який використовує інтерфейс LLVM - clang. Це дозволить забезпечити кросплатформенність розробки та підвищити ефективність

компіляції. Розробка та тестування проводиться на операційній системі GNU/Linux, а саме: Arch Linux x86\_64 з ядром 6.8.4-273-tkg-bore-llvm.

Для розробки програмного інтерфейсу було вибрано бібліотеку Qt версії 6.7 та її компоненти:

- **Core**: містить основні класи та функції, які забезпечують базову функціональність для всіх застосунків на базі Qt. Він містить такі компоненти, як системи управління подіями, файлами, мережею, а також утиліти для роботи з датою, часом та іншими базовими даними;
- **Widgets**: надає широкий набір готових графічних елементів інтерфейсу користувача, таких як кнопки, меню, вікна, діалогові вікна та інші;
- **LinguistTools**: забезпечує набір інструментів та утиліт, які полегшують процес перекладу та локалізації застосунків на базі Qt, дозволяючи створювати програмне забезпечення, яке легко адаптується до різних мов та культурних контекстів.

А також бібліотеку QCustomPlot за допомогою якої можна малювати високопродуктивні графічні представлення.

Системою збірки було вибрано CMake замість QMake (є за замовчуванням для qt) тому, що вона є більш гнучкою. Вона дозволяє легко поєднувати Qt-проекти з іншими бібліотеками та інструментами.

Програму було спроектовано в модульному стилі, де зменшено залежності між її компонентами. Це дозволить легко модифікувати та масштабувати програму в майбутньому.

Самі основні класи та їхні призначення:

- **AlgorithmHandler**: керування доступними алгоритмами (реєстрація, видалення, ввімкнення/вимкнення);
- **SourceModelHandler**: керування моделями які надають дані, на основі яких проводяться тестування. Зберігає класи (моделі) які є дочірніми від *AbstractModelHandler*;

- **MainWindow**: головне вікно, відображає всю необхідну інформацію програми та реалізовує засоби керування нею.

Всі класи, які закінчуються на «Handler» є глобальними об'єктами реалізованими за допомогою патерну «Singleton», це зроблено для запобігання створенню декількох керуючих об'єктів класу в програмі.

Таким чином інтерфейс програми складається із головної панелі яка вміщає в собі графічні та табличні представлення результатів тестування, та панелі конфігурації тестування, зображено на рис. 2.

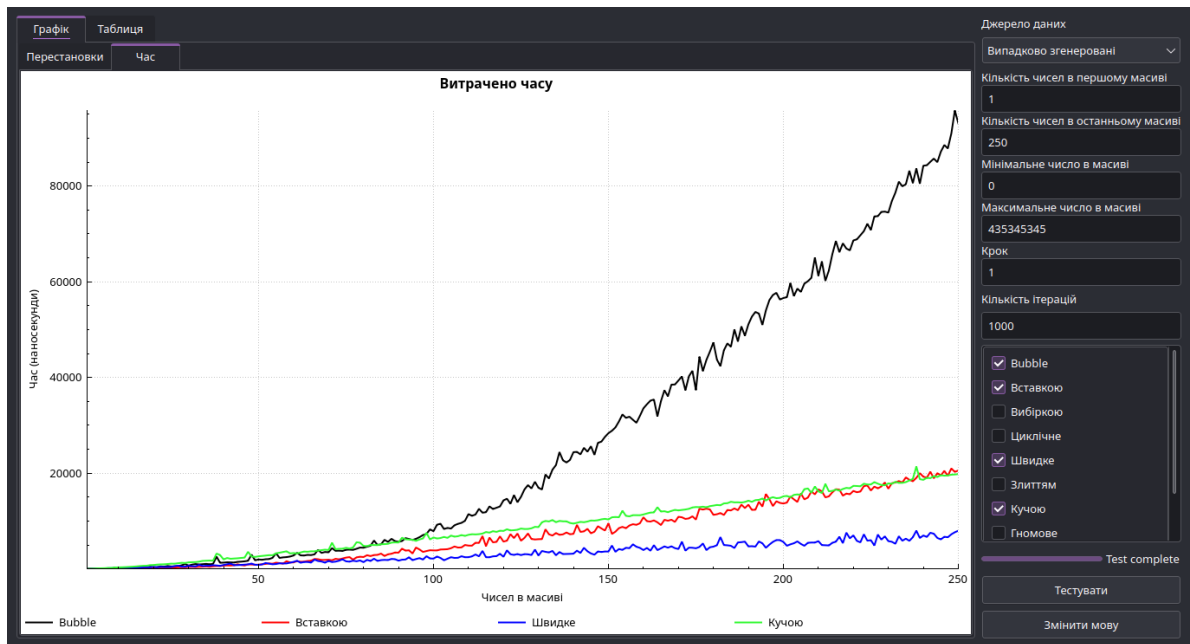


Рисунок 2 — Інтерфейс програми

На рис. 2 інтерфейс програми із часовим графіком порівняння ефективності сортування декількох популярних алгоритмів.

Тестування проводилось на основі випадково згенерованих даних. Було проведено щонайменше 1000 ітерацій, що дозволило мінімізувати вплив різних факторів.

Підбиваючи підсумки проведеного тестування всіх алгоритмів найефективнішим як за часом, так і за кількістю ітерацій став швидкий («Quick») алгоритм сортування, зображено на рис. 3.

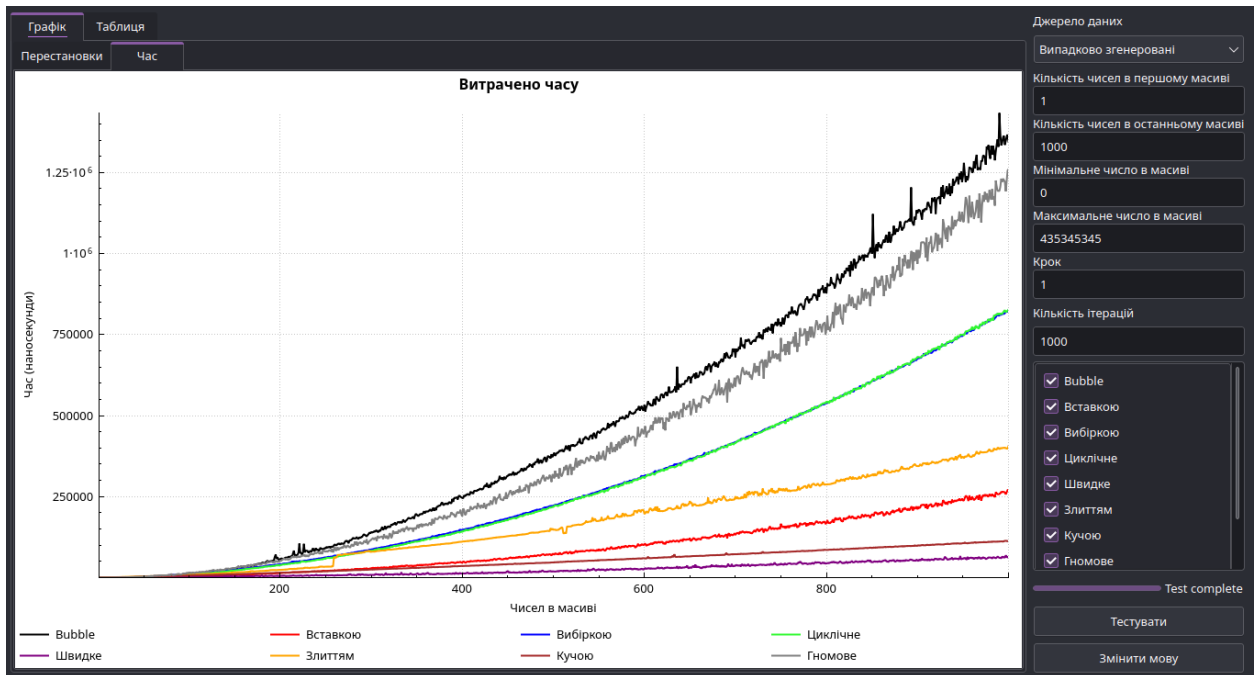


Рисунок 3 — Порівняння алгоритмів сортування за швидкістю

У роботі було проаналізовано та реалізовано основні популярні алгоритми сортування. Створено програмне забезпечення, яке допомогло протестувати ефективність різних алгоритмів сортування. Результати тестувань підтвердили, що:

- Алгоритм швидкого сортування (Quicksort) показав найкращу продуктивність для великих масивів даних, демонструючи найшвидше сортування;
- Алгоритм злиття (Mergesort) виявився ефективним для сортування масивів середнього розміру, забезпечуючи стабільну та надійну продуктивність;
- Алгоритм вставки (Insertion Sort) показав кращі результати для невеликих масивів даних, де він демонстрував найменшу кількість операцій порівняння та обміну;
- Алгоритм бульбашкового сортування (Bubble Sort) виявився найменш ефективним серед протестованих алгоритмів, особливо для великих масивів, через його високу асимптотичну складність.

Отримані результати дозволяють зробити висновок, що вибір алгоритму сортування залежить від розміру та характеристик вхідних даних, а також від вимог до швидкості та стабільності сортування.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Поняття складності алгоритмів – UA5.org. *UA5.org – Матеріали з інформаційних технологій*. URL: <https://ua5.org/algorithm/1884-ponyattya-skladnosti-algorytmiv.html> (дата звернення: 16.05.2024).
2. Учасники проектів Вікімедіа. Обчислювальна складність – Вікіпедія. *Вікіпедія*. URL: [https://uk.wikipedia.org/wiki/Обчислювальна\\_складність](https://uk.wikipedia.org/wiki/Обчислювальна_складність) (дата звернення: 16.05.2024).
3. Qt 6.7. *Qt Documentation | Home*. URL: <https://doc.qt.io/qt-6/> (date of access: 16.05.2024).
4. Sorting Algorithms - GeeksforGeeks. *GeeksforGeeks*. URL: <https://www.geeksforgeeks.org/sorting-algorithms/> (дата звернення: 16.05.2024).



*Музика Михайло, здобувач ОКР фаховий молодший бакалавр*

*III курсу спеціальності «Комп'ютерні науки»*

*науковий керівник*

*Кузик В. М.*

## **АРХІТЕКТУРА ТА ПРИНЦИПИ РОБОТИ СИСТЕМИ МІЖПРОЦЕСНОЇ ВЗАЄМОДІЇ D-BUS В ОПЕРАЦІЙНИХ СИСТЕМАХ GNU/LINUX**

В сучасних масштабованих операційних системах важливим є забезпечення всіх компонентів системою взаємодії, яка дозволяє їм ефективно обмінюватися даними та керувати один одним. Однією з таких систем міжпроцесної взаємодії є D-Bus яка використовується в операційних системах GNU/Linux.

D-Bus (Desktop Bus) є міжпроцесним комунікаційним механізмом розробленим як частина проєкту freedesktop.org, який дозволяє програмам обмінюватися повідомленнями та викликати методи одна одної. Він був розроблений для використання в операційних системах на базі Linux та інших Unix-подібних системах. Він широко використовується в багатьох Linux-додатках, таких як GNOME, KDE, systemd та інших, для забезпечення взаємодії між різними компонентами системи.

D-Bus, як міжпроцесний комунікаційний механізм, приділяє значну увагу безпеці, щоб забезпечити надійну та контрольовану взаємодію між різними програмами. Він використовує систему правил авторизації, яка визначає, які програми можуть отримувати доступ до певних об'єктів та методів. Ці правила можуть бути налаштовані як на рівні системи, так і на рівні окремих додатків, що дозволяє гнучко контролювати доступ. Це допомагає запобігти несанкціонованому доступу до критичних компонентів системи. D-Bus підтримує ізоляцію програм, що дозволяє обмежити доступ

програм до певних об'єктів та методів, запобігаючи поширенню вразливостей між різними компонентами системи.

На системному рівні, D-Bus надає механізми для налаштування політик безпеки, які визначають, які програми можуть отримувати доступ до певних об'єктів та методів. Ці політики можуть бути застосовані до критичних системних компонентів, забезпечуючи захист від несанкціонованого доступу. Наприклад, системні служби, які відповідають за управління живленням процесора, відеокарти чи іншими важливими аспектами системи, можуть бути захищені за допомогою таких політик.

На користувацькому рівні, D-Bus дозволяє налаштовувати права доступу для окремих додатків. Це означає, що користувачі можуть контролювати, які програми можуть взаємодіяти з певними об'єктами чи викликати певні методи. Це особливо корисно для додатків, які працюють з конфіденційною інформацією або мають доступ до критичних функцій системи.

Також він реалізує ізоляцію між самими користувачами. Коли користувач входить в систему, він отримує свій власний D-Bus сеанс (session bus), який ізолюваний від інших користувачів. Кожен сеанс має свій набір унікальних об'єктів, методів та сигналів, доступних лише в межах цього сеансу. Для отримання доступу до об'єктів, що належать іншим користувачам, потрібні спеціальні права, які зазвичай надаються лише адміністраторам або іншим привілейованим користувачам.

D-BUS використовує концепцію сервісів, яка передбачає унікальне місце розташування застосунків на шині. Коли програма запускається, вона реєструє один або кілька сервісів, якими вона буде володіти доти, поки самостійно не звільнить. До цього моменту жоден інший застосунок, що претендує на той же сервіс, не зможе його зайняти. Сервіси мають унікальні імена.

Коли програма закривається, її асоційовані сервіси також звільнюються, і D-BUS надсилає сигнал про закриття сервісу. Інші програми можуть

отримувати такі сигнали та реагувати на них відповідним чином. Таким чином, сервіси в D-BUS забезпечують унікальну ідентифікацію застосунків.

Сервіси вміщують в собі об'єкти, які являють собою абстрактні сутності, які містять інтерфейси.

Кожен об'єкт має унікальний шлях, який визначає його місце розташування в ієрархії об'єктів. Цей шлях складається з імен, розділених слешами, подібно до файлової системи.

Інтерфейси в D-BUS є аналогічними до об'єктів в об'єктноорієнтованому програмуванні. Вони являють собою абстрактні сутності, які можуть мати свої властивості (параметри), методи та сигнали. Назва інтерфейсу складається зі слів розділених крапками, наприклад «org.kde.PlasmaShell».

Приклад структури сервісу D-Bus зображено на рис. 1.

