



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КРИВОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ
XIII ВСЕУКРАЇНСЬКА НАУКОВО-ПРАКТИЧНА
ВЕБ КОНФЕРЕНЦІЯ АСПІРАНТІВ,
СТУДЕНТІВ ТА МОЛОДИХ ВЧЕНИХ



КОМП'ЮТЕРНІ ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ ТА МЕРЕЖІ

Матеріали конференції
24-26 березня 2020 р.

КСМ-2020

Кривий Ріг

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КРИВОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
КАФЕДРА КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ

ХІІІ ВСЕУКРАЇНСЬКА НАУКОВО-ПРАКТИЧНА
WEB КОНФЕРЕНЦІЯ АСПРАНТІВ,
СТУДЕНТІВ ТА МОЛОДИХ ВЧЕНИХ

КОМП'ЮТЕРНІ ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ ТА МЕРЕЖІ

Матеріали конференції
24-26 березня 2020 р.

Видавничий центр
Криворізький національний університет
Кривий Ріг 2020

УДК 681.3.06
ББК 32.973.202
К60

Відповідальний за випуск д-р техн. наук,
професор Купін А. І.

Друкується згідно з рекомендацією Вченої Ради ФІТ Криворізького національного університету (протокол №8 від 05.03.2020 р.).

Змістова частина друкованого матеріалу збірки викладена згідно з електронними носіями, поданими авторами.

К60 **Комп'ютерні інтелектуальні системи та мережі.** Матеріали XIII Всеукраїнської науково практичної WEB конференції аспірантів, студентів та молодих вчених (24-26 березня 2020 р.). – Кривий Ріг: Криворізький національний університет, 2020. – 264 с.

Містить матеріали науково-практичної WEB конференції аспірантів, студентів та молодих вчених з питань розробки, проектування, діагностики та моделювання комп'ютерних систем та мереж, розробки програмного та апаратного забезпечення; розглядаються проблеми створення та використання систем паралельних і розподілених обчислень, штучного інтелекту, а також питання захисту інформації.

УДК 681.3.06
ББК 32.973.202
Криворізький національний університет, 2020

СЕКЦІЯ 1. DIAGNOSTICS. ДІАГНОСТИКА КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ

Ксенич Є.Г.
Державний університет “Житомирська політехніка”

ОПТИМІЗАЦІЯ КОРИСТУВАЦЬКОГО ДОСВІДУ З ВИКОРИСТАННЯМ СУЧАСНИХ ФРЕЙМВОРКІВ.

Проаналізовано перспективи застосування Фреймворків: IONIC 2, React Native, Swift для інтерактивних додатків. Дано характеристику React Native та зроблено порівняння його швидкодії з іншими технологіями.

Користувацький досвід рід дуже важлива. Оскільки користувацький досвід залежить від швидкості відповіді. Швидкість реагування системи напряму залежить від потужності пристрою, та технологій, які забезпечують обробку. Якщо на потужність пристрою ми не можемо вплинути, то на технологій – можемо. Розглянемо найпопулярніші технології.

У сучасному світі, де користувач має великий програм, що по суті своїй мають схожий функціонал – боротьба за користувача доволі жорстока. І якщо функціонал майже однаковий вирішальним стає – користувацький досвід. А велику частину цього досвіду складає – взаємодія з додатком. Та швидкість у цьому є вирішальною. Є навіть стандарти, які описують скільки часу користувач може чекати відповіді від додатку.

І якщо решта параметрів рівні – то використовувати краще, швидшу бібліотеку, чи технологію – ось що може вивести розроблений нами продукт у лідери.

React Native - це програма для мобільних додатків з відкритим кодом, створена Facebook. Він використовується для розробки додатків для Android, iOS, Web та UWP, дозволяючи розробникам використовувати React разом із можливостями рідної платформи.

Принципи роботи React Native практично ідентичні React, за винятком того, що React Native не маніпулює DOM за допомогою Virtual DOM. Він запускається у фоновому процесі (який інтерпре-

тує JavaScript, написаний розробниками) безпосередньо на кінцевому пристрої та спілкується з нативною платформою через серіалізацію, асинхронний та пакетний міст. [1].

У таблиці 1 показано порівняльну характеристику використання ресурсів пристрою для виконання певних дій. Дії, які виконувалися за допомогою різних бібліотек: відкриття карт, відкриття карт додатку facebook, відкриття основної сторінки тестового додатку, та відкриття генерованого списку. Дані типи задач – є доволі справедливими. Адже вони задовольняють список задач, які, зазвичай, використовує, користувач, або схожих на такі.

Таблиця 1 – Час виконання операцій з використанням різних JavaScript-бібліотек

Ресурс	Карти		Профіль		Сторінка		Список	
	Swift	RN	Swift	RN	Swift	RN	Swift	RN
CPU	34	24	17	16	10	15	27	26
GPU	17	22	25	24	15	14	16	22
Memory	69	33	1.2	1.2	0.4	0.3	2.7	2.3

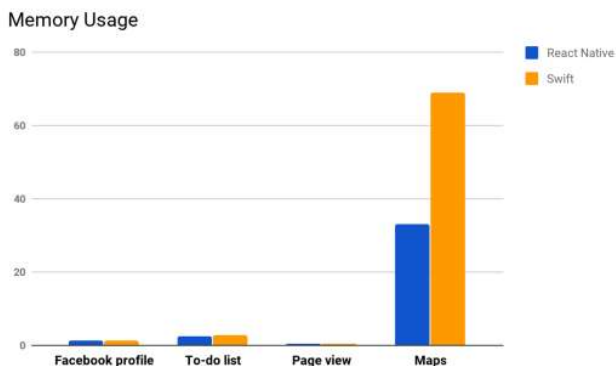


Рис. 1. Діаграма використання пам'яті.

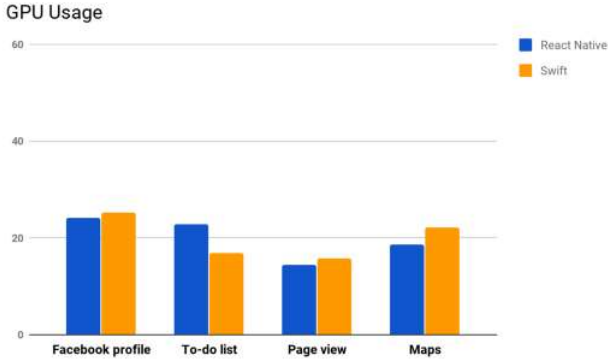


Рис. 2. Діаграма використання відео ядра.

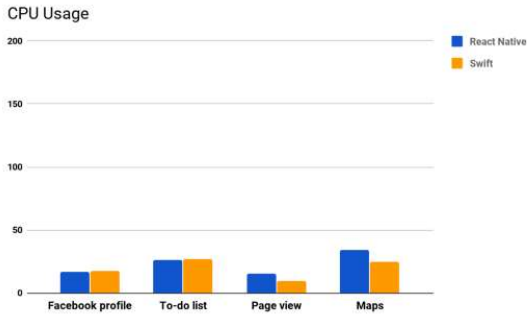


Рис. 3. Діаграма використання головного процесора.

ВИСНОВКИ

Таким чином React Native являє собою технологію, що найкраще підходить для виконання поставленої задачі. Та, що, хоча у деяких параметрах він програє swift, але в цілому працює з меншими затратами основних типів ресурсі. Хоча у деяких параметрах він і програє, але вирішальним має бути діаграма використання пам'яті.