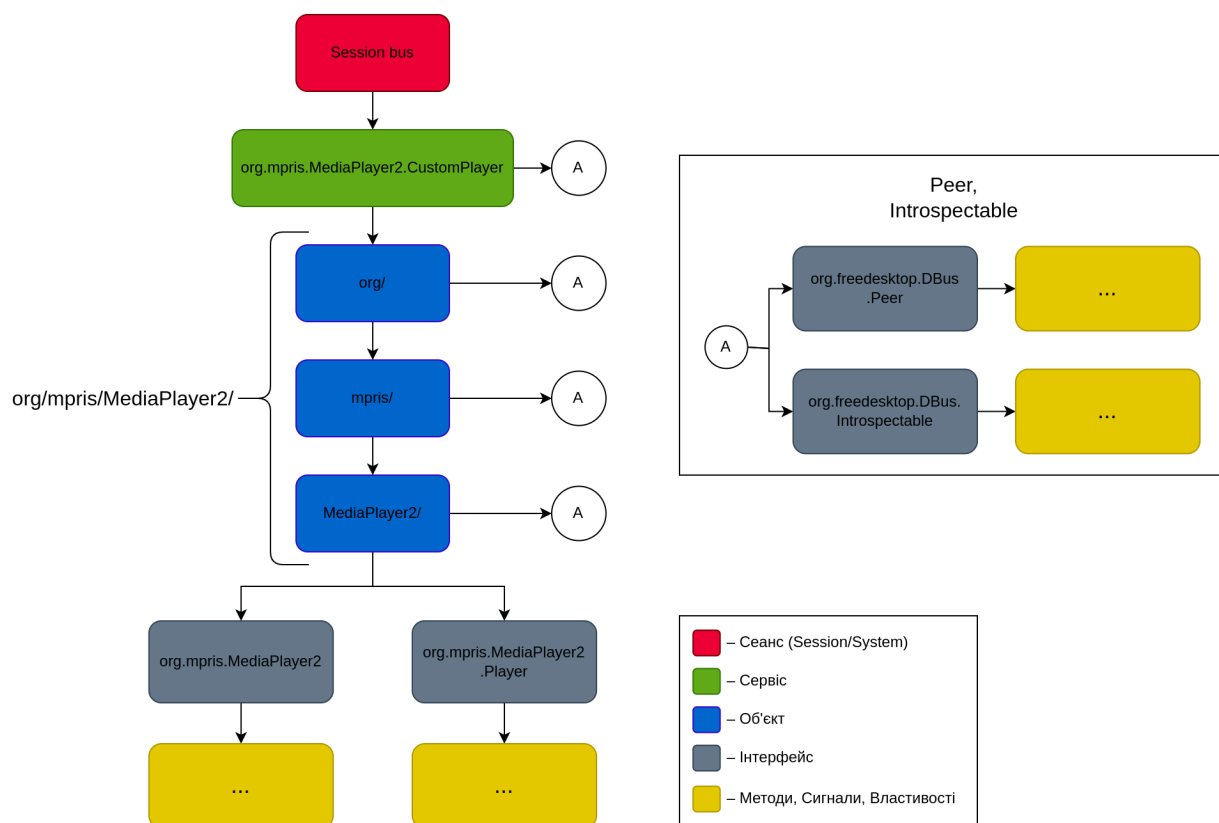


Рисунок 1 – Ієрархія об'єктів медіаплеєра в шині D-Bus

Таким чином основними об'єктами шини D-Bus є:

- Сервіси.
- Об'єкти.
- Інтерфейси:
  - Методи.
  - Параметри.
  - Сигнали.

Наведений приклад реалізує сервіс D-Bus необхідний для інтеграції музичного плеєра на операційних системах GNU/Linux. Також існують інтерфейси «*org.freedesktop.Dbus.Peer*» та «*org.freedesktop.Dbus.Introspectable*» (рис. 2), вони використовуються для забезпечення основних функцій взаємодії та самодокументування об'єктів та сервісів.

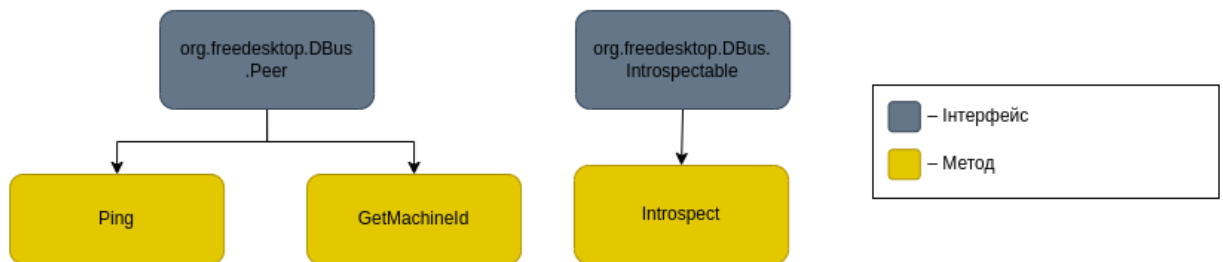


Рисунок 2 — Базові інтерфейси

Інтерфейс «*org.freedesktop.Dbus.Peer*» надає базові методи для взаємодії між D-Bus-клієнтами:

- «*Ping*»: використовується для перевірки доступності сервісу чи об'єкту;
- «*GetMachineId*»: повертає унікальний ідентифікатор машини, на якій виконується процес. Це може бути корисно для виявлення ситуацій, коли клієнти працюють на різних машинах у мережі.

Інтерфейс «*org.freedesktop.Dbus.Introspectable*» має лише один метод - «*Introspect*», який повертає XML-опис інтерфейсів та методів, що підтримуються об'єктом. Це дозволяє динамічно отримувати інформацію про

те, які методи, сигнали та властивості підтримуються певним об'єктом, що полегшує інтеграцію та використання сторонніх служб. Приклад такого опису:

```
<!DOCTYPE node PUBLIC "-//freedesktop//DTD D-BUS Object Introspection
1.0//EN" "http://www.freedesktop.org/standards/dbus/1.0/introspect.dtd">
<node name="/org/mpris/MediaPlayer2">
  <interface name="org.mpris.MediaPlayer2.Player">
    <method name="Seek">
      <arg name="Offset" type="x" direction="in"/>
    </method>
    <property name="CanGoNext" type="b" access="read"/>
  </interface>
</node>
```

Таким чином, DBus є дуже зручною системою міжпроцесної взаємодії, що забезпечує ефективний спосіб комунікації між різними процесами. Вона уніфікує взаємодію між додатками, надаючи єдину систему для передачі повідомлень та даних.

DBus підтримує механізми контролю доступу, що дозволяє налаштовувати права доступу для різних сервісів та клієнтів, забезпечуючи захист даних та операцій. Ця система також відзначається високою масштабованістю, підходячи як для простих локальних взаємодій, так і для складних мережових комунікацій. Завдяки цим характеристикам, DBus став основним інструментом для забезпечення міжпроцесної взаємодії в операційних системах GNU/Linux, та широко використовується в сучасних середовищах розробки.

У роботі було досліджено систему міжпроцесної взаємодії DBus, яка забезпечує ефективний та стандартизований спосіб комунікації між різними програмними компонентами. Аналіз включав об'єкти, шини, безпеку, інтерфейси, методи, властивості, сигнали та ієрархію об'єктів, а також логіку роботи цієї системи.

Дослідження підтвердило, що DBus є потужним інструментом для створення розподілених систем, який спрощує взаємодію між програмами та забезпечує їхню надійність та безпеку. Ця технологія відкриває широкі

можливості для розробників у побудові складних програмних рішень та систем з розподіленою архітектурою.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Contributors to Wikimedia projects. D-Bus - Wikipedia. *Wikipedia, the free encyclopedia*. URL: <https://en.wikipedia.org/wiki/D-Bus> (дата звернення: 24.05.2024).
2. D-Bus - ArchWiki. *ArchWiki*. URL: <https://wiki.archlinux.org/title/D-Bus> (дата звернення: 24.05.2024).
3. D-Bus Tutorial. *dbus*. URL: <https://dbus.freedesktop.org/doc/dbus-tutorial.html> (дата звернення: 24.05.2024).
4. D-Bus Viewer | Qt D-Bus. *Qt Documentation | Home*. URL: <https://doc.qt.io/qt-5/qdbusviewer.html> (дата звернення: 24.05.2024).
5. What Is DBus Practically Useful For? | Baeldung on Linux. *Baeldung on Linux*. URL: <https://www.baeldung.com/linux/dbus> (дата звернення: 24.05.2024).

*Москалюк Назарій, здобувач фахової передвищої освіти IV курсу  
спеціальності «Комп'ютерні науки»  
науковий керівник Кульчинська Н.З.*

## **ТЕХНОЛОГІЇ РОЗРОБКИ ОНЛАЙН-ТЕСТІВ**

У сучасному світі онлайн-тести стали невід'ємною частиною освітнього процесу, дозволяючи ефективно оцінювати знання здобувачів освіти. Технології розробки онлайн-тестів постійно вдосконалюються, пропонуючи нові можливості та функціонал для створення та проведення тестувань.

Метою дослідження є огляд існуючих інструментів та технологій для створення онлайн тестувань.

Онлайн-тести є потужним інструментом для оцінювання знань здобувачів освіти. Вони дозволяють ефективно оцінити рівень розуміння матеріалу та забезпечують швидке та зручне отримання результатів. Використання сучасних технологій розробки онлайн-тестів дозволяє створювати цікаві та ефективні тести, які сприяють покращенню якості навчання та навчального процесу в цілому.

Онлайн-тести мають низку переваг порівняно з традиційними паперовими тестами. По-перше, вони дозволяють економити час як вчителям, так і учням, оскільки процес оцінювання та надання результатів автоматизований. По-друге, онлайн-тести можуть бути інтерактивними, з використанням зображень, аудіо та відео матеріалів, що робить процес навчання більш цікавим та інтерактивним. Крім того, сучасні інструменти для тестувань надають користувачам досить широкі можливості для налаштувань, наприклад можуть бути доступні для проходження в певний час та з будь-якого місця, що дозволяє оптимально спланувати час та ефективно організувати освітній процес.

Для розробки онлайн-тестів існує велика кількість спеціалізованих платформ та програм. Одні з найпоширеніших інструментів це Google Forms, Moodle, Quizlet, Kahoot, Socrative, і багато інших. Ці платформи надають зручний інтерфейс для створення тестів, можливість додавання різноманітних типів питань (від відкритих до мультिवибору), а також автоматичне оцінювання результатів тестування.

Однією із популярних платформ для створення тестів є Moodle – модульне об'єктноорієнтоване динамічне навчальне середовище, яке називають також системою управління навчанням (LMS), системою управління курсами (CMS), віртуальним навчальним середовищем (VLE) або просто платформою для навчання, яка надає викладачам, учням та адміністраторам дуже розвинутий набір інструментів для комп'ютеризованого навчання. Зокрема, дана платформа надає типовий набір інструментів для створення онлайн-тестів.

Процес створення тесту подано на рисунку 1.

▼ Загальне

Назва ❗

Опис

Рисунок 1 — Створення тестувань на Moodle

Безперечно, платформа Moodle має цілий ряд переваг для створення тестів:



- Безкоштовність та відкритий код: Moodle є вільно розповсюджуваною платформою з відкритим кодом, що дозволяє використовувати її безкоштовно та вносити зміни в програмний код за необхідності.
- Зручний інтерфейс: Moodle має зручний та інтуїтивно зрозумілий інтерфейс, який дозволяє легко створювати тести та налаштовувати їх параметри.
- Різноманітність типів питань: Платформа підтримує різноманітні типи питань, такі як одиничний вибір, множинний вибір, відкрита відповідь тощо, що дозволяє створювати різноманітні тести.
- Можливість автоматичного оцінювання: Moodle дозволяє налаштовувати оцінки та автоматично оцінює результати тестування, що дозволяє значно економити час викладачам та студентам.
- Гнучкість налаштувань: Платформа надає можливість налаштування різних параметрів тесту, таких як тривалість, кількість спроб, використання підказок тощо.

Також слід зазначити певні недоліки використання Moodle для створення тестів:

Обмежені можливості в оформленні тестів: У порівнянні з іншими платформами для створення тестів, Moodle може мати обмежені можливості в оформленні тестів та їх зовнішньому вигляді.

Потреба в технічній підтримці: Для належного функціонування та підтримки Moodle може знадобитися наявність технічної підтримки та знань з адміністрування платформи.

Обмежені можливості для створення складних тестів: У деяких випадках створення складних тестів з великою кількістю різних типів питань та логічних зв'язків може бути складнішим у Moodle порівняно з іншими платформами.

Отже, хоча Moodle має багато переваг для створення тестів, платформа також має певні недоліки, які варто врахувати при розробці онлайн тестів.

Оскільки доступні онлайн-платформи не завжди надають користувачам достатньо гнучкий інструментарій для створення авторських тестів, необхідно розглянути можливості реалізації власного сервісу з необхідним функціоналом та достатньою кількістю налаштувань, із врахуванням потреб цільової аудиторії.

Для створення онлайн платформ для тестувань використовуються різні технології, залежно від потреб користувачів та необхідної функціональності платформи. Перш за все це вебтехнології (HTML, CSS, JavaScript) – більшість онлайн платформ для тестувань базуються на саме цих веб-технологіях для створення користувацького інтерфейсу та взаємодії з користувачем.

Для зберігання тестових питань, відповідей та результатів тестувань використовуються різні типи баз даних (MySQL, PostgreSQL, MongoDB), що дозволяють ефективно організовувати та зберігати великі обсяги даних.

Також можуть бути використані фреймворки (Django, Flask, Ruby on Rails). Використання фреймворків дозволяє швидко створювати та розгортати вебдодатки для тестування, забезпечуючи гнучкість та розширюваність системи.

Для інтегрування платформ для тестувань з іншими системами, такими як системи управління навчанням (LMS) або електронні журнали використовуються API (Application Programming Interface).

Ці технології допомагають створювати потужні та ефективні онлайн платформи для тестувань, які відповідають сучасним вимогам навчання та розвитку.

Для реалізації онлайн тестів також важливо розглянути роль баз даних, в яких безпосередньо зберігається набір питань та відповідей, а також відповідних налаштувань тестів. Загалом, база даних — це набір упорядкованих даних, які можна легко зберігати, сортувати, отримувати та шукати. Найчастіше з метою реалізації тестувань використовуються реляційні бази даних.

Реляційні бази даних [3] – це бази даних розташовані в рядках і стовпцях, зберігають і надають дані в кількох таблицях, а також дозволяють ідентифікувати дані та отримувати доступ до них у співвідношенні. Усі реляційні бази даних використовують мову запитів SQL. Microsoft SQL Server [4] є прикладом системи керування реляційною базою даних (СУБД).

Для маніпулювання даними в реляційних СУБД використовується мова структурованих запитів SQL. Мова SQL була розроблена IBM на початку 1970-х років і стала комерційно доступною у 1979 році. Дана мова визнана у всьому світі як стандартна мова для керування реляційною базою даних (RDBMS). В ній використовуються набори ключових слів для отримання даних із баз даних.

Щодо застосування вебтехнологій, то базовим умінням звичайно є знання мови HTML – мова гіпертекстової розмітки, яка є необхідною для створення вебсайтів. Дана мова складається з ключових слів – тегів, які є відносно легкими для розуміння і забезпечують розміщення різних елементів на вебсторінці та взаємодію з ними користувачів [5].

Створення вебсторінки для тестувань передбачається використання групи тегів form.

Форми (form) використовуються для введення даних від користувача. Їх можна використовувати для реєстрації, входу на вебсайт або для створення свого профілю на вебсайті тощо. Інформація, яку можна зібрати з форми, може бути різного типу. Дані форми обробляються серверною частиною, про яку розповідатиметься пізніше. Існують різні елементи форми, які можна використовувати, як-от текстові поля, текстові області, випадаючі списки, прапорці, перемикачі тощо.

Тег, який використовується для створення форми, це тег <form>, він має такий синтаксис:

```
<form action="form-url" method="form-method">  
    // різні елементи форми  
</form>
```

Наступні елементи HTML можна використовувати з елементом форми [7]:

- Input. Це один із найбільш використовуваних елементів форми. Він використовується кількома способами залежно від типу поля, яке ми хочемо визначити.

- Label. Використовується для визначення мітки поля, яке ми визначили. Це корисно, коли користувач намагається зосередитися на елементі введення. Тому ми використовуємо елемент label з елементом input.

- Select. Використовується для створення розкривного списку. Це дозволяє користувачам вибирати скільки завгодно параметрів із списку.

- Text area. Використовується для створення багаторядкового текстового поля. Це використовується в основному у формі для збору відгуків і коментарів. Розмір текстової області можна вказати певними атрибутами.

Button. Використовується для створення кнопки, яку можна натиснути.

Для опрацювання даних зазвичай використовуються засоби мови JavaScript. Це динамічна, об'єктноорієнтована прототипна мова програмування. Реалізація стандарту ECMAScript. Найчастіше використовується для створення сценаріїв вебсторінок, що надає можливість на боці клієнта (пристрої кінцевого користувача) взаємодіяти з користувачем, керувати браузером, асинхронно обмінюватися даними з сервером, змінювати структуру та зовнішній вигляд вебсторінки.

Використовуючи все вище зазначене, як приклад можна навести наступний код HTML для опрацювання даних форми:

```
<label for="question1" class="form-label">2+2</label>
  <input type="text" class="form-control" name="question1"
id="question1"/>
      <small id="helpId" class="form-text
text-muted">Цифра</small>

  <label for="question2" class="form-label">Перекладіть з
англійської "Hello"</label>
  <input type="text" class="form-control" name="question2"
id="question2"/>
```

```

        <small id="helpId" class="form-text text-muted">Слово з
великої літери</small>
        <label for="question3" class="form-label">2+2=4?</label>
        <input type="text" class="form-control" name="question3"
id="question3"/>
        <small id="helpId" class="form-text text-muted">Так або
Hi</small>
        <!-- Кнопки перемикання -->
        <button type="button" class="btn btn-primary"
id="prevQuestion">Попереднє питання</button>
        <button type="button" class="btn btn-primary"
id="nextQuestion">Наступне питання</button>
        <button type="button" class="btn btn-primary"
onclick="checkAnswers()" data-bs-toggle="modal" data-bs-
target="#exampleModal">Завершити </button>

```

Аналогічний код на JavaScript виглядатиме наступним чином:

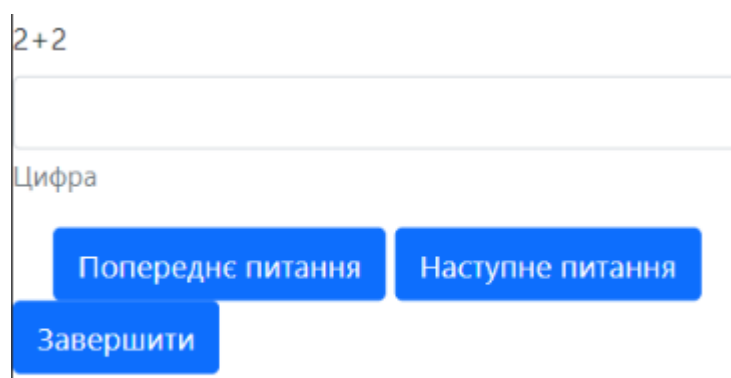
```

function checkAnswers() {
    var answer1 = document.getElementById('question1').value;
    var answer2 = document.getElementById('question2').value;
    var answer3 = document.getElementById('question3').value;
    var correctAnswers = 0;
    if (answer1 === '4') {
        correctAnswers++;
    }
    if (answer2.toLowerCase() === 'привіт') {
        correctAnswers++;
    }
    if (answer3.toLowerCase() === 'так') {
        correctAnswers++;
    }
    var resultMessage = 'Кількість правильних відповідей: '
+ correctAnswers;
    document.getElementById('resultMessage').innerHTML =
resultMessage;

    // Очистка полів вводу
    document.getElementById('question1').value = '';
    document.getElementById('question2').value = '';
    document.getElementById('question3').value = '';
}

```

На рисунку 2 подано візуальне представлення елементів керування для опрацювання даних форми.



The image shows a web form interface. At the top, there is a text input field containing the mathematical expression "2+2". Below the input field is a label "Цифра" (Digit). Underneath the label are three blue buttons with white text: "Попереднє питання" (Previous question), "Наступне питання" (Next question), and "Завершити" (Finish).

Рисунок 2 — Вигляд форми

Ця вебсторінка є прикладом простого тестування, яке складається з трьох питань. JavaScript, після введення даних, отримує їх та обробляє. Після обрахунку кількості правильних відповідей, сайт виводить користувачу результат.

Якщо ж використовувати базу даних, потрібно використовувати декілька базових SQL запитів, таких як SELECT, що вибирає необхідні дані з бази даних та INSERT INTO що вводить дані в таблиці. Ось приклади кожного з них:

```
SELECT * FROM attempts_to_pass_topic_test WHERE user_id = ?
```

```
INSERT INTO Customers(first_name, last_name, age, country)VALUES('James', 'Bond', 48, 'USA');[8]
```

Дослідивши різні підходи до створення тестувань – засобами онлайн-платформ, зокрема Moodle, або так званої «ручної» розробки з використанням мови розмітки та програмування, можна зробити висновки, що всі перелічені інструменти є досить ефективними та зручними для створення і проведення онлайн тестувань. Отже, для створення онлайн тестів можна використовувати різні технології та платформи, залежно від потреб користувачів та функціональності системи, при цьому потрібно враховувати цілі тестування та вміння користувача.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Новий рівень твого онлайн-курсу. *Блог: вебсайт.*  
URL: <https://myownconference.com/blog/uk/onlajn-testuvannya-ta-ankety/> (дата звернення: 12.05.2024).
2. Огляд основних SQL запитів. *ITVDN: вебсайт.*  
URL: <https://itvdn.com/ua/blog/article/m-sql> (дата звернення: 12.05.2024).
3. Elements of a form Tag - GeeksforGeeks. *GeeksforGeeks: вебсайт.*  
URL: <https://www.geeksforgeeks.org/elements-of-a-form-tag/> (date of access: 12.05.2024).
4. Hayes A. HyperText markup language (HTML): what it is, how it works. *Investopedia: вебсайт.*  
URL: <https://www.investopedia.com/terms/h/html.asp> (date of access: 12.05.2024).
5. HTML forms. *W3Schools Online Web Tutorials: вебсайт.*  
URL: [https://www.w3schools.com/html/html\\_forms.asp](https://www.w3schools.com/html/html_forms.asp) (date of access: 12.05.2024).
6. INSERT INTO в SQL - aCode. *aCode: вебсайт.*  
URL: <https://acode.com.ua/insert-into-sql/> (дата звернення: 12.05.2024).
7. What is SQL and what is it used for? - FutureLearn. *FutureLearn: вебсайт.*  
URL: <https://www.futurelearn.com/info/blog/what-sql-used-for> (date of access: 12.05.2024).
8. Windows server 2022. *Майкрософт: вебсайт.*  
URL: <https://www.microsoft.com/ru-ru/windows-server> (дата звернення: 12.05.2024).

*Нищота Юрій, здобувач фахової передвищої освіти IV курсу  
спеціальності «Комп'ютерні науки»  
науковий керівник Кульчинська Н.З.*

## **ТЕХНОЛОГІЇ РЕАЛІЗАЦІЇ КРОСПЛАТФОРМЕНОСТІ WEB- ЗАСТОСУНКІВ**

Під час розробки застосунків досить часто виникає питання як саме програмний продукт повинен застосовуватись та які обрати технології для його реалізації. Якщо завданням є реалізувати застосунок, який працюватиме на найвідоміших операційних системах, наприклад, сімейства Windows, можна застосувати певні технології, які будуть працювати на Windows, але не будуть працювати, наприклад на MacOS. Але якщо передбачається застосування додатка і на інших ОС, необхідно використовувати набір технологій для його реалізації, які забезпечать коректну роботу застосунку на будь-якій програмній платформі. Допомогти у вирішенні цієї проблеми можуть фреймворки для створення кросплатформених додатків.

Фреймворк – це готові інструменти, технології, які полегшують створення програмного забезпечення. Отже, кросплатформений фреймворк – це, як раніше було згадано, технології на інструменти, завдяки яким можна створити програмне забезпечення під практично будь-яку операційну систему: Windows, MacOS, Linux тощо.

Фреймворки, які досліджуються, працюють лише з мовою програмування JavaScript – що є зараз базовим інструментом для створення вебзастосунків. Також варто зазначити, що для того, щоб працювати з цими бібліотеками бажано мати навички використання фреймворку ReactJS, що є практично одним з найвідоміших фреймворків JavaScript, який спрощує створення вебпродуктів. ReactJS – це бібліотека яка вирішує проблему



створення користувацьких інтерфейсів та проблему часткового оновлення вмісту односторінкових застосунків.

Кросплатформенних бібліотек на даний момент створено досить велику кількість і кожен обирає, що йому до вподоби: ElectronJS, WebView\_Deno, DeskGap, NeutralinoJS і ще дуже багато інших.

Neutralinojs — це легкий і портативний кросплатформенний фреймворк для розробки програм. Він дозволяє розробляти програми за допомогою JavaScript/TypeScript, HTML і CSS [1].

Для цього фреймворку можна виділити деякі переваги та недоліки [2].

Переваги:

- проста розробка програмного забезпечення;
- містить базовий набір функцій для керування програмним забезпеченням.

Недоліки:

- повільність та велике споживання ресурсів;
- необхідність встановлювати багато різноманітних модулів.

В цілому цей фреймворк є доволі відомий серед зацікавлених користувачів, тому на ньому можливо створити прості або середньої складності програми.

Наступний фреймворк є не гіршим, називається NW.js — це JavaScript-фреймворк, який дозволяє створювати кросплатформні настільні програми для Windows, MacOS та Linux за допомогою вебтехнологій на базі Node.js та V8 [3, 4].

Переваги:

- широко використовується для створення сайтів та ігор;
- можливість швидко і просто зробити проект кросплатформним;
- можливість зробити додаток дуже якісним та продуктивним.

Недоліки:

- великий розмір програми;
- мала стабільність.

Завдяки тому, що даний фреймворк створений на базі NodeJS, також одному з базових фреймворків JavaScript, можна швидко та легко зрозуміти код та синтаксис цієї бібліотеки.

Electron — фреймворк, розроблений GitHub. Дозволяє розробляти різні графічні застосунки для настільних операційних систем за допомогою вебтехнологій. Фреймворк містить в собі Node.js для роботи з back-end і бібліотеку рендерингу із Chromium [5].

Цей фреймворк можна виділити лідером серед всіх інших, він є практично найвідомішим серед своїх конкурентів та найпрактичнішим. Також для нього можна визначити такі переваги та недоліки.

Переваги:

- повторне використання коду;
- можливість використати інші технології, щоб забезпечити гнучкість і якість продукту.

Недоліки:

- проблеми з пам'яттю та з розміром програми;
- продуктивність додатків.

Як приклад застосування фреймворків для розробки кросплатформених застосунків було реалізовано програмне забезпечення з використанням Electron.JS. В якості прикладу розроблено простий статичний сайт, який створений з використанням HTML/CSS. Для реалізації кросплатформенності написано такий програмний код:

```
const { app, BrowserWindow } = require("electron");

const createWindow = () => {
  const win = new BrowserWindow({
    width: 800,
    height: 600,
  })
  win.loadFile('src/index.html');
}
app.whenReady().then(() => createWindow());
app.on("window-all-closed", () => app.quit());
```

Для того, щоб використати цей код потрібно зініціювати прості пакети node після чого завдяки команді 'npm i electron' встановлюємо саму

бібліотеку. Далі потрібно створити файл, для прикладу `index.js`, де буде описаний весь код який був вказаний вище. В ньому імпортується бібліотека ElectronJS, створюється генерація додатка та визначається розмір вікна. Також вказується з якої саме сторінки завантажувати додаток і декілька перевірок на те, чи додаток запущений або вимкнений.

Результат подано на рисунках 1 та 2:

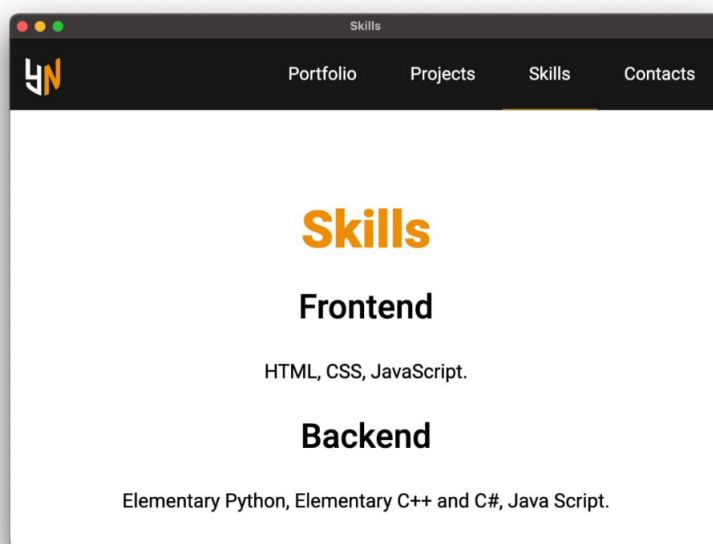


Рисунок 1 — Приклад використання фреймворку ElectronJS

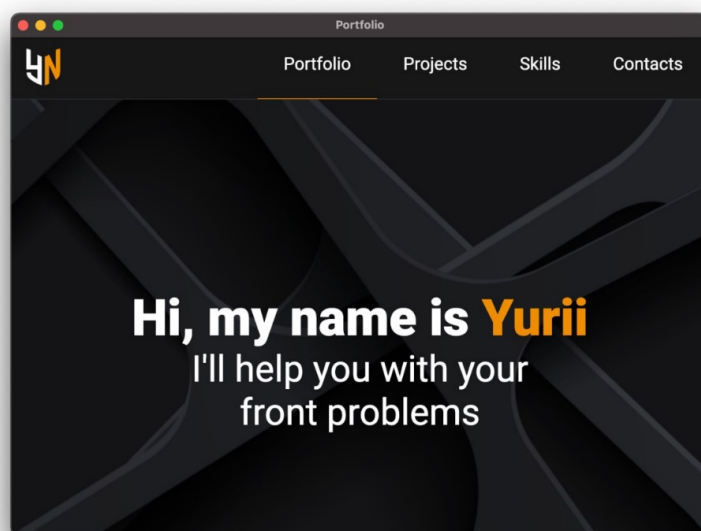


Рисунок 2 — Приклад використання фреймворку ElectronJS

Потрібно враховувати, що імпортовані методи в кодї є не лише одними в бібліотеці та можна застосувати ще інші та цікавіші методи, які зроблять додаток більш привабливим, оптимізованим та якісним.

Кросплатформні фреймворки для розробки вебзастосунків мають численні переваги, які роблять їх привабливими для розробників і бізнесу. Вони дозволяють створювати одну кодову базу, яка може бути використана для розробки різних платформ, таких як Android, iOS, веб і навіть десктоп. Звичайно, розробка кросплатформних додатків є значно ефективнішою, оскільки не потрібно писати окремий код для кожної операційної системи окремо, де буде працювати застосунок. Це дозволяє заощадити час і гроші на розробці та підтримці додатків.

Більшість кросплатформних фреймворків надають зручні інструменти для розробників, такі як готові методи і різноманітні бібліотеки, що спрощують розробку і полегшують підтримку додатків. Окрім того, використання одного мовного стеку для розробки різних платформ спрощує навчання та підтримку додатків. Кросплатформні фреймворки мають свої особливості та обмеження, але загалом вони можуть значно підвищити ефективність розробки вебзастосунків.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Інформаційна стаття використання NeutralinoJS. *Slant.com*: вебсайт. URL: <https://www.slant.co/options/39034/~neutralinojs-review> (дата звернення 11.05.2024).
2. Інформаційна стаття про переваги та недоліки NeutralinoJS від Shalitha Suranga. *Habr.com*: вебсайт. URL: <https://habr.com/en/articles/443572/> (дата звернення 11.05.2024).
3. Вікіпедія про кросплатформний фреймворк NW.js. *Wikipedia.com*: вебсайт. URL: <https://ru.wikipedia.org/wiki/NW.js> (дата звернення 12.05.2024).
4. Особливості використання фреймворку NW.js. *Packt.com*: вебсайт. URL:

<https://subscription.packtpub.com/book/web-development/9781785280863/1/ch01lv11sec09/features-and-drawbacks-of-nw-js> (дата звернення 12.05.2024).

5. Документація використання Electron. *ElectronJS.org*: вебсайт. URL: <https://www.electronjs.org/docs/latest/tutorial/quick-start> (дата звернення 12.05.2024).