ЛІТЕРАТУРА

1. React Native [Електронний ресурс] – Режим доступу до ресурсу: https://en.wikipedia.org/wiki/React_Native.

*Яковлева І.Д.,
ст. викл., Одеський національний політехнічний університет,
Попов М.В.
магістр, Одеський національний політехнічний університет*

ДОСЛІДЖЕННЯ СИСТЕМИ ВІДДАЛЕНОГО КЕРУВАННЯ МОДУЛЯМИ З ЗВОРОТНИМ ЗВ'ЯЗКОМ ЧЕРЕЗ ІНТЕРНЕТ

В доповіді розглянуті існуючі комп'ютерні системи віддаленого керування модулями. Проаналізовані їх переваги, недоліки, та основні характеристики. Розрахована пропускна здатність мережі для реалізації зворотнього зв'язку через Інтернет.

В умовах сучасного часу освіта у вищих навчальних закладах (ВНЗ) трансформується, приймаючи формат «Digital Education». Це передбачає собою можливість в онлайн прослуховувати лекції, проходити практичні заняття, а також використовувати налагоджувальні модулі поза межами ВНЗ. З метою підвищення якості освіти виникла необхідність у комп'ютерній системі, яка дозволить студенту у реальному часі мати доступ до віддаленому модулю, проводити виміри або програмувати його, а також спостерігати результати процесу для корегування чи запису.

Основними елементами конфігурації відомих комп'ютерних систем віддаленого керування налагоджувальними стендами (модулями) являються: віддалені робочі станції, Інтернет, сервер, системи програмування локальної робочої станції [1].

Недоліком розглянутих систем є відсутність можливості в реальному часі спостерігати результати програмування для додаткових модулів таких як: LCD дисплей, світлодіоди, LED семисегментний індикатор.

Розроблена комп'ютерна система, в якій віддалені робочі станції через Інтернет і сервер підключаються до локальної робочої станції. Відладочні модулі EVAL-8031 та STM-32 підключені до локальної робочої станції через програматор STK500 (EVAL-8031) і через USB порт (STM-32). В якості додаткових модулів використані: LCD дисплей, світлодіоди, LED семисегментний індикатор та матричний індикатор 5x7 пікселів. За необхідністю спостерігати у реальному часі за результатами вимірів з додаткових модулів для

зміни, чи корегування параметрів, які задаються, до складу комп'ютерної системи додано елемент – веб камеру. Зображення з якої транслюється до локальної робочої станції і далі через сервер та Інтернет до віддалених робочих станцій.

Чим більша кількість робочих станцій тим більші вимоги до продуктивності серверів та пропускної здатності мережі, так як зображення з локальної робочої станції передається до користувача через мережу. Тому доцільно провести аналіз необхідної пропускної здатності мережі.

Необхідна пропускна здатність мережі (P_n) визначається за формулою (1) :

$$P_n = P_{врс} + P_{нф} + P_k, \quad (1)$$

де: $P_{врс}$ – необхідна пропускна здатність мережі для трансляції віддаленого робочого столу, $P_{нф}$ – необхідна пропускна здатність мережі для передачі файлів, P_k – необхідна пропускна здатність мережі для передачі зображення з камери.

Так як для підключення до віддаленого робочого столу використовується програма TeamViewer, значення $P_{врс}$ дорівнює необхідній мінімальній пропускній здатності, яка вказана виробником (1 Мбіт/сек).

Так як передача файлів виконується завдяки пакету TeamViewer, тому вказаної необхідної пропускної здатності мережі у 1 Мбіт/сек достатньо як для підключення до віддаленого робочого столу, так і для передачі файлів ($P_{нф}$).

Необхідна пропускна здатність мережі для передачі зображення з камери P_k визначається за формулою (2):

$$P_k = \frac{P_{рск} \cdot FPS \cdot n \cdot 8}{1024}, \quad (2)$$

де: $P_{рск}$ – розмір стислого кадру, залежить від типу кодеку, який використовує веб камера [2] (дивись табл.1), FPS – частота кадрів веб камери в секунду (кадри/сек), n – кількість веб камер.

Таблиця 1 - Розмір стислого кадру

Кодек	Роздільна здатність	Початковий розмір, кБайт	Розмір після стиснення, кБайт	Ступінь стиснення

MJPEG	1280x720	2700	175	15,4
JPEG2000	1280x720	2700	153,6	17,6
MxPEG	1280x960	3599	149	24,1
H.264	1280x960	3599	48	74,9

Аналіз приведених результатів показав що мінімальна необхідна пропускна здатність мережі, як випливає з формули (2), дорівнює 7Мбит/сек.

ВИСНОВКИ

За рахунок включення у комп'ютерну систему віддаленого керування веб камери з'явилась можливість встановити зворотній зв'язок. Це дозволило студенту налагоджувати розроблену програму в реальному часі за рахунок спостереження відео результатів на додаткових зовнішніх модулях індикації.

ЛІТЕРАТУРА

1. SIG Remote Labs and Online Experimentation. [Електронний ресурс]. – Режим доступу: <http://eatel.eu/special-interest-groups/sig-remote-labs-and-online-experimentation>
2. EvoTrade. [Електронний ресурс]. – Режим доступу: <https://evotrade.com.ua/ru/raschet-arhiva-sistemy-videonablyudeniya>

Попов М. Д.

Криворізький національний університет

Вдовиченко І.Н.

к.т.н., доцент, Криворізький національний університет

СИСТЕМА МОНІТОРИНГУ ТА КОНТРОЛЮ СТАНУ ПАРАМЕТРІВ ВІДДАЛЕНОГО ВУЗЛА ЗВ'ЯЗКУ

Розглянуто функції етапи роботи даного проекту. Проаналізовані основні функції контролеру та сенсори які можливо під'єднати.

Метою роботи є розробка системи моніторингу в інформаційній структурі, будь то маленька компанія або величезний дата-центр, потрібен, щоб системні адміністратори були сповіщені про

помилки й проблеми в інфраструктурі раніше, або хоча б одночасно з користувачами. Необхідність прогнозування, а тим самим і запобігання помилок, оповіщення про них і зберігання інформації про стан систем і служб (Рис. 1).

Важливість роботи пояснюється неможливістю стеження за деякими важливими показниками працездатності сервісів і серверів за допомогою стандартних засобів моніторингу, або ж високою трудомісткістю таких методів. Створений моніторинг з мінімальними витратами та невисокою трудомісткістю вирішує поставлені завдання.

“Контролер може бути використаний для виконання наступних функцій:

- 1) Охорона і сигналізація.
- 2) Оповіщення користувачів.
- 3) Індикація станів.
- 4) Захист зовнішнього обладнання від зависань та можливість перезавантаження.
- 5) Ведення журналів та оповіщення про події.
- 6) WEB доступ.
- 7) Резервування і відновлення конфігурації при внутрішніх збоях” [2].

Контролер підключається до мережі передачі даних через вбудовані інтерфейси Fast Ethernet. Віддалене керування через вбудований інтуїтивно-зрозумілий WEB інтерфейс. Як відзначено в [2] контролер дозволяє віддалено контролювати наступні типи датчиків:

- 1) Температура.
- 2) Вологість.
- 3) відкривання дверей.
- 4) Рух в зоні.
- 5) Змінна напруга.
- 6) Постійна напруга.
- 7) Обертання вентиляторів.