*Ольховецький Андрій,  
здобувач фахової передвищої освіти  
IV курсу спеціальності «Комп'ютерна інженерія»,  
науковий керівник Павлюсь Василь Петрович*

## **ПРИСТРІЙ АВТОМАТИЗОВАНОГО КЕРУВАННЯ ВІДДАЛЕНОЮ СИСТЕМОЮ СПОСТЕРЕЖЕННЯ**

Моніторинг руху об'єктів за горизонтом є важливою складовою сучасних систем безпеки, охорони, рятувальних операцій та наукових досліджень. Використання тепловізорів у цій сфері дозволяє виявляти об'єкти навіть в умовах низької видимості, таких як нічний час або погані погодні умови. Однак для ефективного використання тепловізорів необхідна висока точність і стабільність при наведенні на об'єкти. Традиційні методи ручного керування часто не забезпечують необхідного рівня точності та зручності. У зв'язку з цим, розробка розумного штатива для віддаленого керування тепловізором та створення мобільного додатку для управління цим пристроєм є актуальним завданням.

1. Необхідність автоматизованого керування тепловізорами для моніторингу руху об'єктів за горизонтом:

Тепловізійні системи є ефективним інструментом для виявлення і моніторингу об'єктів в різних умовах, але їх ефективність залежить від точності і стабільності керування. Ручні методи наведення тепловізора обмежують можливості спостереження, особливо на великих відстанях і у важкодоступних місцях. Автоматизоване керування за допомогою крокових двигунів та інтелектуальної платформи може значно підвищити точність і стабільність тепловізійних спостережень.

2. Розробка інтелектуальної платформи для керування тепловізором:

Для створення розумного штатива було обрано крокові двигуни, які забезпечують точне і плавне позиціонування тепловізора. Мікроконтролер STM32F10C8T6 використовується для керування цими двигунами. Платформа забезпечує автоматичне та дистанційне керування тепловізором, що дозволяє виконувати моніторинг без безпосередньої присутності оператора на місці. Живлення від павербанка забезпечує автономність роботи пристрою в польових умовах.

3. Розробка мобільного додатку для віддаленого керування тепловізором:

Мобільний додаток дозволяє оператору керувати тепловізором на відстані за допомогою планшета або смартфона. Додаток забезпечує інтуїтивно зрозумілий інтерфейс для віддаленого керування, передачі відеосигналу в режимі реального часу та збереження даних для подальшого аналізу. Використання бездротових технологій, таких як Wi-Fi або Bluetooth, забезпечує надійний зв'язок між мобільним додатком і платформою.

Необхідність автоматизованого керування тепловізорами

Тепловізійні камери забезпечують можливість виявлення об'єктів за допомогою теплового випромінювання, що робить їх незамінними у темний час доби або в умовах обмеженої видимості. Проте, ефективне використання тепловізорів потребує стабільного і точного наведення на об'єкт. Ручне керування, особливо на великих відстанях, може призводити до помилок і втрат важливої інформації. Автоматизоване керування тепловізором за допомогою крокових двигунів дозволяє досягти високої точності наведення, що є критичним для ефективного моніторингу руху об'єктів за горизонтом. Це особливо актуально для рятувальних операцій, де швидкість і точність виявлення об'єктів можуть врятувати життя.

Розробка інтелектуальної платформи для керування тепловізором

Основою розумного штатива є механічна платформа, оснащена кроковими двигунами, які забезпечують плавне і точне наведення тепловізора. Для керування цими двигунами використовується

мікроконтролер STM32F10C8T6, який забезпечує високу надійність і швидкодію. Платформа інтегрується з тепловізором і дозволяє автоматично змінювати його положення у відповідь на команди оператора, що надходять через мобільний додаток. Живлення від павербанка забезпечує автономність роботи, що особливо важливо для польових умов, де відсутні постійні джерела електроенергії. Така конструкція дозволяє проводити моніторинг в реальному часі, оперативно реагувати на зміни в оточенні та забезпечує високу ефективність роботи.

Розробка мобільного додатку для віддаленого керування тепловізором

Мобільний додаток є ключовим елементом системи, який забезпечує інтуїтивно зрозумілий інтерфейс для віддаленого керування тепловізором. За допомогою додатку оператор може в реальному часі керувати положенням тепловізора, переглядати відеосигнал, зберігати та аналізувати отримані дані. Використання бездротових технологій, таких як Wi-Fi або Bluetooth, забезпечує надійний зв'язок між додатком і платформою, дозволяючи оператору працювати на безпечній відстані від об'єкта моніторингу. Це особливо важливо в умовах небезпеки, де безпека оператора є пріоритетом. Додаток також надає можливість програмування маршрутів та автоматичного моніторингу визначених зон, що підвищує ефективність і зручність використання системи.

Розробка розумного штатива для віддаленого керування тепловізором та створення мобільного додатку для управління ним є важливим кроком у підвищенні ефективності моніторингу руху об'єктів за горизонтом. Інтелектуальна платформа з кроковими двигунами та мікроконтролером STM32F10C8T6 забезпечує високу точність і стабільність наведення, а мобільний додаток дозволяє оператору здійснювати керування на відстані, забезпечуючи безпеку і зручність роботи. Така система має широкі можливості застосування в різних галузях, від безпеки та охорони до рятувальних операцій та наукових досліджень, що робить її універсальним і потрібним інструментом в сучасних умовах.



*Пелех Мар'ян, здобувач ОКР фахового молодшого бакалавра  
III курсу спеціальності «Комп'ютерні науки»  
науковий керівник О.Б. Сиротюк*

## **SOFT SKILLS TA HARD SKILLS РОЗРОБНИКА ІГРОВИХ ПРОГРАМ ТА ЇХ СПЕЦИФІКА В УКРАЇНІ**

Протягом останніх десятиліть в Україні стрімко розвивається ІТ-галузь і зростає попит на кваліфікованих фахівців в цій галузі. Завдяки своїм фаховим компетентностям, українські фахівці заклали фундамент для становлення інформаційно-технологічного майбутнього України, відповідно наші ІТ-компанії ведуть успішну діяльність як в Україні, так і на міжнародній арені. Однак, статистика з відкритих джерел вказує на те, що найпопулярнішими галузями є розробка вебсайтів та додатків, а також галузі зв'язані з аналізом даних і систем. На РІП (розробку ігрових програм) припадає значно менший відсоток фахівців та обсяг прибутку.

На даний момент ринок ігрових програм в Україні слабо розвинений. Найбільший його зріст був зафіксований у 2013 році (300 млн доларів США). Однак, повномасштабне вторгнення 2022 року спричинило велике падіння обсягу ІТ-спеціалістів, в тому числі – з розробки ігрових програм, що змушує деякі компанії закриватися або набирати нові кадри.

Цілі даного дослідження включають:

- дослідити сферу розробки ігрових програм в Україні;
- визначити пріоритетність Soft Skills та Hard Skills для розробника ігрових програм;
- окреслити тенденції у сфері розробки ігрових програм в Україні.

Використано такі методи дослідження:

- спілкування з працівниками сфери;
- власна розробка ігрових програм;

- пошук в мережі Інтернет;
- відвідування бесіди про Soft Skills.

Було проведено розмову з українськими розробниками ігор, що працюють різний період часу (від кількох місяців до 5 років), а також з іноземним інді-розробником.

Деякі висновки було зроблено в процесі розробки власних ігор, а також спілкування з одногрупниками. Проаналізовано випадки роботи в команді й самотужки. Було переглянуто декілька вебсайтів, що пропонують статистику щодо сфери розробки ігрових програм. Матеріали для дослідження також отримано після проведення розмови про Soft Skills в ІТ з представником ІТ-компанії SaaSJet.

#### 1. Про сферу розробки ігрових програм в Україні

З отриманих матеріалів випливає, що сфера РІП в Україні стабільно розвивалася, беручи початок ще у 1970-х роках. Однак в останні роки великі компанії галузі потребують нових спеціалістів, адже велика їх частка влаштувалася на роботу в іноземних компаніях.

#### 2. Про пріоритетність Soft та Hard Skills

Часто люди, що збираються влаштуватися на роботу в сфері розробки ігрових програм (і в загальному в ІТ), вважають, що найважливішими є саме їх Hard Skills, проте згідно з проведеним дослідженням Soft навички є чи не більш пріоритетними для будь-якого працівника ІТ.

Саме Soft Skills дозволяють розробнику проводити комунікацію з колегами чи клієнтами. У більшості випадків Soft Skills є тим, що потрібно для успішного проходження співбесіди й подальшого спілкування в команді. Високі навички комунікації й роботи в групі дозволяють швидше розвивати Hard Skills, а також затвердити цінність для команди як людини.

Дослідження показало, що для інді-розробників (які розробляють ігри в командах з 1-9 осіб) менш важливі, наприклад, навички комунікації, адже в їхньому випадку успіх проекту залежить лише від них.

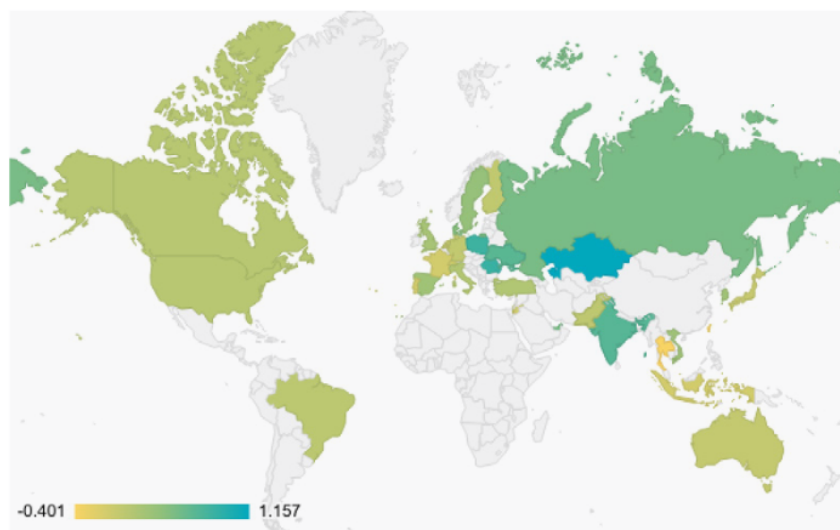
Натомість для розробників у команді критично необхідні Soft Skills. Часто вони допомагають вийти з ситуацій, у яких розробник не зміг би розібратися базуючись лише на своїх практичних вміннях.

### 3. Про тенденції розробки ігрових програм в Україні та світі

Оцінюючи сьогоднішні світові успіхи в розробці ігор, все більше молоді зацікавлене в створенні своїх власних ігрових розробок, або бути причетними до розробки можливих майбутніх шедеврів ігрової індустрії. На сферу РПІ значною мірою вплинуло усвідомлення власної національної ідентичності у великої частини українського суспільства після 2022 року й усвідомлення, що Україна повинна продовжувати власний розвиток в тому числі й в розробці ігор.

Останніми роками збільшилася кількість інді-розробників в Україні. Статистика показує, що за 2023 кількість українських інді-ігор зросла на 66.9% (рис. 1).

**Chart 2.3: Indies around the world are starting more games.**



New indie games have grown significantly in several countries over the past year:



Рисунок 1 — Статистика кількості інді-ігор

Виходить більше ігор, які призначені винятково для мобільних пристроїв. Їх випускають всі види студій, та найвищий показник у великих компаній. Там випуск винятково мобільних ігор зріс на 44% за рік.

Великі студії збільшують кількість багатоплатформних ігор. У 2022-му їх виходило на 16% більше, ніж у 2021-му. Тим часом інді-студії, де працює менше за 50 людей, перейшли до випуску ігор для однієї платформи. З них 77% створюють ігри для настільних пристроїв.

У 2022-му році мобільні ігри грали більше людей, ніж у 2021 році. Щоденна кількість активних користувачів у світі зросла на 8% для середньої гри. Водночас кількість користувачів, які купують контент в грі, впала на 2%. Unity, якому належить ця статистика, припускає, що монетизація через перегляд реклами в іграх може бути більш актуальною, ніж покупки в грі.

Тривалість «життя» ігор зростає з року в рік. Студії працюють на довгострокову перспективу, подовжують актуальність мобільних ігор на 33%. Найбільші показники зростання у казино, карткових ігор та RPG (рис. 2).

**Chart 5.1: Games are around for longer compared to 2022.**

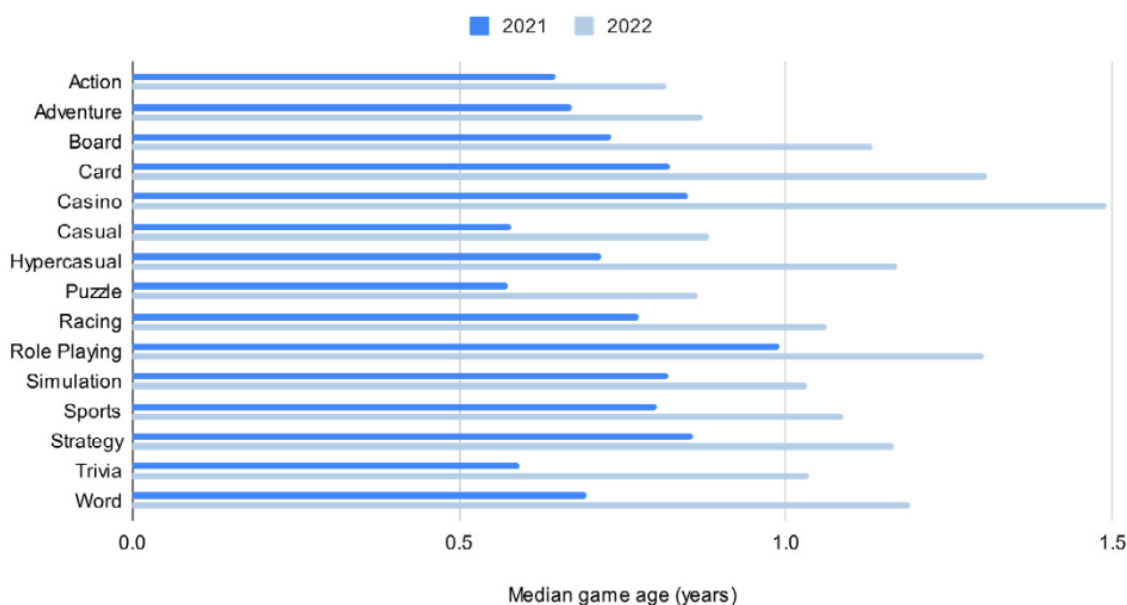


Рисунок 2 — Зростання середньої тривалості життя ігор

Виходячи з даного дослідження було зроблено висновки, що галузь розробки ігрових програм в Україні існує вже досить давно і продовжує розвиватися. Однак, останніми роками швидкість розвитку великих українських компаній впала через війну і нестачу спеціалістів, готових працювати у великих командах, натомість надаючи перевагу інді-розробці.

Також було зроблено висновок, що для успішної кар'єри в сфері розробки ігрових програм потрібні однаково і Hard Skills, і Soft Skills, але акцент потрібно робити в залежності від способу роботи.

Офіційна статистика щодо РІП у світі від Unity показує, що:

- Економічний тиск добре впливатиме на винахідливість розробників;
- Менші студії випускатимуть більше мобільних ігор рівня AAA; гравці звикають до глибоких і захопливих ігор на мобільних пристроях;
- Зросте популярність гібридно-казуальних (hybrid-casual) ігор;
- Більше гравців використовуватимуть ігри як інструменти для створення контенту. І цей контент допомагатиме втримати аудиторію;
- представники 62% студій, які опитали в рамках звіту, заявили, що вони використовують штучний інтелект у своїх робочих процесах; здебільшого мова йде про швидке прототипування та для створення концепцій (рис. 3);
- студії диверсифікують свої підходи до доходів — оскільки середній дохід від покупки в додатку (IAP) на щоденного активного користувача (ARPDАU) знижується, ARPDАU від реклами в додатку (IAA) зростає; студії розширюють свій набір стратегій IAA;
- у період з 2021 по 2023 роки кількість мультиплатформених ігор пропорційно зросла на 40%; у той час, як у 2022 році студії з обмеженими ресурсами здебільшого дотримувались стратегії випуску на одній платформі, за останні два роки вони створили на 71% більше мультиплатформених ігор;
- розробники віддають перевагу багатокористувацьким іграм, попри більшу складність і вартість — у 2023 році мобільні ігри з функціями мультиплеєра мали на 40,2% більше щомісячних активних користувачів

(MAU), ніж ігри для одного гравця, а дохід від багатокористувацьких ігор зріс на \$2,3 млрд доларів США (або ж 10%);

- студії прагнуть продовжити термін служби IP своїх ігор, посиливши прихильність аудиторії — сервіси аналітики, які використовуються для адаптації оновлень до гравців, зросли на 44%, тоді як 27% опитаних розробників повідомили, що додають створений користувачами контент (UGC) у свої ігри.

### 1.5: Developers harness AI to create NPCs.

AI elements for worldbuilding

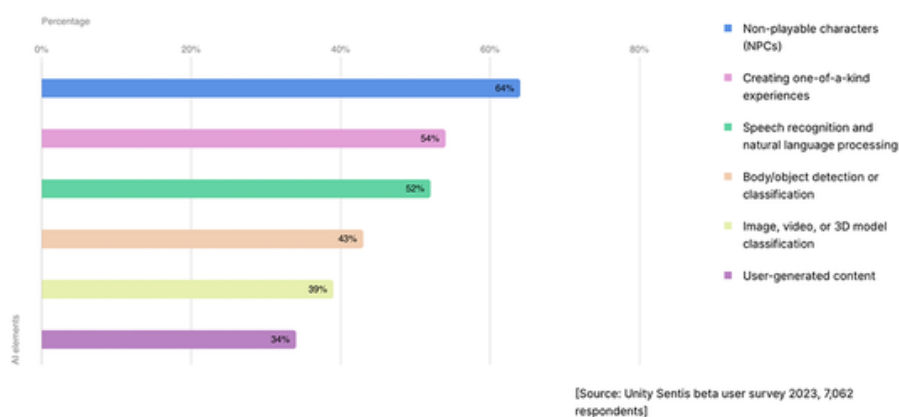


Рисунок 3 — Використання ШІ в розробці NPC

З огляду на зазначені тенденції, можна стверджувати, що галузь розробки ігрових програм в Україні має значний потенціал для подальшого зростання, проте для цього потрібно вирішити питання дефіциту кваліфікованих кадрів, посилити підтримку інди-розробників та впроваджувати новітні технології у процес розробки, зокрема штучний інтелект.

## **ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ**

1. Аналітика Unity про властивості й тренди ігрової індустрії. 2023 Unity Gaming Report. URL: <https://gamedev.dou.ua/news/unity-gaming-report-2023-trend-and-changes/> (Дата звернення: 13.05.2024)
2. Статистика про ІТ в Україні за 2023 рік. Interfax Україна. URL: <https://interfax.com.ua/news/telecom/973724.html> (Дата звернення: 13.05.2024)
3. Відеоігри в Україні. Вікіпедія. URL: <http://surl.li/astqw> (Дата звернення: 13.05.2024)

*Підодвірний Костянтин, здобувач фахової  
передвищої освіти II курсу  
спеціальності «Комп'ютерні науки»  
науковий керівник Гавришків Н. Г.*

## **СПЕЦИФІКА ВИКОРИСТАННЯ TINKERCAD ДЛЯ РОЗРОБКИ ПРОЄКТІВ**

Сучасний розвиток технологій і методів проєктування відіграє ключову роль у різних галузях науки та техніки. Одним із потужних інструментів для моделювання і розробки є TinkerCAD — онлайн-платформа для створення тривимірних моделей і електронних схем. Завдяки своїй доступності та простоті у використанні, TinkerCAD став популярним серед студентів, викладачів і професіоналів, які прагнуть швидко втілити свої ідеї у реальність.

Метою даного дослідження є аналіз специфіки використання TinkerCAD для розробки проєктів, зокрема в освітньому середовищі. У статті розглядаються основні можливості та переваги платформи, її застосування у навчальних процесах, а також приклади успішних проєктів, виконаних за допомогою TinkerCAD. Особливу увагу приділено вивченню того, як TinkerCAD сприяє розвитку практичних навичок у студентів, покращуючи їх розуміння електроніки, програмування та тривимірного моделювання [1].

TinkerCAD дозволяє здобувачам працювати над проєктами разом, обмінюватись ідеями та співпрацювати у режимі реального часу, незалежно від місцезнаходження. Такий вид роботи сприяє розвитку навичок командної роботи та управління проєктами.

Програмування в TinkerCAD виконується в середовищі, адаптованому під Scratch. Для більш складних проєктів можна використовувати та C-подібну мову.



Розглянемо особливості використання онлайн платформи TinkerCAD на прикладі реалізованого проєкту, який демонструє принцип випадковостей зображений на рисунку 1.

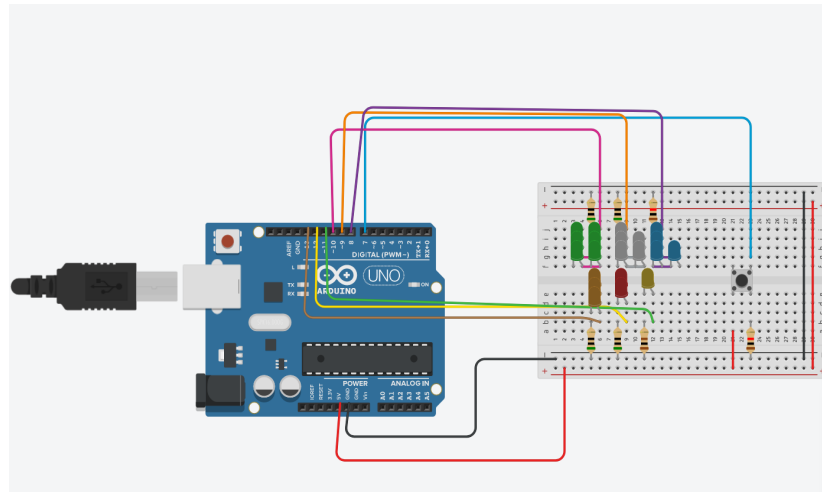


Рисунок 1 — Приклад проєкту з використанням випадковостей

При натисканні на кнопку, підключену до цифрового піна 7, загоряється від 1 до 6 світлодіодів, підключених до пінів 8-13.

Щоб проєкт працював потрібно написати код для Arduino або використовувати код-блок [2]. В даному проєкті використовувалося код для Arduino (рисунок 2):

```
1  int var;
2
3  void setup() {
4      pinMode(13, OUTPUT);
5      pinMode(12, OUTPUT);
6      pinMode(11, OUTPUT);
7      pinMode(10, OUTPUT);
8      pinMode(9, OUTPUT);
9      pinMode(8, OUTPUT);
10     pinMode(7, OUTPUT);
11 }
12
13 void loop() {
14     if(digitalRead(7) == 1){
15         var = random(1,6+1);
16         if(var == 1){
17             digitalWrite(13, LOW);
18             digitalWrite(12, LOW);
19             digitalWrite(11, HIGH);
20             digitalWrite(10, LOW);
21             digitalWrite(9, LOW);
22             digitalWrite(8, LOW);
23         }
24     }
25     else if(var == 2){
26         digitalWrite(13, LOW);
27         digitalWrite(12, HIGH);
28         digitalWrite(11, LOW);
29         digitalWrite(10, LOW);
30         digitalWrite(9, LOW);
31         digitalWrite(8, LOW);
32     }
33     else if(var == 3){
34         digitalWrite(13, HIGH);
35         digitalWrite(12, LOW);
36         digitalWrite(11, LOW);
37         digitalWrite(10, LOW);
38         digitalWrite(9, LOW);
39         digitalWrite(8, LOW);
40     }
41     else if(var == 4){
42         digitalWrite(13, LOW);
43         digitalWrite(12, LOW);
44         digitalWrite(11, LOW);
45         digitalWrite(10, LOW);
46         digitalWrite(9, LOW);
47         digitalWrite(8, HIGH);
48     }
49     else if(var == 5){
50         digitalWrite(13, LOW);
51         digitalWrite(12, LOW);
52         digitalWrite(11, LOW);
53         digitalWrite(10, LOW);
54         digitalWrite(9, HIGH);
55         digitalWrite(8, LOW);
56     }
57     else if(var == 6){
58         digitalWrite(13, LOW);
59         digitalWrite(12, LOW);
60         digitalWrite(11, LOW);
61         digitalWrite(10, HIGH);
62         digitalWrite(9, LOW);
63         digitalWrite(8, LOW);
64     }
65 }
66 }
67 }
```

Рисунок 2 — Код для роботи проєкту

Цей проєкт є гарним прикладом того, як можна використовувати випадкові події для навчання здобувачів освіти роботі з апаратним забезпеченням та програмуванням мікроконтролерів. Він також підкреслює важливість практичних навичок у розробці систем безпеки, які можуть бути використані для захисту життя та майна.

Завдяки цьому проєкту здобувачі освіти отримують практичні знання та навички, необхідні для розробки та впровадження ефективних технічних рішень у галузі безпеки.

Використання Arduino та макетної плати для побудови такого проєкту дає студентам можливість зрозуміти основи електроніки та програмування, що є важливими для будь-якого інженера чи технічного спеціаліста в сучасному світі.

Дане дослідження висвітлює потенціал використання TinkerCAD у професійній діяльності для швидкого прототипування та тестування ідей. Завдяки своїй інтеграції з іншими інструментами й платформами, TinkerCAD дозволяє створювати комплексні проєкти, що включають механічні, електронні та програмні компоненти.

Використання TinkerCAD на заняттях є ефективним способом розвитку навичок проєктування у студентів. Це інтерактивний та доступний інструмент, який сприяє творчості та інноваціям, дозволяє отримати практичний досвід та підготуватися до реальних інженерних задач.

Однак, важливо враховувати обмеження цієї платформи та забезпечити доступ до необхідних ресурсів для максимального використання її потенціалу. Таким чином, TinkerCAD є цінним доповненням до навчального процесу, що допомагає нам студентам засвоїти основи робототехніки та підготуватися до майбутньої професійної кар'єри.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Tinkercad Features [Електронний ресурс] / Офіційний сайт tinkercad.com. URL:<https://www.tinkercad.com/about/features> (дата звернення: 10.05.2024).
2. Методичні рекомендації «Використовуємо Arduino симулятор замість самої плати». URL:<https://naurok.com.ua/metodichni-rekomendaci-vikoristovujemo-arduinosisimulyator-zamist-samo-plati-78684.html> (дата звернення: 10.05.2024).

*Поточняк Роман, здобувач фахової передвищої освіти I курсу  
спеціальності «Комп'ютерна інженерія»*

*науковий керівник Спільніченко О.І.*

*Касперська Л. А.*

## **РОЗРОБКА ТА РЕАЛІЗАЦІЯ ВЕБСАЙТУ ДЛЯ НАВЧАННЯ ХІМІЇ: ІНТЕРАКТИВНИЙ ПІДХІД ДО ОСВІТИ**

У нашому цифровому світі, де технології стають все більш доступними та інтегрованими в освітній процес, необхідно розглядати нові підходи до навчання. Особливо важливо це у наукових дисциплінах, таких як хімія, де велике значення має візуалізація складних концепцій та практичні вправи. Таким чином, розробка вебсайту з навчальною спрямованістю на хімію стає важливим завданням, що кроком в створенні сайту була розробка логотипа. Я знайшов потрібний шаблон Photoshop та розробив логотип з назвою «Хімія: Галицький Фаховий Коледж».[1] Наступним кроком була розробка самого сайту.

Я створив проєкт Django, додав деякі функції представлення, з'єднав їх з URL-адресами, потім зверстав HTML-сторінку згідно PSD шаблону та додав потрібні стилі CSS. На головній сторінці я розмістив великий блок, який генерує випадковий факт, що стосується хімії, з його загальною назвою та сам факт, також буде випадково відтворюватись фон цього блоку при перезавантаженні сторінки. Саме тоді я збагнув, як ці дрібниці можуть допомогти користувачам легко дізнатись багато нового та цікавого про хімію. Тоді я зробив верхню навігацію та футер сайту. Навігація складається з таких пунктів: «Головна», «Підручники», «Кращі проєкти», «Медіатека» та «Профіль».

Після загальної частини розробки я приступив до представлень головних сторінок. Розпочав я з вкладки «Профіль», зображений іконкою

користувача, для того, щоб проєкт Django мав змогу взаємодіяти з користувачем.

В розділі «Підручники» я створив модель підручника з полями: автор, клас та PDF-файл. Завантажив декілька книг зі стороннього вебсайту.[3] Потім я продовжив з будови сторінки, створив flexbox з карток підручників, де розміщувалися фото, автор, клас та кнопка, натискаючи на яку, користувач переходить на сторінку детального перегляду підручника, де його можна прочитати онлайн та завантажити. Також я розмістив фільтрацію книг з пунктами: «Усі підручники», «7 клас», «8 клас», «9 клас», «10 клас», «11 клас» та «Інші». Цей розділ буде корисним для різних категорій здобувачів освіти, яким більше подобається цифрове навчання.

Для розділу «Кращі проєкти» я створив модель проєкту, представлення та з'єднав URL-адреси. У цьому розділі можна побачити назви та авторів проєктів. Ще я додав систему вподобайок, коментарів та переглядів, щоб виділити кращі серед них. Натиснувши на рядок з проєктом, перейдемо на сторінку детального перегляду. Прогорнувши вниз, можна побачити кількість вподобайок та переглядів. Після перегляду проєкту користувач може залишити свій відгук у формі додавання коментаря. Після його надсилання користувач може його видалити, натиснувши на іконку кошика.

У розділі «Медіатека» я розмістив 4 блоки з описом та посиланням на такі пункти: «Відеоуроки», «Відеоексперименти», «Підготовка до НМТ» та «Симулятори». Тут користувач може побачити відео, зображення та інший тип файлу з можливістю завантажити його. Також знизу я розмістив навігацію між попередньою та наступною сторінками. Наші відео, зображення та інші файли можуть не відповідати пунктам розділу «Медіатека», тому я додав ще один пункт під назвою «Інше». Цей пункт я поділив на три підпункти: «Фото», «Відео», «Корисні посилання».

Перед заповненням найбільшого та найважливішого розділу «Медіатека», науковий керівник підібрав цікаві відеоуроки, відеоексперименти нашого коледжу. Також заповнили сторінку для

підготовки до НМТ незамінними конспектами, тут навіть є можливість додати відеоурок, запропонувати завдання у вигляді тексту, а найголовніше – посилання на онлайн тест, де студент зможе попрактикуватись та перевірити свій результат. Цей пункт буде корисним студентам, які збираються здавати НМТ з предмета хімії, та допоможе краще підготуватись до тесту, з цікавим процесом навчання та можливістю отримати найвищий бал. Далі, при заповненні пункту «Симулятори», ми знайшли цікаві інтерактивні хімічні процеси на платформі PhET та додали їх на сайт у вигляді посилання.[5] Це буде цікаво багатьом любителям хімії, досліджуючи ті, чи інші хімічні процеси. На цьому розробка сайту завершилася.

Інтернет-платформи стали невід'ємною частиною сучасної хімічної галузі, сприяючи збільшенню доступності і поширенню знань у цій області. Цією роботою я хочу показати користувачам значення хімії у світі та нашому житті. Так як хімія є однією з природничих наук, вона мусить мати місце в освіті людей та в мережі. Сайт може бути корисним для різних категорій: здобувачів освіти та педагогів закладів загальної середньої, професійно-технічної, фахової передвищої освіти.