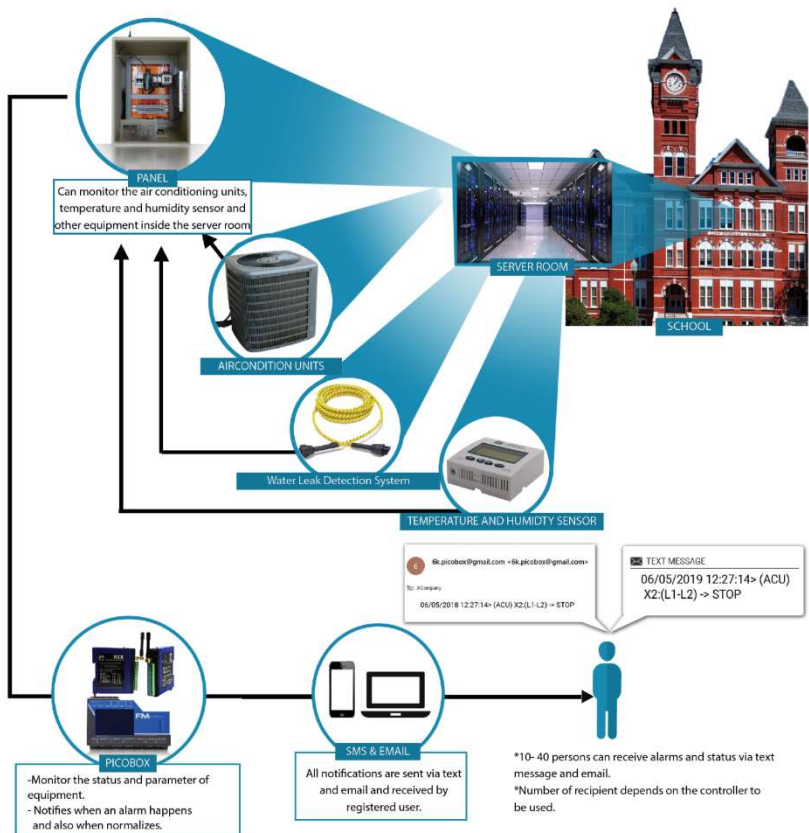


Рис. 1. Принцип роботи

ВИСНОВКИ

Запропонований варіант прискорить та полегшити контроль стану параметрів, також за допомогою WEB інтерфейсу спростить отримання інформації. Щоб забезпечити працездатність ІТ-підприємства, необхідно заздалегідь виявляти вузькі місця в конфігурації систем і мереж, а також швидко дізнаватися про наявність помилок і її причини. Для цих потреб в компаніях, де подібне стеження неможливо за рахунок тільки фахівців, заведено використовувати системи моніторингу.

ЛІТЕРАТУРА

1. Моніторинг сети [Електронний ресурс] – Режим доступу до ресурсу: <https://www.motadata.com/ru/what-is-network-monitoring/>.
2. Функції контролеру [Електронний ресурс] – Режим доступу до ресурсу: <http://coldtrade.spb.ru/proizvodstvo/kontrollery-upravleniya-i-monitoringa/>.

СЕКЦІЯ 2. PARALLEL COMPUTING. ВИСОКОПРОДУКТИВНІ КОМП'ЮТЕРНІ СИСТЕМИ, ПАРАЛЕЛЬНІ ТА РОЗПОДІЛЕНІ ОБЧИСЛЕННЯ

*Мотькін М.А.,
Харківський національний університет радіоелектроніки
Аксак Н.Г., д.т.н., професор,
Харківський національний університет радіоелектроніки*

ПОРІВНЯННЯ ПРОДУКТИВНОСТІ ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ ПРИ ВИКОРИСТАННІ РІЗНИХ ПЛАНУВАЛЬНИКІВ OPENMP

Проаналізована продуктивність обчислень в паралельній програмі, що написана із застосуванням бібліотеки паралельних обчислень OpenMP при використанні різних планувальників циклу.

На поточний момент все частіше і частіше постає питання продуктивності обчислень на персональних комп'ютерах. Найчастіше це питання стосується звичайних розробників програмного забезпечення, у яких немає доступу до потужних суперкомп'ютерів або обчислювальних кластерів. Для вирішення проблеми продуктивності розробники програмного забезпечення можуть вдаватися до ряду архітектурних або програмних рішень, здатних збільшити продуктивність обчислень. Одним з таких рішень є застосування OpenMP.

OpenMP – це бібліотека, яка призначена для створення паралельних програм на мовах C, C++ і Fortran. Ця бібліотека найчастіше застосовується для прискорення роботи циклів, так як саме цикли найкраще підходять для паралелізації програми. Для цього в OpenMP на C ++ застосовуються директиви `#pragma omp parallel` і `#pragma omp for`. Директива `#pragma omp for` дозволяє розбити дані на кілька блоків, які будуть оброблені паралельно, що призведе до значного збільшення продуктивності програми.

Для планування розподілу циклу використовується опція `schedule`. У OpenMP є 5 типів планувальників[1]: `static`, `dynamic`, `guided`, `runtime` і `auto`. Перші 3 планувальники визначають різні стра-

тегії планування розподілу даних. Планувальник runtime використовує стратегії інших планувальників в залежності від значення змінної середовища OMP_SCHEDULE під час запуску програми. Планувальник auto встановлює режим автоматичного вибору планувальника і розміру блоку під час компіляції програми.

У табл. 1 представлені результати тестування продуктивності програми для обчислення результуючого масиву розміром 100 000 000

$$result[i]=(array[i-1]+array[i]+array[i+1])/3.$$

Планувальники циклів OpenMP можуть оптимізувати способи поділу ітерацій циклу. Під час тестування були отримані результати для планувальників static, guided, dynamic і runtime(auto).

Тестування проводилося на комп'ютері на комп'ютері з операційною системою Windows 10 x64, процесором Ryzen 5 2500U (3,6 GHz) та оперативною пам'яттю RAM DDR4 1200 MHz.

Таблиця 1 – Значення часу виконання паралельної області

Планувальник	Розмір блоку				
	16	64	256	1024	5000
static	2,102	1,913	1,828	1,765	1,841
dynamic	1,979	1,792	1,710	1,709	1,709
guided	1,641	1,660	1,665	1,654	1,664
runtime(auto)	1,942	1,942	1,942	1,942	1,942

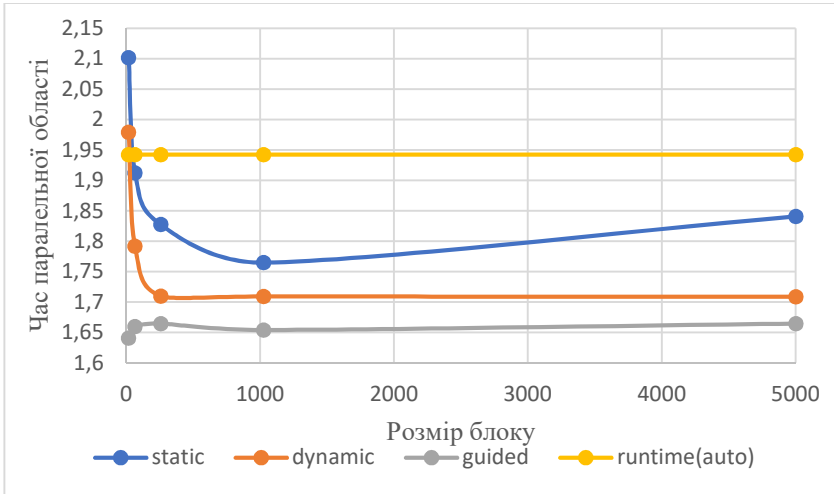


Рис. 1. Графіки залежності часу виконання паралельної області від розміру блоку і планувальника

ВИСНОВКИ

В ході аналізу планувальник `guided` показав найкращі результати. Таким чином, можна зробити висновок, що використання планувальника `guided` є найбільш доцільним для розпаралелювання циклів при вирішенні поставленого завдання.