Хімія, як важлива галузь науки, вимагає від учнів та студентів не лише засвоєння теоретичних знань, але й розуміння та вміння застосовувати їх на практиці. Візуалізація хімічних концепцій та інтерактивні методи навчання можуть значно полегшити процес засвоєння матеріалу та підвищити мотивацію до вивчення. Саме тому ця наукова робота присвячена розробці та реалізації вебсайту для навчання хімії, з фокусом на інтерактивному підході до освіти. Ця робота має на меті сприяти подальшому розвитку онлайн-освіти та забезпечити учням та студентам нові можливості для здобуття знань у галузі хімії.

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. PSD шаблон. URL: <https://psd.in.ua/> (дата звернення: 24.12.2023 р.)
2. Сайт хімії. URL: <https://sites.google.com/view/allhemi/> (дата звернення: 15.01.2024 р.)
3. Шкільні підручники з хімії. URL: <https://pidruchnyk.com.ua/> (дата звернення: 30.01.2024 р.)
4. Heroku. URL: <https://www.heroku.com/> (дата звернення: 12.04.2024 р.)
5. PhET. URL: <https://phet.colorado.edu/> (дата звернення: 28.03.2024 р.)

*Рибчак Максим, здобувач фахової передвищої освіти IV курсу  
спеціальності «Комп'ютерні науки»  
науковий керівник Кульчинська Н.З.*

## **ТЕХНОЛОГІЇ РЕАЛІЗАЦІЇ ІГРОВИХ ДОДАТКІВ**

Під час розробки ігрових додатків є досить великий вибір програмних засобів для реалізації проєкту, кожен з яких має свої особливості використання, свої переваги та недоліки. Одні рушії відзначаються своєю можливістю обробки графіки високого рівня, проте низьким рівнем оптимізації. Інші в свою чергу мають хорошу оптимізацію, проте менший вибір графічних рішень та методів їх реалізації. Також важливу роль вносить їх доступність для користувачів. Деякі з таких рішень безкоштовні, деякі реалізують систему підписок, деяку купівлю програмного продукту.

Ігровий рушій – програмний засіб який надає розробникам ігор інструментальні засоби для створення і відтворення ігрового середовища. Застосунок надає змогу обробки комп'ютерної графіки, фізичних властивостей, звуку, онлайн з'єднання. Також в сучасних ігрових проєктах досить популярним стає використання штучного інтелекту. Це застосовується здебільшого при розробці ботів. Ігрові рушії дають змогу розробникам зосередитися на створенні контенту та геймплею, використовуючи готові інструменти та ресурси, що значно спрощує процес розробки та підвищує продуктивність.

Одним із найпоширеніших ігрових рушіїв є Unity – це інтегроване середовище розробки (IDE), що спеціалізується на створенні ігор та інтерактивних додатків для різних платформ, включаючи ПК, консолі, мобільні пристрої та Інтернет [1]. Unity відома не лише своєю потужністю та ефективністю, але й широким набором функцій та інструментів, які вона пропонує розробникам усіх рівнів кваліфікації.



Unity підтримує ряд мов програмування, але C# є основною мовою для розробки додатків на цій платформі. Ця мова відома своєю ефективністю та потужними можливостями, які дозволяють створювати складну ігрову логіку, керувати різними аспектами гри та взаємодіяти з різними системами.

Однією з найважливіших особливостей Unity є кросплатформенна сумісність. Розробники можуть створювати один проєкт і переносити його на різні платформи без значних змін у коді чи логіці гри. Це значно спрощує розробку і дозволяє швидко запускати ігри для різних користувачів.

Крім того, Unity надає потужні інструменти для роботи з графікою, включаючи текстурування, освітлення, анімацію та спецефекти. Розробники можуть створювати складні візуальні сцени та контент, щоб зробити свої ігри кращими та цікавішими для гравців.

Окрім створення графічного контенту, Unity також надає інструменти для роботи з фізикою, звуком, штучним інтелектом та мережами. Це робить Unity комплексним інструментом для реалізації широкого спектру ідей та концепцій в ігровій індустрії.

Ці можливості зробили Unity одним з найпопулярніших у світі інструментів для розробки ігор, який використовують як початківці, так і досвідчені розробники для досягнення своїх творчих цілей і надання чудового ігрового досвіду гравцям по всьому світу.

#### Переваги Unity

- Кросплатформеність: за допомогою Unity розробники можуть створити гру або застосунок один раз і експортувати його на різні платформи, такі як Windows, macOS, iOS, Android, Xbox і PlayStation.

- Мова програмування C#: використовуючи C#, розробники отримують доступ до потужних об'єктно-орієнтованих функцій, що дає змогу легко створювати складну логіку та взаємодіяти з іншими системами.

- Багатий інструментарій: Unity надає розробникам інструменти для роботи з різними аспектами ігор, включно з графікою, анімацією, фізикою, звуком, штучним інтелектом і мережевою взаємодією.

- Активна спільнота та підтримка: Unity має велику спільноту розробників і безліч можливостей для навчання та підтримки, включно з документацією, форумами, навчальними курсами та онлайн-ресурсами.

- Екосистема магазину активів: Unity Asset Store надає доступ до тисяч активів, включно з моделями, текстурами, анімацією, плагінами та іншими корисними ресурсами для розширення функціональності вашого проєкту.

- Швидкість розробки: завдяки широкому спектру можливостей і готовим інструментам Unity дає змогу значно скоротити час між розробкою та випуском продукту на ринок.

Для розробки гри також необхідно створити чи відредагувати різноманітні графічні моделі, відповідно потрібен певний графічний редактор. Найчастіше для цих цілей використовують Blender – потужний та безкоштовний графічний 3D редактор [2]. Включає в себе засоби для реалізації моделей, анімацій, рендерингу та композитингу. Він простий у використанні та налаштуванні, інтерфейс інтуїтивно зрозумілий і компактний. Редактор також можна охарактеризувати як добре оптимізований, його можна встановити на офісне обладнання.

Переваги :

- безкоштовний та відкритий код;
- широкий функціонал;
- різноманітність методів анімацій та рендириунгу.

Недоліки:

- відсутність підтримки деяких індустріальних стандартів;
- високий вхідний поріг знань.

Незважаючи на ці недоліки, Blender залишається дуже потужним інструментом для 3D моделювання, анімації та візуалізації, особливо для користувачів, які цінують відкритий вихідний код і його багату функціональність.

Для реалізації взаємодії між різними об'єктами в грі потрібно використати певне середовище програмування. Оптимальним вибором у цьому випадку буде Visual Studio – інтегроване середовище розробки від Microsoft, яке використовується для кодування, налагодження, тестування та відлагодження програмного коду [3].

Переваги Visual Studio:

- інтеграція з платформами Microsoft;
- широкий функціонал;
- розширюваність;
- інтеграція з Azure для розробки хмарних додатків та послуг.
- підтримка декількох мов програмування.

Недоліки Visual Studio:

- високі вимоги до ресурсів;
- досить великий розмір пакету інсталяції.
- складність інтерфейсу;
- висока ціна;
- залежність від екосистеми Microsoft.

При розробці ігрових додатків вибір інструментів має вирішальне значення для успішної роботи та продуктивності. Unity став одним з лідерів у розробці ігор завдяки своїй крос-платформенній сумісності, потужним інструментам та підтримці мови програмування C#. Завдяки широкому спектру можливостей, таких як створення графічного контенту, фізика, штучний інтелект та мережеві технології, Unity є ідеальним вибором для розробників, які цінують якість та ефективність. Однак, такі інструменти, як Blender та Visual Studio також мають свої сильні та слабкі сторони, які слід враховувати при виборі правильного інструменту для конкретного проекту та потреб [4].

Остаточний вибір інструменту повинен ґрунтуватися на вимогах проекту, рівні досвіду розробника та необхідних ресурсах. У швидкозмінній індустрії розробки програмного забезпечення важливо вміти адаптуватися до

нових технологій та ефективно розв'язувати проблеми, зберігаючи при цьому високу якість продукту.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Не тільки Unity та Unreal Engine. Добірка ігрових рушіїв для розробників. *dev.ua*. URL: <https://dev.ua/news/game-engines-list-1657124713> (дата звернення: 17.06.2024).

2. Програми для створення 3D графіки: Топ-8. *CGI School*. URL: <https://cgischool.ua/programy-dlia-stvorennia-3d-grafiky/> (дата звернення: 17.06.2024).

3. Download – blender.org. *blender.org*. URL: <https://www.blender.org/download/> (date of access: 17.06.2024).

4. Учасники проєктів Вікімедіа. Microsoft Visual Studio – Вікіпедія. *Вікіпедія*. URL: [https://uk.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](https://uk.wikipedia.org/wiki/Microsoft_Visual_Studio) (дата звернення: 17.06.2024).

*Романовська Анна, здобувач ОКР Фаховий молодший бакалавр*

*IV курсу спеціальності «Комп'ютерні науки»*

*науковий керівник*

*Сиротюк Н.С.*

## **ВПЛИВ ЕКОНОМІЧНИХ ЗАВДАНЬ У ГРІ НА РОЗВИТОК ФІНАНСОВОЇ ГРАМОТНОСТІ ГРАВЦІВ**

Фінансова грамотність – це ключ до успішного управління власними фінансами. Володіння цією навичкою дозволяє приймати обдумані рішення, планувати майбутнє та досягати фінансової стабільності. Вона відкриває двері до фінансової свободи та можливостей, допомагаючи кожному зробити свої фінансові мрії реальністю.

З поширенням цифрових технологій програми фінансової грамотності та ігри виходять на передовий план як перспективні інструменти для підвищення фінансових знань та навичок прийняття рішень, зокрема серед дітей та підлітків.

Це дослідження спрямоване на аналіз впливу цих змін на економічну поведінку гравців, використовуючи аналіз наукових статей і досліджень інших дослідників.

Дослідження проведене Апреа, Шультеїсом та Штолле (2018) [1] спрямоване на вивчення можливостей інтеграції цифрових навчальних ігор у процес навчання фінансовій грамотності. Автори ретельно аналізують роль цифрових ігор у залученні учнів до навчального процесу та сприянні ефективному досвідченому навчанню у цій області. Дослідження розглядає можливості використання відеоігор для створення інтерактивного навчального середовища, яке спонукає до активної участі. Ці ігри можуть відтворювати реальні фінансові ситуації, дозволяючи учням отримати практичний досвід у прийнятті фінансових рішень та управлінні фінансами.

Автори досліджують можливості ігрового навчання для сприяння експериментальному навчанню. Ігрове середовище дозволяє учням випробувати різні стратегії та реакції на фінансові виклики без реальних фінансових ризиків.

Загалом, результати дослідження свідчать про позитивний вплив інтеграції цифрових навчальних ігор у фінансову грамотність, який може мати важливі наслідки для підвищення рівня фінансової грамотності серед учнів.

Звіт Бойла та співавторів (2016) [2] та дослідження Конноллі (2012) [3] зосереджується на систематичному огляді літератури щодо емпіричних даних, що стосуються впливів та результатів комп'ютерних ігор та серйозних ігор на навчання. Обидва дослідження використовують систематичний підхід до аналізу літератури, що передбачає докладне дослідження наукових статей та досліджень, що опубліковані в даній області. Це дозволяє здійснити комплексний аналіз доступних даних та зробити висновки щодо загальних тенденцій. Дослідження охоплюють різноманітні аспекти використання комп'ютерних ігор у навчанні, включаючи різні вікові групи, типи ігор, а також контекст їхнього використання.

Обидва дослідження підтверджують позитивний вплив використання комп'ютерних ігор на навчання та розвиток навичок. Зокрема, спостерігається покращення мотивації учнів, підвищення зацікавленості у предметі та збільшення рівня засвоєння матеріалу. Вони підтверджують, що комп'ютерні ігри можуть бути ефективними інструментами для навчання, особливо коли вони використовуються в контексті інтерактивного навчання та адаптовані до потреб навчального процесу. Такий підхід може сприяти покращенню результатів навчання та мотивації учнів.

Мотиваційні аспекти навчання на основі ігор досліджуються за допомогою теорії самовизначення Деці, Райана та Гуай (2013) [4] та Райана та Деці (2000) [5]. Ці теорії надають висновки щодо того, як автономія,

компетентність та спорідненість впливають на внутрішню мотивацію та залучення осіб до навчальних занять.

Висновки цього дослідження показують, що програми фінансової грамотності та серйозні ігри мають позитивний вплив на економічні навички прийняття рішень гравців. Шляхом інтеграції інструктивних стратегій цифрові ігри пропонують інтерактивні та цікаві навчальні досвіди, які підвищують фінансові знання, навички та уявлення. До того, мотиваційні фактори, такі як автономія, компетентність та інтерес, відіграють важливу роль у підтримці залучення учнів та навчальних результатів.

Отже, це дослідження підкреслює потенціал програм фінансової грамотності та серйозних ігор у сприянні економічної компетентності та фінансового благополуччя осіб, особливо дітей та підлітків. Майбутні напрями дослідження можуть зосередитися на розробці та оцінці інноваційних інтервенцій, які використовують цифрові технології для ефективного сприяння фінансової грамотності.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Інструктивна інтеграція цифрових навчальних ігор у фінансову грамотність. Апреа, К., Шультеїс, Дж., & Штолле, К. (2018). Фінансова грамотність для дітей та молоді (с. 69–88). Пітер Ланг.

2. Оновлення систематичного огляду літератури емпіричних доказів впливів та результатів комп'ютерних ігор та серйозних ігор. Бойл, Е. А., Хейні, Т., Конноллі, Т. М., Грей, Г., Ірп, Дж., Отт, М., ... Перейра, Ж. (2016). Комп'ютери та освіта, 94, 178–192. <https://doi.org/10.1016/j.compedu.2015.11.003>

3. Систематичний огляд літератури емпіричних доказів щодо комп'ютерних ігор. Конноллі, Т. М., Бойл, Е. А., МакАртур, Е., Хейні, Т., & Бойл, Дж. М. (2012). Комп'ютери та освіта, 59(2), 661–686. <https://doi.org/10.1016/j.compedu.2012.03.004>

4. Теорія самовизначення та реалізація людського потенціалу. Деці, Е. Л., Райан, Р. М., & Гуай, Ф. (2013). Нові перспективи на процеси самопроцесів та людський розвиток (с. 109–133). Інформаційне видавництво віку.

5. Внутрішня та зовнішня мотивація: Класичні визначення та нові напрями. Райан, Р. М., & Деці, Е. Л. (2000). Сучасна освітня психологія, 25(1), 54–67. <https://doi.org/10.1006/ceps.1999.1020>



*Рудник Вадим, здобувач фахової передвищої освіти  
IV курсу спеціальності «Комп'ютерна інженерія»,  
науковий керівник Посвятовська Ольга Богданівна*

## **СИСТЕМА СПОВІЩЕННЯ ПРО ПОВІТРЯНУ ТРИВОГУ**

Однією з найбільших загроз для життя цивільного населення під час збройних конфліктів є ракетні та авіаційні удари. Своєчасне сповіщення громадян про повітряну тривогу відіграє надзвичайно важливу роль, адже дозволяє вчасно перейти у безпечне місце та зберегти життя.

В умовах повномасштабної російської агресії проти України питання ефективної системи сповіщення про можливі обстріли чи авіанальоти набуло особливої актуальності [1].

Через численні обстріли міст та населених пунктів України життя мільйонів людей опинилося під загрозою. Наявність чіткої та оперативної системи сповіщення про повітряну тривогу є життєво необхідною для захисту цивільного населення.

Система моніторингу поширення повітряної тривоги зі звуковим сповіщенням яка відповідатиме таким ключовим вимогам:

- своєчасність сповіщення про повітряну тривогу, що дозволить громадянам вчасно сховатися у безпечному місці;
- точність визначення зони, на яку поширюється попередження про тривогу;
- доступність системи для широкого кола користувачів.