ЛІТЕРАТУРА

1. The OpenMP API specification for parallel programming [Електронний ресурс] – Режим доступу до ресурсу: <https://www.openmp.org>.

Носов В. С.,

Харківський національний університет радіоелектроніки

Аксак Н. Г.

д.т.н., професор, Харківський національний університет радіоелектроніки

ПОРІВНЯЛЬНИЙ АНАЛІЗ ЗАСОБІВ ПРОФІЛЮВАННЯ ПРОДУКТИВНОСТІ ПРИ РОБОТІ З OPENMP ДОДАТКАМИ

Проаналізовано перспективи застосування засобів профілювання: Intel VTune Profiler, AMD uProf, засоби профілювання Visual Studio, TAU для OpenMP додатків. Зроблено порівняння можливостей Intel VTune Profiler з іншими засобами профілювання.

OpenMP — досить популярна модель паралельного програмування, особливо для високопродуктивних обчислень. Але щоб цієї високої продуктивності досягти, OpenMP конструкції частенько доводиться "налаштовувати" [1]. Для чого потрібно використання профілювальника. Більшість засобів профілювання видають дані про продуктивність, що асоційовані з функціями або циклами, але не дають повної картини щодо конкретних регіонів OpenMP. В результаті програміст втрачає контекст. А без OpenMP-контексту діагностика дисбалансів або накладних витрат стає досить складним завданням.

Гарним прикладом засобів профілювання, які не взаємодіють з OpenMP, можуть виступити AMD uProf та інструментарій Visual Studio. Засоби профілювання Visual Studio можна використовувати для аналізу проблем з продуктивністю в додатку. Профілювальник від Visual Studio (використання ЦП) збирає відомості про функції, які виконуються в додатку. Крім того, цей засіб перераховує функції, які виконували максимальний обсяг роботи, а також надає графік часової шкали, що дозволяє зосередити увагу на певних сегментах сеансу вибірки. Це допоможе переходити до функцій, де спостерігаються проблеми з продуктивністю. Можна також використати розширення Concurrency Visualizer, яке надасть графічну, табличну та текстову інформацію про активність потоків, але цієї інформації недостатньо для аналізу регіонів OpenMP. Схожий результат було

отримано при використанні AMD uProf. При використанні цього інструменту аналізу продуктивності була отримана інформація про конкретні функції — їх назву та час їх роботи, були отримані графіки навантаження потоків процесора, але ці графіки були не деталізовані та з дуже великим масштабом.

Для отримання більш інформативного результату, інструмент профілювання повинен підтримувати технологію OpenMP на рівні початкового коду. Серед таких інструментів можна виділити Intel VTune Profiler [2]. Даний засіб доступний як окремий продукт, так і в складі пакета Intel Parallel Studio. Intel VTune Profiler дозволяє аналізувати послідовні та паралельні програми. За допомогою цього засобу було отримано інформацію про профілі окремих паралельних регіонів OpenMP (час CPU, гарячі функції і т. д.), баланс роботи всередині окремого паралельного регіону OpenMP, баланс паралельного/послідовного коду, рівень гранулярності паралельних завдань, інформацію про використання потоків процесора, а також про об'єкти синхронізації, часу очікування і передачі управління між потоками. Отримані графіки дозволили дуже чітко побачити паралельні регіони OpenMP і відповідну їм активність потоків.

TAU — це інструмент оцінки продуктивності, який підтримує профілювання і трасування для програм, написаних на мовах C, C++, Fortran, Java, Python і Spark [3]. Для профілювання програм OpenMP TAU включає в себе інструментарій вихідного рівня (Opari), API runtime "collector" (званий ORA), вбудований компілятор OpenMP (OpenUH), і бібліотеку часу виконання OpenMP, підтримуючу OMPT стандарт OpenMP 5.0. TAU підтримує як прямі виміри, так і засновані на подіях режими вибірки для профілювання. Для трасування TAU надає візуалізатор трасування з відкритим вихідним кодом (Jumpshot) і може генерувати власні файли трасування OTF2, які можуть бути візуалізовані в візуалізатор трасування Vampir. Tau Commander спрощує робочий процес і установку TAU. TAU підтримує набори інструментів PAPI і LIKWID для доступу до даних лічильника продуктивності низькорівневого процесора, специфічного для апаратного забезпечення, щоб співвіднести його з областями коду OpenMP. TAU поставляється з ліцензією на стиль BSD.

ВИСНОВКИ

В результаті порівняння різних програм для профілювання додатків було виявлено, що найкраще всього серед зазначених профілювальників із завданням аналізу продуктивності програми з OpenMP впорався Intel VTune Profiler. Intel VTune Profiler надає можливість поглянути глибоко всередину OpenMP додатка. За допомогою нього можна оцінити баланс послідовного і паралельного коду і як програма поводить себе в кожному паралельному регіоні. Intel Vtune Profiler може допомогти знайти проблеми з балансом навантаження між потоками OpenMP, проблеми з гранулярністю, оцінити накладні витрати і зрозуміти схему синхронізації. Прив'язка детальної статистики використання процесора до конкретного регіону OpenMP дозволяє краще зрозуміти поведінку програми. Мінусом даного профілювальника є те, що програма повинна бути обов'язково скомпільована за допомогою компілятора від Intel, інакше в результаті аналізу продуктивності не буде зібрана потрібна інформація про роботу OpenMP, а також частина функціоналу програми обмежена використанням процесорів від Intel.

ЛІТЕРАТУРА

1. The OpenMP API specification for parallel programming [Електронний ресурс] – Режим доступу до ресурсу: <https://www.openmp.org/>.
2. Intel® VTune™ Profiler [Електронний ресурс] – Режим доступу до ресурсу: <https://software.intel.com/en-us/vtune>.
3. Tuning and Analysis Utilities [Електронний ресурс] – Режим доступу до ресурсу: <https://www.cs.uoregon.edu/research/tau/home.php>.

*Штанько О. Ю.,
Криворізький економічний інститут
Київського національного економічного
Університету імені Вадима Гетьмана
Вдовиченко І. Н.,
к.т.н., доцент, Криворізький національний університет*

ВИКОРИСТАННЯ ДНК ЯК ЗАСІБ ЗБЕРЕЖЕННЯ ІНФОРМАЦІЇ

Проаналізовано використання ДНК для зберігання інформації

Від початку розквіту електронних технологій постають питання в збереженні інформації, такі як:

- Зменшення розміру пристроїв, котрі її зберігають;
- Збільшення швидкості обробки інформації, а саме час запису та зчитування;
- Збільшення ресурсу роботи пристроїв до виходу їх зі строю.

Зменшення розмірів зі збільшенням ємності завжди було під увагою великих та малих компаній, адже це зменшує витрати, котрі необхідно кожного року вкладати в нове обладнання та нове зведення дата-центрів. Однією з компаній котра досягла у цьому успіху є Lexar. Вона у 2019-ому році представила SD-карту ємністю в 1 ТБ [1]. Розміри ж цієї карти: довжина – 32мм, ширина – 24мм та висота 2.1мм [2]. Розміри дійсно невеликі, якщо порівнювати з картами котрі випускалися кілька десятків років тому, та навіть, якщо порівняти з SD-картою річної давності, прогрес мінімум у два рази, що суттєво збільшує можливість використання даного носія інформації.

Проте вчені незважаючи на такі успіхи в зберіганні інформації на твердих носіях, мають намір перейти до використання ДНК. Використання ДНК обумовлюється її стабільністю, доволі міцністю та неважким копіюванням. Поштовхом до такої розробки став підрахунок використання пам'яті людством, так у 2013 році було виготовлено 4,4 зетабайта різної інформації, а до 2025р. ця цифра повинна вирости до 160 зетабайта, проте у 2040-ому людство буде вже не в силі впоратися зі збереженням всієї інформації [3].

Компанія Microsoft першою внесла великий вклад у розвиток використання ДНК. Вона розробила прототип котрий перетворює будь-які цифрові данні в послідовність ДНК для зберігання, а потім може зчитати збережену інформацію.

Першим записаним та зчитаним об'єктом стало слово «hello». Проте за кілька років учені записали на рівні ДНК такі твори світової літератури та музики, як Smoke on the Water гурту Deep Purple, «Війну і мир» Лева Товстого та навіть гіфку з конем.

Проте незважаючи на всі перспективи, на зараз технологія лише в розробці. Тому швидкість обробки інформації ще надто мала та дорожня технологія теж не сприяє використанню на сьогоднішній час. Але розвиток дійсно простежується, хоча б у ціні. Перші проекти обходилися приблизно у \$100 млн, а на зараз вже \$1000.

На графіку рис. 1. спадання вартості обробки, на малюнку секвенування, можна помітити швидке падіння, тому у майбутньому коли технологія буде використовуватися у всьому світі її вартість не буде на багато вищою ніж використання сьогоднішніх пристроїв збереження інформації [4].

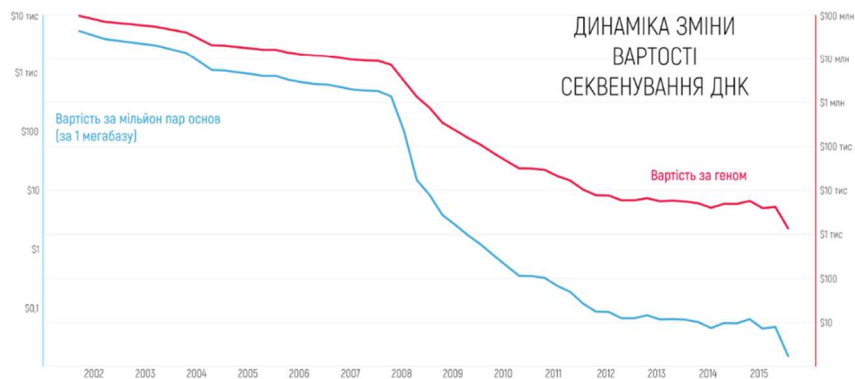


Рис. 1 – Спадання ціни з розвитком технології

ВИСНОВКИ

Використання ДНК для зберігання інформації дуже перспективна річ. Вона в свою чергу вирішить майбутню проблему з нестачею пам'яті для зберігання інформації. Проте для успішного

впровадження технології у повсякденне використання, вона повинна удосконалитися, а саме збільшити швидкість та зменшити ціну на обробку інформації.

ЛІТЕРАТУРА

1. SD-карта на 1 ТБ [Електронний ресурс] – Режим доступу до ресурсу: <https://www.iphones.ru/iNotes/sd-karta-na-1-tb-uzhe-prodaetsya-no-stoit-ochen-dorogo-01-10-2019>;
2. Secure Digital [Електронний ресурс] – Режим доступу до ресурсу: https://ru.wikipedia.org/wiki/Secure_Digital;
3. Зберігання даних на ДНК [Електронний ресурс] – Режим доступу до ресурсу: <https://www.imena.ua/blog/storage-of-data-on-dna/>;
4. Система зберігання даних на ДНК [Електронний ресурс] – Режим доступу до ресурсу: https://innovationhouse.org.ua/statti/systema_zberihannya_danux_na_dnk_yak_ce_pracyuye_i_dlya_choho_potribno/;

Ситдиков Б.

*Криворожский экономический институт
«Киевский национальный экономический университет
имени Вадима Гетьмана»*

Вдовиченко И.Н.

К.т.н., доцент, Криворожский национальный университет

ПЕРСПЕКТИВЫ РАЗВИТИЯ МИКРОПРОЦЕССОРНЫХ УСТРОЙСТВ

Представлены предпосылки и перспективы развития микропроцессоров. Рассматривается закон Мура и его влияние на увеличение вычислительной мощности CPU.

В основе электронно-вычислительных устройств лежит использование микропроцессоров. Классическими целями развития МП считаются, поднятие тактовой частоты работы и повышение численности одновременно выполняемых команд, за счет увеличения числа транзисторов в процессорах.

19 апреля 1965 года, в журнале Electronics в рубрике «Эксперты смотрят в будущее» вышла знаменитая статья Гордона Мура «Объединение большого количества компонентов в интегральных схемах». В этой статье Мур (как один из основателей корпорации Intel), дал прогноз развития микроэлектронной промышленности на ближайшие десять лет на основании анализа шестилетнего развития микроэлектроники, предсказав, что количество элементов на кристаллах электронных микросхем будет удваиваться каждый год [1].

Впоследствии за этим прогнозом закрепилось название «Закон Мура». И, по иронии судьбы, это эмпирическое наблюдение стало самым знаменитым законом в IT-сфере и полупроводниковой индустрии. Более того, для многих разработчиков микропроцессоров - это стало руководством к действию.

В 2003 году Мур опубликовал работу «No Exponential is Forever: But „Forever“ Can Be Delayed!», в которой признал, что экспоненциальный рост физических величин в течение длительного времени невозможен, и постоянно достигаются те или иные пределы. Лишь эволюция транзисторов и технологий их изготовления позволяла продлить действие закона ещё на несколько поколений.

В 2007 году Мур заявил, что закон, очевидно, скоро перестанет действовать из-за атомарной природы вещества. Размер транзистора на текущий момент времени составляет несколько атомов, при том, что размер атома кремния составляет 0,2 нанометра. Другое физическое ограничение — это скорость передачи данных, которая не может превышать скорость света. Требуется несколько тактовых циклов, чтобы данные попали из одного конца CPU в другой конец. По мере того как микросхемы становятся

крупнее с большим и большим количеством транзисторов, скорость начинает ограничиваться самой передачей данных на микросхеме.



Рис. 1 Закон Мура в действии

Если увеличение плотности транзисторов станет возрастать сегодняшними темпами, то в отсутствие модернизации управления питанием МП начнут выделять десятки тысяч Вт тепла на см². Для того чтоб удовлетворить потребности грядущего будущего, нужно значительно уменьшить мощность потребления.