Для реалізації такої системи обрано компоненти, що розглянуті далі.

Мікроконтролер ESP32 з Wi-Fi модулем для отримання даних про тривогу. Це сучасний мікроконтролер з вбудованим модулем Wi-Fi, що дозволяє підключатися до бездротових мереж та отримувати дані з Інтернету. У розробленій системі сповіщення ESP32 використовується для з'єднання з

мережею Інтернет та отримання актуальної інформації про повітряну тривогу від авторизованих джерел [2].

OLED-дисплей SSD1306 для відображення часу та дати. SSD1306 - це невеликий OLED-екран, який підключається до мікроконтролера ESP32 через інтерфейс I2C. На ньому система відображає поточний час та дату, що допомагає користувачам орієнтуватися в часі під час повітряної тривоги.

Адресні світлодіоди WS2812 для індикації зони поширення тривоги. WS2812 - це адресні світлодіоди, кольором яких можна керувати індивідуально. В системі сповіщення вони використовуються для візуальної індикації зони, на яку поширюється попередження про повітряну тривогу.

Окрім візуальних компонентів, у розробленій системі також використовується динамік для відтворення звукових сигналів. Він підключається безпосередньо до мікроконтролера ESP32 та керується ним. При отриманні даних про оголошення повітряної тривоги ESP32 подає відповідну команду на динамік, який відтворює звуковий сигнал. Це забезпечує додатковий канал сповіщення, який дозволяє привернути увагу людей навіть якщо вони не можуть візуально спостерігати за картою. Передбачена можливість регулювання гучності динаміка для забезпечення комфортних умов для користувачів.

Після підключення пристрою до джерела живлення, мікроконтролер ESP32 створює власну точку доступу Wi-Fi. Користувачеві необхідно підключитися до цієї точки та ввести дані своєї домашньої або іншої бездротової мережі. Після введення логіна та пароля ESP32 перезавантажується та підключається до вказаної мережі.

В разі успішного підключення до мережі Інтернет, ESP32 встановлює зв'язок з авторизованими джерелами та отримує актуальні дані про повітряну тривогу в регіоні, зокрема інформацію про зони, на які поширюється тривога.

Отримані дані обробляються мікроконтролером. На OLED-дисплеї SSD1306 відображається поточний час та дата. Водночас, адресні світлодіоди WS2812 світяться певним кольором, що вказує на зону поширення тривоги

згідно із заздалегідь визначеним кодуванням кольорів, Використання динаміка в поєднанні з OLED-дисплеєм та світлодіодами створює мультисенсорну систему сповіщення, яка задіює одразу кілька каналів сприйняття інформації людиною (зір та слух). Це підвищує ефективність сповіщення та зменшує ризик пропустити попередження про повітряну тривогу [3].

Система базується на мікроконтролері ESP32 з підтримкою Wi-Fi та можливістю отримувати дані з Інтернету. Це дозволяє оперативно отримувати актуальні дані про повітряну тривогу від авторизованих джерел.

Портативний дизайн системи з живленням від павербанка робить її досить автономною та мобільною. Це дає змогу використовувати її навіть у разі відсутності централізованого електропостачання.

Завдяки невисокій вартості компонентів та відкритому програмному коду, система є доступною для широкого кола користувачів, включно з громадськими організаціями та навчальними закладами.

Отже, в умовах російської агресії проти України питання своєчасного сповіщення населення про повітряну тривогу набуває важливості. Розроблена система с на базі мікроконтролера ESP32 є актуальним, сучасним та доступним рішенням, що може ефективно доповнювати існуючі системи моніторингу.

Поєднання візуальних (світлодіоди, OLED-дисплей) та звукових (динамік) компонентів забезпечує мультисенсорний підхід, що підвищує надійність сповіщення. Використання бездротових технологій дозволяє оперативно отримувати дані про тривогу від перевірених джерел.

Система є автономною, портативною та легко масштабованою, що розширює можливості її застосування. Відносно низька вартість компонентів та відкритий програмний код роблять систему доступною для широкого кола користувачів.

Подальший розвиток поданої реалізації системи може включати удосконалення алгоритмів обробки даних, додавання додаткових функцій та

вдосконалення дизайну пристрою. Впровадження розробленої системи може сприяти покращенню обізнаності громадян про загрозу повітряних ударів та зменшити ризики для життя і здоров'я цивільного населення.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Як діяти під час сигналу “Повітряна тривога!”. Державна служба України з надзвичайних ситуацій. URL: <https://dsns.gov.ua/uk/news/ostanni-novini/iak-diiati-pid-cas-signalu-povitriana-trivoga> (дата звернення: 09.05.2024).
2. Благодійний фонд «Демократичні ініціативи» імені Ілька Кучеріва / Неофіційний веб-ресурс. URL: <https://alerts.com.ua/> (дата звернення: 09.05.2024).
3. Мапа тривог своїми руками на базі esp32! Гайд для новачків. URL: <https://drukarnia.com.ua/articles/mapa-trivog-svoyimi-rukami-na-bazi-esp32-gaid-dlya-novachkiv-NW25t> (дата звернення: 09.05.2024).

*Солтис Н.М, Стиран В.А. здобувачі фахової  
передвищої освіти II курсу  
спеціальності «Комп'ютерна інженерія»,  
науковий керівник Слепцова О.Я.*

## **ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ В СУЧАСНОМУ СВІТІ**

Штучний інтелект (ШІ) – це захоплююча галузь, що охоплює широкий спектр технологій, спрямованих на створення систем, які можуть сприймати, навчатися, приймати рішення та вирішувати завдання подібно до людського інтелекту. Цей напрям постійно розвивається, відкриваючи нові можливості у різних сферах життя. Штучний інтелект – це галузь комп'ютерних наук, яка займається створенням програм та систем, здатних виконувати завдання, які традиційно вимагають людського інтелекту, такі як розпізнавання мови, навчання, планування та розуміння.

Машинне навчання – це підгалузь штучного інтелекту, яка зосереджена на розробці алгоритмів, здатних навчатися з досвіду. Системи машинного навчання використовують великі масиви даних для виявлення закономірностей та вдосконалення своєї продуктивності без явного програмування.

Нейронні мережі – це алгоритми, натхненні структурою та функціонуванням людського мозку. Вони складаються з великої кількості з'єднаних вузлів, або “нейронів”, які можуть обробляти дані паралельно та виконувати складні завдання, такі як розпізнавання образів та обробка мови [1].

Комп'ютерний зір – це галузь штучного інтелекту, яка дозволяє комп'ютерам “бачити” та розуміти вміст зображень та відео. Системи

комп'ютерного зору аналізують візуальну інформацію, розпізнаючи об'єкти, обличчя, сцени та інші елементи у візуальних даних.

Штучний інтелект знаходить застосування в багатьох галузях, зокрема в охороні здоров'я, де системи ШІ аналізують медичні зображення для виявлення патологій, допомагають у діагностиці захворювань на ранніх стадіях та аналізують великі обсяги медичних даних для виявлення трендів. У фінансовому секторі, ШІ-технології автоматизують торговельні стратегії, аналізують фінансові новини та ринкові дані для прогнозування цінних рухів та допомагають у виявленні шахрайства. У транспортному секторі, безпілотні автомобілі використовують ШІ для навігації та безпечного водіння, системи ШІ оптимізують логістику та прогнозують дорожній рух, що покращує ефективність громадського транспорту. Інтелектуальні помічники, такі як Siri або Alexa, використовують складні алгоритми обробки природної мови для розуміння запитань користувача та надання персоналізованих рекомендацій, і можуть інтегруватися з розумною домашньою технологією для управління календарем користувача та надання актуальної інформації [2].

Етичні та соціальні аспекти штучного інтелекту включають важливі питання, такі як прозорість, конфіденційність та безпека. Прозорість в алгоритмах ШІ є критичною для забезпечення того, щоб процеси прийняття рішень були зрозумілими та пояснюваними, що допомагає уникнути упередженості та несправедливих рішень. Захист персональних даних стоїть на передовому краї викликів конфіденційності, оскільки системи ШІ можуть обробляти великі обсяги особистої інформації, що ставить під загрозу недоторканність приватного життя. Безпека систем ШІ є ключовою для запобігання непередбаченим наслідкам та забезпечення надійності системи перед можливими загрозами чи зловмисним використанням. Розробники повинні враховувати ці аспекти при створенні ШІ-систем, щоб забезпечити етичне та соціально відповідальне застосування цих потужних технологій.

Майбутнє розвитку штучного інтелекту обіцяє бути захоплюючим із значними проривами та інноваціями. Автономні системи, які можуть

самостійно приймати рішення та виконувати завдання без участі людини, продовжують розвиватися, що може призвести до збільшення ефективності та безпеки в багатьох сферах, включаючи транспорт та виробництво. Співпраця людини та ШІ вже показує обнадійливі результати у вирішенні складних проблем, де комбінація людської креативності та аналітичних можливостей ШІ може досягти найкращих результатів. Технологічні прориви продовжують виникати, пропонуючи нові революційні розробки, які можуть кардинально змінити наше майбутнє, включаючи способи лікування хвороб, управління економікою та захист навколишнього середовища. Важливо пам'ятати про етичну сторону цього розвитку та забезпечити, щоб інновації ШІ служили загальному благу.

На сьогоднішній день не існує ШІ, який би мав повну самосвідомість або “особистість”, як це розуміється в контексті людської свідомості. Більшість існуючих систем ШІ базуються на нейронних мережах, які є моделями обчислень, натхненними структурою та функціями людського мозку. Вони можуть вчитися з досвіду та покращувати свою продуктивність у певних завданнях, але не мають власної свідомості чи емоцій. Це не “просто величезний набір операторів if-else”, а складна система з можливостями до певного ступеня імітувати аспекти людської когнітивної діяльності. Однак, питання про “самосвідомий” ШІ залишається предметом філософських дебатів та наукових досліджень [3].

Відсутність самосвідомості означає, що нейромережа, така як ChatGPT не має власних думок, бажань чи емоцій, і тому вона може об'єктивно та безпристрасно виконувати завдання, які їй доручаються. Це може бути перевагою у ситуаціях, де потрібна точність та надійність без впливу людських емоцій [4].

Однак, питання про необхідність самосвідомого ШІ є складним і має багато аспектів. З одного боку, самосвідомий ШІ може мати переваги у певних сферах, наприклад, у виробництві або креативних індустріях, де потрібне інноваційне мислення та здатність до самовдосконалення. З іншого

боку, існують етичні та філософські питання про права та статус таких ШІ, а також потенційні ризики пов'язані з їх непередбачуваною поведінкою.

Нами було проведено дослідження серед студентів Галицького фахового коледжу імені В.Чорновола про використання ними штучного інтелекту у їх студентському житті. В опитуванні взяли участь 141 студент різних спеціальностей. На рис. 1 та рис. 2 наведено результати даного дослідження.

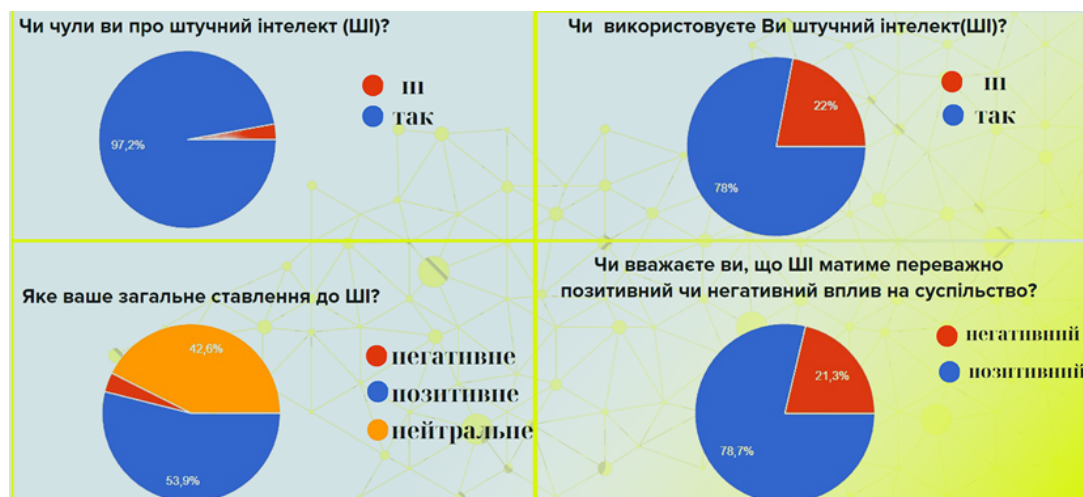


Рисунок 1 — Анкетування з використання ШІ серед студентів

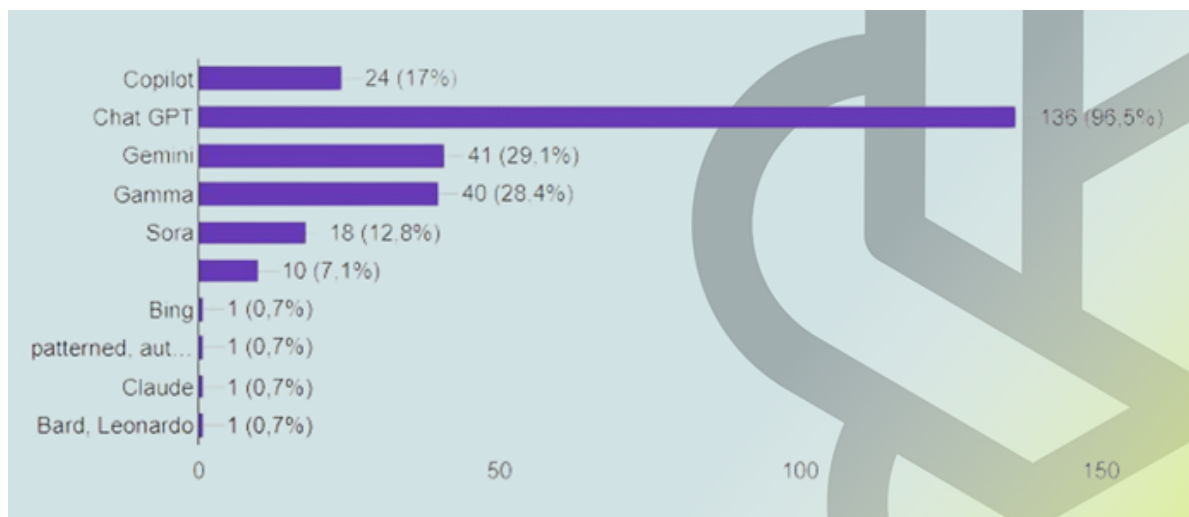


Рисунок 2 — Найбільш популярні мережі серед студентів Галицького фахового коледжу ім.В. Чорновола

Як і очікувалося студентам добре знайомі з ШІ, а сервіс GPT Chat є одним із самих популярних серед студентства.



У питанні про необхідність самосвідомого ШІ також важливо розглядати це з точки зору розвитку технологій та їх впливу на суспільство. Чи будемо ми готовими до створення технологій, яким можна буде приписати “особистостей”, і як це змінить наше розуміння інтелекту та свідомості? Це питання залишається важливим для подальших дискусій та досліджень у галузях штучного інтелекту.

Штучний інтелект це безумовно один з найбільш захоплюючих та перспективних напрямків сучасних технологій. Дослідження показало, як штучний інтелект полегшує наше життя у повсякденних рутинних задачах чи складних рішеннях. Відкривши нові можливості, які раніше здавалися надто складними, завдяки штучному інтелекту ці завдання стають простішими та не займають багато часу. Але з потужними можливостями приходить і відповідальність. Наше завдання полягає в тому, щоб розвивати й використовувати штучний інтелект для підвищення загального добробуту та просування етичних цінностей і справедливості. Важливо, щоб технології служили людям, а не навпаки.