Область nano технологий включает в себя 4 основных направления. Первым из них является молекулярная электроника (МЭ); вторыми на очереди стоят биохимические и органические решения. Третьим направлением считаются квазифизические решения на основе nano-трубок (НТ). Четвертым направлением считается квантовые компьютеры [2].

В МЭ особое внимание уделяется системам на основе молекул. В котором молекулы представляют квантовую структуру, состоящую из отдельно стоящих атомов. В МЭ движение электронов задается квант-химическими законами. На данный момент производится активный отбор теорий совершенствования МЭ и физических закономерностей действий.

Ученые в области биофизики выявили более 50 комбинаций, на базе которых возможно будут строиться МП - модели нелинейных задач. Процессор, имеющий химическое управление называется белковым.

На основе углеродных нано трубок создаются электронные устройства нано метрового размера. Предполагается, что в ближайшем будущем они сменят составляющие подобного назначения в компьютерах.

Инженеры International Business Machines нашли новый действенный метод увеличить продуктивность микропроцессоров - изготавливать полупроводники из других материалов и усовершенствовать их структуру. В связи с этим максимальной скорости вычисления интегральная микросхема в ее нынешнем виде добьется уже через 5-10 лет.

ЛИТЕРАТУРА

1. IXBT. Закон Мура против нанометров: 02.11.2011 [Электронный ресурс] —<https://www.ixbt.com/cpu/microelectronics-3.shtml>.
2. Денис Крючков. Хабрахабр [Электронный ресурс]: коллективный блог, — Режим доступа к блогу: <https://habr.com/ru/post/307158/>, свободный.

Юхимець М.В.

Державний університет «Житомирська політехніка»

Грабар О.І.

Державний університет «Житомирська політехніка»

ВИБІР КРАЩОГО ТА ПРОСТОГО В ОСВОЄННІ ДВИЖКА ДЛЯ СТВОРЕННЯ ІНДІ-ІГОР

В роботі було проведено аналіз характеристик інді-ігор, аналіз доступності, актуальності та функціональних можливостей інструментальних засобів розробки ігор, і мов програмування. На основі проведеного дослідження було обрано оптимальний движок для створення інді-ігор.

З кожним днем ігрова індустрія розвивається дуже швидко і напевно кожна людина грала або досі грає у відеоігри. Існує велика кількість напрямків у «ігровому світі», серед них стають дуже популярними невеликі проекти, інді-ігри створені одним або маленькою командою розробників, які хочуть зробити щось своє. Іноді це

переростає в грандіозні проекти, які полоняють серця мільйонів геймерів. Тому для того щоб обрати для свого додатку найкраще з кращих було проведено аналіз найкращих інді-ігор.



Рис. 1 –Limbo



Рис. 2– Super Meat Boy



Рис. 3–Cuphead

Таблиця 1 – Порівняння додатків

Назва	Дизайн інтерфейсу	Складна гра	Різноманітність рівнів	Комерційність
Limbo	Похмурий	Так	середня	Платна

Super Meat Boy	Простий	Так	середня	Безкоштовна
Cuphead	Простий та привабливий	Так	Непристойно велика	Платна

Провівши аналіз по таблиці можна дійти до висновку що гра має бути з простим та привабливим інтерфейсом, складною, з середньою або великою різноманітністю рівнів та бажано безкоштовною.

Таблиця 2 – Аналіз доступності та актуальності інструментальних засобів

Назва	Доступність	Мова написання	Дата останнього релізу
Unity	Умовно безкоштовна	C#/JS/Python	25.07.2019
Unreal Engine	Умовно безкоштовна	Unreal Script	31.03.2016
pixi.js	безкоштовна	JS/HTML 5	22.12.2015

Таблиця 3 Аналіз функціональних можливостей інструментальних

Назва	Функціональні можливості				
	2D - графіка	3D - графіка	Обробка колізії	Обробка фізики	Вбудовані механізми відладки
Unity	Так	Так	Так	Так	Так
Unreal Engine	Так	Так	Так	Так	Так
pixi.js	Так	Ні	Ні	Ні	Ні

C#

C# дозволяє стартувати розробку швидше, а це дозволяє швидше отримати прототип рішення Швидкість розробки на початкових етапах проекту значно вище в порівнянні з іншими мовами.

C# спроектований бути кросплатформним, однак його розвиток не пішло в цьому напрямку, тому під Windows утворилася досить повна .net інфраструктура, а на інших платформах рівноцінної інфраструктури не з'явилося. При розробці невеликих проектів продуктивність C # не поступається іншим мовам програмування, однак при збільшенні вихідного коду, алгоритмів і т.д. - швидкість роботи додатків значно падає. C # володіє пристойним кількістю бібліотек зі старту, що істотно полегшує розробку[2].

HTML і JavaScript

Чи є незамінними при створення простих веб-додатків та ігор, проте на

цьому переваги закінчуються.

C ++

Є основною мовою програмування при розробці ігор, так як дозволяє здійснити повний контроль над засобами і логікою. Є абсолютним рекордсменом по продуктивності, по платформах, по сумісності і за кількістю існуючих бібліотек серед інших мов програмування. Однак багато програмістів виділяють головний його недолік - складність освоєння даної мови.

Таблиця 4 – Аналіз мов програмування

Критерії	C#	JavaScript/HTML	C++
Продуктивність	3	2	5
Кількість бібліотек	3	2	5
Зручність програмування	4	3	3
Складність пізнання	4	4	3
Кросплатформність	3	5	4

ВИСНОВКИ

1. Для реалізації новачкам найбільше підійде Unity. Він має значну перевагу над рixi.js та набагато простіший для вивчення ніж Unreal Engine.

2. Додаток повинен бути складним, UI дизайн має бути функціональним та ергономічним і динамічним. Можемо зробити висновок що для реалізації нашого проекту більш за все буде підходити движок Unity та мова програмування C#.

ЛІТЕРАТУРА

1. <https://dtf.ru/gamedev/7227-orel-ili-reshka-sravnenie-unity-i-unreal-engine>
2. <https://habr.com/ru/company/piter/blog/269415/>
3. <https://itproger.com/news/124>

СЕКЦІЯ 3. DESIGN. ПРОЕКТУВАННЯ КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ

*Котвицький О. В.,
Державний університет «Житомирська політехніка»
Вакалюк Т. А.,
Державний університет «Житомирська політехніка»*

ВИКОРИСТАННЯ ХМАРНИХ ТЕХНОЛОГІЙ В ОРГАНІЗАЦІЇ НАВЧАЛЬНОГО ПРОЦЕСУ

Проаналізовано програмні рішення для управління віддалених серверів: Amazon Web Services, Jelastic PaaS. Дано характеристику Jelastic PaaS та зроблено порівняння з аналогічним рішенням від компанії Amazon.

На сьогоднішній день майже кожен заклад вищої освіти має свою веб-сторінку. Декі заклади освіти мають свої освітні портали, або навіть веб-системи для організації олімпіад з програмування. Постає питання організації та підтримки серверів, що використовуються для зберігання та функціонування веб-сайтів, тобто так званого «хостингу». Зазвичай, веб-сервери налаштовуються вручну та потребують постійного контролю з боку системних адміністраторів, особливо якщо серверів декілька, або вони згруповані у кластери.

Такі програмні продукти, як: Amazon Web Services та Jelastic PaaS, надають програмне забезпечення для контролю, швидкого та зручного налаштування як одиничних серверів, так і серверів, які знаходяться у кластерах.

Jelastic PaaS – це так звана платформа-як-сервіс (Platform as a service) [1]. Це означає, що програмний продукт є додатковим шаром між сервером та користувачем (користувач в даному випадку – це адміністратор сервера). Користувач даної платформи може зручно створювати віртуалізовані контейнери, які є абстрактними одиницями поділу ресурсів сервера, таких як: постійна пам'ять (ROM), оперативна пам'ять (RAM), завантаження центрального процесора (або процесорів). Також дані контейнери вже налаштовані для завантаження (deploy) на них готових веб-сайтів для їх функціонування

та зберігання. Для кожного стеку веб-технологій є свій налаштований шаблон контейнера. Так, якщо сайт використовує мову PHP, сервер Apache та базу даних MySQL, в платформі Jelastic обирається відповідний шаблон, який забезпечує функціонування даної зв'язки програмного забезпечення. Все, що користувачу потрібно зробити – це завантажити свій сайт в контейнер та обрати доменне ім'я.

У порівнянні з Amazon Web Services, Jelastic має зручний та зрозумілий інтерфейс, систему автоматичного розширення контейнерів на серверах, якщо навантаження зростає і заздалегідь зарезервованих ресурсів користувачем не вистачає [2].

На рис. 1 зображено графік, як працює автоматичне розширення контейнерів. Reserved Resources – це ресурси, виділені користувачем для контейнера (вони не залежать від навантаження). Dynamic Resources – це ресурси, які виділяються контейнеру за потреби. Maximum Scaling Limit – це межа виділення ресурсів контейнеру.

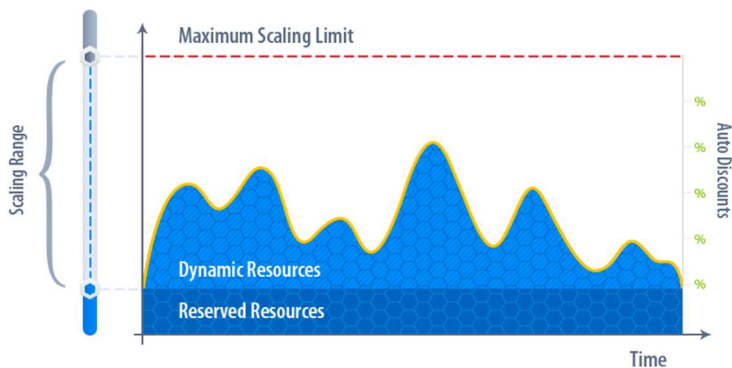


Рис. 1. Графік використання ресурсів контейнером

Таким чином, Jelastic PaaS є зручним рішенням для автоматизації налаштування та адміністрування серверів. Даний програмний продукт забезпечує зберігання та функціонування веб-сайтів. У порівнянні з Amazon Web Services, Jelastic має автоматичне розширення контейнерів, а отже економію ресурсів, які не використовуються в певний момент часу.

ЛІТЕРАТУРА

1. Офіційний сайт Jelastic [Електронний ресурс] – Режим доступу до ресурсу: <https://jelastic.com/>
2. Офіційний сайт Amazon Web Services [Електронний ресурс] – Режим доступу до ресурсу: <https://aws.amazon.com/>

Бойко Д. С.,
Криворізький національний університет
Кумченко Ю. О.
канд. техн. наук, доцент,
Криворізький національний університет

ВИКОРИСТАННЯ СУЧАСНИХ ТЕХНОЛОГІЙ БЕЗДРОВОТИХ МЕРЕЖ У ЗАКЛАДАХ ВИЩОЇ ОСВІТИ

Проаналізовано перспективи застосування сучасних технологій бездротових мереж: Wi-Fi6, WPA3, Wi-Fi Mesh - у закладах вищої освіти. Сформульовано висновки, щодо доцільності використання проаналізованих технологій.

Сучасний процес проектування та розгортання бездротових мереж за технологією Wi-Fi у закладах вищої освіти (ЗВО) має ряд проблем, зокрема: збільшення числа користувачів мережі, обсягів інформації, яку необхідно передати, та підвищення захищеності мережі загалом. Для вирішення цих проблем, комітетом зі стандартів LAN/MAN Інституту інженерів з електротехніки та електроніки (IEEE) і організацією Wi-Fi Alliance було розроблено новий стандарт бездротових мереж Wi-Fi 6 (802.11ax), а також набір протоколів захисту передачі даних у бездротових мережах WPA3[1].

Стандарт Wi-Fi 6 забезпечить більшу кількість одночасно працюючих користувачів у ЗВО, за рахунок використання технології множинного доступу з ортогональним частотним розподілом каналів, що дозволяє оптимізувати передачу даних, шляхом розподілу частотного каналу на підрядні шириною близько 78 кГц. Теоретична максимальна швидкість передачі даних стандарту Wi-Fi 6 – 9,6 Гбіт/с[1]. У разі, якщо поряд обладнано декілька існуючих точок доступу до бездротової мережі і частотні канали завантажені, підви-

щується ризик втрати пакетів та помилок передачі даних, для вирішення цієї проблеми у Wi-Fi 6 реалізовано маркування трафіку, що дозволяє розрізняти власний трафік та трафік інших точок доступу, ефективніше формувати запити на передачу даних, зменшуючи затримки та підвищуючи надійність та ефективність передачі даних у ЗВО.

За безпеку передачі даних відповідає новий набір протоколів захисту бездротових мереж WPA3, до складу якого входить новий метод автентифікації пристроїв – SAE (Simultaneous Authentication of Equals), який визначає спосіб взаємодії пристрою користувача та точки доступу до бездротової мережі, при обміні інформацією та криптографічними ключами, таким чином при кожному новому з'єднанні встановлюється новий шифрувальний пароль. Також у новому стандарті захисту використовуються 192-бітні протоколи захисту та шифрування даних, для захисту всієї мережі, WPA3 може використовувати 256-бітний протокол Galois/Counter Mode для шифрування, 384-бітний Hashed Message Authentication Mode режим для створення і підтвердження ключів та алгоритми Elliptic Curve Diffie-Hellman exchange, Elliptic Curve Digital Signature Algorithm для автентифікації ключів, що дозволить забезпечити захист інформації на усіх етапах передачі інформації від реєстрації пристрою в мережі до безпосередньої передачі даних. Для захисту користувача, у відкритих мережах ЗВО, використовується протокол Enhanced Open, що будуватиметься на принципі опортуністичного бездротового шифрування, для захисту від пасивного прослуховування, шляхом покращення шифрування даних[1].

Використання технології Wi-Fi Mesh у ЗВО дозволить швидко та ефективно розгорнути бездротову мережу на велику відстань, перевагою пристроїв побудованих на основі даної технології є можливість асинхронного спілкування між точками доступу для пересилання пакетів даних без суттєвого завантаження частотного діапазону та функціонування в рамках єдиної мережі, що означає кінцевому користувачу не доведеться при переміщенні підключатись до нової мережі, через втрачання сигналу віддаленої точки доступу, також дана технологія дозволяє спростити процес налаштування обладнання, так як, точки доступу можуть пересилати дові-

рені налаштування на сусідні станції та встановлювати їх автоматично. Декілька точок доступу покривають більшу відстань ніж традиційні, за рахунок функціонування у єдиному комплексі[1].