#### ПЕРЕЛІК ДЖЕРЕД ПОСИЛАННЯ

1. Нейромережі та штучний інтелект: майбутнє технологій: веб-сайт. URL: <https://cpashka.biz/blog/nejromerezhi-ta-shtuchnyj-intelekt-majbutnie-tekhnologij> (дата звернення: 15.06.2024).
2. “Я, нейромережа”. Як працює штучний інтелект і до чого все йде: веб-сайт. URL: <https://www.bbc.com/ukrainian/articles/cg3x4ddexg9o> (дата звернення: 16.06.2024).
3. Штучний інтелект, нейромережі і машинне навчання: веб-сайт. URL: <https://www.youtube.com> (дата звернення: 29.05.2024).
4. Нейромережа: що це, як працює, найкращі приклади: веб-сайт. URL: <https://www.gen.tech/post/sho-take-neiromerezhi> (дата звернення: 16.06.2024).

*Худоб'як Костянтин, здобувач фахової  
передвищої освіти IV курсу  
спеціальності «Комп'ютерні науки»  
науковий керівник Кульчинська Н.З.*

## **ЗАХИСТ БАЗ ДАНИХ ВІД SQL ІН'ЄКЦІЙ**

Забезпечення безпеки баз даних від атак SQL ін'єкцій є важливою задачею в інформаційних системах. Під час еволюції цифрових технологій інформаційні системи стають уразливими перед цією формою атаки.

Дослідження в цій області спрямоване на виявлення та усунення вразливостей. У цьому контексті важливо розглянути різні методи захисту баз даних від SQL ін'єкцій, зокрема використання параметризованих запитів, валідацію введених даних та моніторинг та логування активності.

Метою дослідження є захист від SQL ін'єкцій при розробці та адмініструванні програмного забезпечення.

SQL-ін'єкція — це метод отримання несанкціонованого доступу (різновид злому) до бази даних, при якому шкідливий код виконується прямо з поля вводу звичайної форми.

Якщо SQL-ін'єкція пройшла успішно, неавторизовані особи зможуть читати, створювати, оновлювати або навіть видаляти записи з таблиць бази даних. Цей метод використовується хакерами, пентестерами, тестувальниками.[1]

Зважаючи на значення захисту від SQL ін'єкцій, підходи до архітектури програмного забезпечення враховують цю проблему на різних рівнях. Відмінною рисою архітектурних вирішень у контексті захисту є їхнє спрямування на запобігання атакам на безпеку даних в базах даних, зокрема від SQL ін'єкцій.

На рівні розробки програмного забезпечення, важливо розглядати такі аспекти як валідація введених даних, використання параметризованих запитів та обмеження привілеїв доступу до бази даних. Ці практики допомагають уникнути вразливостей, пов'язаних із введенням шкідливих SQL запитів через вебінтерфейси або інші канали.

У адмініструванні баз даних, важливо встановлювати належні налаштування безпеки, такі як обмеження прав доступу до бази даних, шифрування даних у базі та в мережі, а також вчасне встановлення патчів і оновлень для запобігання використанню відомих уразливостей.

До того, моніторинг та аудит безпеки даних є необхідними елементами захисту від SQL ін'єкцій. Постійний моніторинг активності в базі даних дозволяє вчасно виявляти та реагувати на потенційні загрози безпеці.

У минулому десятилітті стали дуже популярними застосунки, що базуються на співробітництві між сервером і клієнтом. У цій архітектурі, сервер постачає інформацію або послуги, в той час, як клієнт взаємодіє з цими ресурсами. Зв'язок між ними забезпечує мережа. Для зберігання даних використовується база даних.

У процесі розробки програм можна використовувати різноманітні мови програмування, такі як Java, Python, C#, PHP, C++, і для кожної з них є інструменти для розробки як клієнтських, так і серверних додатків.[4]

Щоб захистити бази даних від атак типу SQL ін'єкцій, можна використовувати параметризовані запити або ORM (Object-Relational Mapping). Наприклад, у Python з використанням бібліотеки SQLAlchemy, параметризований запит може виглядати так:

```
from sqlalchemy import create_engine, text
# Підключення до бази даних
engine = create_engine('mysql://user:password@localhost/dbname')
# Параметризований запит
query = text("SELECT * FROM users WHERE username = :username")
# Виконання запиту з параметрами
result = engine.execute(query, username='user1')
```

Це дозволяє уникнути виконання небезпечних SQL-команд, які можуть бути введені через користувацький ввід даних.

Розглянемо приклад вебсайту, де користувач може шукати інформацію про фільми за їхніми назвами. Код для обробки запиту у Python має такий вигляд:

```
import pymysql
# Параметри підключення до бази даних
connection = pymysql.connect(host='localhost',user='user',password='password',database='movies_db')
# Отримання назви фільму з GET-параметра
movie_name = input("Введіть назву фільму: ")
# Екранування введених даних
escaped_movie_name = pymysql.escape_string(movie_name)
# Формування SQL-запиту з екранованим значенням
sql_query = "SELECT * FROM movies WHERE name LIKE '%" + escaped_movie_name + "%'"
# Виконання запиту до бази даних
with connection.cursor() as cursor:
    cursor.execute(sql_query)
    result = cursor.fetchall()
# Виведення результатів
for row in result:
    print(row)
```

У цьому прикладі, користувач може вводити назву фільму, яку він шукає. Це значення потім екранується за допомогою функції `pymysql.escape\_string()`, щоб уникнути впливу спецсимволів на логіку SQL-запиту. Потім з екранованим значенням формується SQL-запит для пошуку фільмів, які містять введenu назву. В результаті виводиться список фільмів, які відповідають критерію пошуку. Такий підхід допомагає уникнути SQL ін'єкцій та забезпечує безпеку бази даних.[3]

У MySQL існує механізм підготовлених запитів, який дозволяє передавати дані окремо від самого SQL-коду. Цей механізм розділяє виконання запиту на два етапи. Спочатку формується шаблон запиту, який є звичайним SQL-виразом, але без конкретних значень параметрів. Потім,

окремо від цього шаблону, передаються значення, які використовуються у цьому запиті. Такий підхід дозволяє уникнути впливу самого вмісту даних на логіку запиту, забезпечуючи безпеку обробки інформації.

#### Приклади SQL-ін'єкцій

За роки існування цього класу вразливостей від SQLi-атак постраждала безліч організацій. Наведемо деякі гучні випадки:

#### Fortnite, 2019 р.

Fortnite – це онлайн-гра з аудиторією, що налічує понад 350 млн. гравців. У 2019 році була виявлена вразливість для SQL-ін'єкції, яка дозволила зловмисникам отримати доступ до облікових записів користувача. Вразливість згодом закрили.

#### Cisco, 2018

У 2018 році було знайдено вразливість для SQL-ін'єкції у Cisco Prime License Manager. Пролом дозволив атакуючим отримати доступ до командної оболонки систем, на яких був розгорнутий диспетчер ліцензій Cisco. Компанія Cisco згодом закрила цю вразливість.

#### Tesla, 2014

У 2014 році фахівці з кібербезпеки заявили про успішний злам вебсайту Tesla методом SQL-ін'єкції – їм вдалося отримати адміністративні привілеї та вкрасти дані користувачів.[2]

Проаналізовано заходи захисту від SQL ін'єкцій, що є однією з найбільш поширених та серйозних загроз безпеці вебдодатків.

Досліджено різні методи та техніки, такі як параметризовані запити та екранування даних, які дозволяють уникнути вразливостей, пов'язаних з введенням користувача.

Проведено порівняльний аналіз ефективності цих методів та їхню придатність у різних вебдодатках. Загальний висновок полягає в тому, що комбінація різних методів захисту, таких як використання підготовлених запитів та обробка введених даних, є найбільш оптимальним підходом для запобігання SQL ін'єкціям у вебдодатках.[5]

## ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. SQL-ін'єкції - aCode. *aCode: вебсайт*. URL: <https://acode.com.ua/sql-injection/> (дата звернення: 11.05.2024).
2. Що таке SQL-ін'єкція?. *UKEY WAF - Надійний захист веб-сайту: вебсайт*. URL: <https://ukeywaf.com/baza/shho-take-sql-inyekcziya/> (дата звернення: 11.05.2024).
3. Захист від SQL-ін'єкцій - Блог - HostZealot. *HostZealot: вебсайт*. URL: <https://www.hostzealot.com.ua/blog/about-servers/zaxist-vid-sql-injekcii> (дата звернення: 11.05.2024).
4. Учасники проектів Вікімедіа. SQL-ін'єкція – Вікіпедія. *Вікіпедія: вебсайт*. URL: <https://uk.wikipedia.org/wiki/SQL-ін'єкція> (дата звернення: 11.05.2024).
5. Протидія кібернетичним загрозам у вигляді SQL-ін'єкцій. *Medium: вебсайт*. URL: <https://medium.com/@serhii.starchikov/протидія-кібернетичним-загрозам-у-вигляді-sql-ін'єкцій-9bad2164fb7e> (дата звернення: 11.05.2024).

*Черних Діана, Процик Святослав, здобувачі фахової  
передвищої освіти II курсу  
спеціальності «Комп'ютерні науки»  
науковий керівник Гавришків Н. Г.*

## **РОБОТОТЕХНІКА В АГРОНОМІЇ: АВТОМАТИЗАЦІЯ ПРОЦЕСІВ КОНТРОЛЮ ЗА ТЕХНІКОЮ ТА СІЛЬСЬКИМ ГОСПОДАРСТВОМ**

Аграрна наука є важливою галуззю народного господарства, основою функціонування та розвитку суспільства. Сучасне сільське господарство є високоінтенсивним і базується на науково-технічних досягненнях. Однією з головних проблем, з якою стикаються фермери, є брак робочої сили та часу для посіву та збору врожаю, що спонукає до розвитку робототехніки для підвищення продуктивності (рисунок 1).

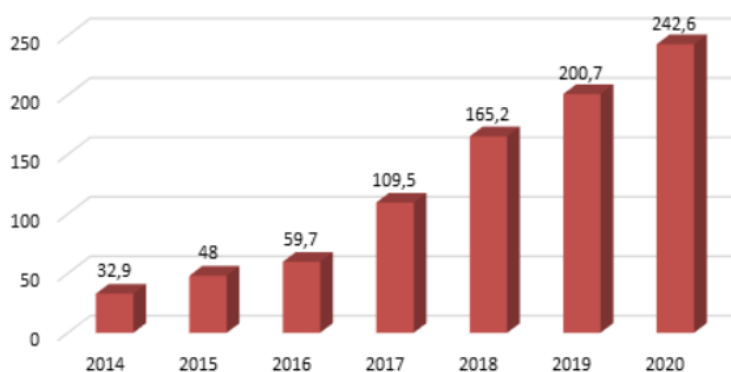


Рисунок 1 — Розвиток робототехніки в період з 2014 р. по 2020 р.

Сільське господарство постійно шукає шляхи підвищення ефективності, зменшення відходів і підвищення врожайності. Роботи стали невід'ємною частиною процесу сільськогосподарського виробництва [1]. Очікується, що до 2050 року чисельність населення світу досягне 9,7 мільярда, тому фермери змушені задовольняти зростаючі потреби в продуктах харчування. В Україні робототехніка стрімко розвивається і

допомагає автоматизувати виробничі процеси для забезпечення безпеки, прибутковості та ефективності.

За допомогою систем, заснованих на штучному інтелекті, сенсорних технологіях, IoT, технології управління даними, розумних алгоритмах, роботах і дронах, традиційні методи землеробства можна замінити більш ефективними.

Це підвищує продуктивність, економічну ефективність і екологічну стійкість, дозволяючи фермерам успішно долати виклики та задовольняти зростаючий попит на продукти харчування [3].

Автоматизація та роботизація стають все більш важливими для сучасного сільськогосподарського бізнесу, підвищуючи ефективність і продуктивність. Однією з інновацій є XAG Autonomous Driving Console System, автоматизована система керування сільськогосподарською технікою, яка дозволяє фермерам точно керувати технікою та оптимізувати робочі процеси (рисунок 2).

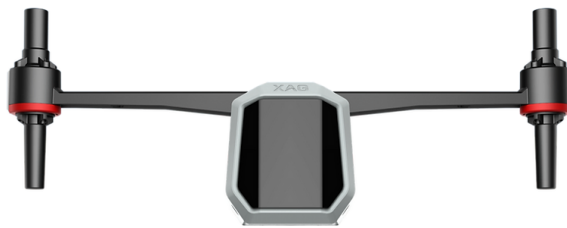


Рисунок 2 — Система XAG AutoPilot Console.

Система забезпечує високу точність обробітку поля, оптимізуючи використання добрив, води та пестицидів, таким чином зменшуючи витрати ресурсів та покращуючи якість роботи. Він автоматично контролює рух техніки, забезпечуючи точне позиціонування до 2 сантиметрів, надаючи детальну інформацію про стан техніки, швидкість і витрату палива. Вбудовані датчики та алгоритми виявляють перешкоди для забезпечення безпечної роботи.



Система скорочує час, витрачений на виконання щоденних завдань, знижує експлуатаційні витрати, зменшує вплив на навколишнє середовище, оптимізує використання палива та зменшує навантаження на ґрунт. Він сумісний з різними пристроями, має гнучкі режими роботи та оснащений засобами безпеки мережі.

Консоль автономного водіння XAG успішно використовується в точному землеробстві, автоматизованих системах захисту рослин і зборі врожаю для підвищення врожайності, зниження витрат і підвищення екологічної безпеки.

Науково-дослідні установи по всьому світу активно вивчають впровадження таких технологій, що допомагає виявити потенційні переваги та недоліки та надати рекомендації щодо їх оптимального використання.

Дослідження показують, що автоматизовані системи керування можуть значно скоротити витрати на добрива та воду. Наприклад, Каліфорнійський університет провів експеримент, у якому використовував автоматизовану систему зрошення на основі даних датчиків вологості ґрунту та погодних умов. Це дає можливість скоротити споживання води на 30% без зниження продуктивності.

Лабораторні випробування автоматизованих систем застосування пестицидів показали, що використання інтелектуальних алгоритмів для визначення того, якими хімікатами необхідно обробити конкретні ділянки сільськогосподарських угідь, може зменшити використання хімікатів на 20-25%. Це не тільки зменшує витрати, але й зменшує негативний вплив на навколишнє середовище. У дослідницьких центрах Європи та США проводять експерименти за допомогою дронів для спостереження за станом рослин. Дані, зібрані дронами, використовуються для точного визначення того, коли і де вносити добрива та інші агротехнічні заходи. Це підвищує врожайність на 10-15% порівняно з традиційними способами.

Випробування автоматизованих систем керування машинами, таких як автономна консоль XAG, показали, що вони можуть значно підвищити

точність землеробства. Наприклад, дослідження, проведене в Австралії, показало, що автономні трактори зменшують перекриття орних смуг, що призводить до економії палива до 10% і зниження витрат на технічне обслуговування [3].

Впровадження таких технологій, як XAG AutoPilot Console, підтверджено численними лабораторними дослідженнями, які підтверджують його ефективність і переваги. Це не тільки покращує продуктивність та економічні переваги сільськогосподарського виробництва, але також допомагає захистити природні ресурси та екологічну стійкість сільськогосподарської практики.

#### ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Основи наукових досліджень в агрономії: Підручник /В.О.Єщенко, П.Г. Копитко, В.П.Опришко, П.В.Костогриз; За ред. В.О.Єщенка. – Дія. – 2018. – 288 с. (дата звернення: 12.05.2024)
2. Роботизовані технології при вирощуванні сільськогосподарської продукції.URL: <http://ela.nati.org.ua:8080/xmlui/bitstream/handle> (дата звернення: 12.05.2024)
3. Сучасний стан та перспективи розвитку робототехніки в Україні URL: <http://oldconf.neasmo.org.ua/node> (дата звернення: 12.05.2024)