ВИСНОВКИ

Таким чином, новий стандарт бездротових мереж Wi-Fi 6 дозволяє будувати мережі більш високої ємності та являє собою більш надійне середовище передачі даних з мінімальними затримками та більш високою швидкістю передачі інформації, так само набір протоколів захисту бездротових мереж WPA3 забезпечує конфіденційність, цілісність та захист інформації, що передається у бездротовій мережі, технологія Wi-Fi Mesh, дозволяє пришвидшити та спростити процес налаштування обладнання та забезпечити зв'язок у рамках однієї мережі на велику відстань. Отже, пропонується впровадження даних сучасних технологій при розробці комп'ютерних мереж у ЗВО для забезпечення швидкого, ефективного та безпечного обміну інформацією.

ЛІТЕРАТУРА

1. 802.11 Wi-Fi Standarts [Електронний ресурс] – Режим доступу до ресурсу: <https://www.networkworld.com/article/3238664/80211-wi-fi-standards-and-speeds-explained.html>

*Лопатина А.А., Лавров А.А.,
Харьковский национальный университет радиоэлектроники
Хаханова И.В.,
д.т.н., проф., Харьковский национальный университет радиоэлектроники*

РАЗРАБОТКА СИСТЕМЫ АВТОМАТИЗАЦИИ ТЕСТИРОВАНИЯ БОЛЬШОГО ОБЪЕМА ПРОЕКТОВ

Представлена концепция системы автоматизированного тестирования проектов в среде EDA Playground и рассмотрены инструменты, подходящие для реализации подобных систем.

На сегодняшний день любая сфера человеческой деятельности находится в процессе развития, используя все современные возможности и наработки. Не исключением стала и образовательная сфера. Одним из стремлений человека является полное или частичное замещение человеческого труда, машинным, то есть автоматизация. Один из относительно новых методов контроля знаний – машинный контроль. Этот метод позволяет с помощью автоматизированной системы проверки результатов производить тестирование работ. В результате чего повышается временная, количественная и качественная составляющие процесса проверки работ. В данном проекте предлагается создать систему автоматизированного тестирования проектов, созданных с использованием среды EDA Playground.

Веб приложение EDA Playground является бесплатной средой разработки для моделирования интегральных схем, которая не нуждается в установке специального программного обеспечения на свой компьютер. Среда позволяет создавать модели интегральных схем и описывать драйвер моделирования, используемый для тестирования модели. Для этого используется редактор кода модели и редактор кода теста. Также преимуществами данной среды, помимо бесплатного доступа, является наличие нескольких симуляторов разных фирм, которые постоянно обновляются [1]. Однако, при работе с большим количеством проектов в данной среде много времени уходит на ручную проверку работ.

Целью исследования является автоматизация тестирования большого количества проектов, созданных потоком студентов в среде EFA Playground. Необходимо проверить каждый проект на соответствие заданию варианта и корректность выполнения задания.

Главной задачей является создание скрипта автоматизации проверки проектов студентов. В соответствии с представленными ссылками скрипт скачивает содержимое проекта в виде архива, содержащего код модели, код теста и результаты моделирования. Далее скрипт производит извлечение файлов из архива. Следующим шагом является анализ результатов моделирования – проверка проекта на соответствие выданному варианту и корректность выполненной задачи. Результаты анализа каждого проекта заносятся в результирующую таблицу для дальнейшего анализа преподавателем.

Для реализации скрипта могут быть использованы такие языки общего назначения, как Python, JavaScript и Ruby.

JavaScript — мультипарадигменный язык программирования. Поддерживает объектно-ориентированный, императивный и функциональный стили. В большинстве случаев при упоминании JavaScript подразумевается так называемый клиентский JavaScript, интерпретатор которого встроен в веб-браузеры. Однако он изначально был разработан как универсальный язык программирования для встраивания в любое приложение и обеспечения возможности написания в нем сценариев.

Ruby — объектно-ориентированный язык программирования общего назначения, характеризуется динамической типизацией и автоматическим управлением памятью. К преимуществам языка можно отнести продвинутое методы манипуляции строками и текстом, простой и чистый синтаксис, расширение возможностей при помощи библиотек, написанных на C или Ruby. К недостаткам относятся недостаток информационных ресурсов, посвящённых Ruby и меньшую производительность по сравнению с другими языками.

Python — язык программирования общего назначения, широко применяется как интерпретируемый язык для скриптов различного назначения. Как и Ruby, Python имеет целью приблизить синтаксис реальной программы, написанной на нём, к описывающему задаче псевдокоду, что позволяет программисту уменьшить объём программы. Основным преимуществом языка являются множество полезных библиотек и расширений языка, которые можно легко использовать в своих проектах благодаря предельно унифицированному механизму импорта и программным интерфейсам [2]. Классический Python имеет общий со многими другими интерпретируемыми языками недостаток — сравнительно невысокую скорость выполнения программ. В некоторой степени ситуацию улучшает сохранение байт-кода, которое позволяет интерпретатору не тратить время на синтаксический разбор текста модулей при каждом запуске.

ВЫВОДЫ

В работе были перечислены преимущества автоматизации в сфере образования, предложена концепция системы автоматизированного тестирования проектов в среде EDA Playground и рассмотрены инструменты, наиболее подходящие для ее реализации.

ЛИТЕРАТУРА

1. EDA Playground Documentation [Электронный ресурс] – Режим доступа: <https://eda-playground.readthedocs.io/en/latest/faq.html>
2. Свейгарт, Эл. Автоматизация рутинных задач с помощью Python: практическое руководство для начинающих. – М.: ООО «И.Д. Вильямс», 2017. – 592с.

Савостенок В. Є.

Криворізький національний університет

Костенко В. В.

асистент, Криворізький національний університет

РОЗГОРТАННЯ ШПД КОМП'ЮТЕРНОЇ МЕРЕЖІ ЗА ТЕХНОЛОГІЄЮ GPON

Проаналізовано типові проблеми розгортання ШПД комп'ютерної мережі за технологією GPON, та відзначено її переваги порівняно з іншими технологіями.

GPON – це технологія швидкого розвитку широкосмуговій передачі даних комп'ютерної мережі з оптичного волокна[3].

Варто виділити типові проблеми, з якими стикаються провайдери, при розгортанні пасивної оптичної мережі GPON:

– Необхідність більшої потужності оптичного сигналу. При розгалуженні в співвідношенні 1:2 енергетичний потенціал лінії зв'язку падає на 3'4 дБ. Тобто при розгалуженні в 1:64 потенціал лінії зв'язку зменшується на 20,4 дБ. В цій ситуації, всі оптичні передавачі повинні забезпечувати в 110 разів більшу потужність оптичного сигналу при передачі на тій самій відстані;

– Загальна смуга пропускання. Смуга пропускання в оптоволоконних ліній мережі GPON використовується більшою кількістю абонентів. Технологія GPON забезпечує пропускну здатність спадаючого потоку, що дорівнює 2,5 Гбіт/с, вона не відповідатиме зросту майбутніх вимог абонентів в довгостроковій перспективі. Особливо, якщо деяку частину смуги пропускання необхідно резервувати для поточкових послуг;

– Неоптимальне використання ресурсу мережі. Зазвичай виконується одночасне підключення ліній зв'язку для всіх абонентів в даному районі. При розгортанні послуг провайдери рідко досягають 100% покриття. Цей показник близько до 30%, що означає, що частина структури GPON простоює, а мережа використовується не оптимально;

– Складність обслуговування, знаходження та усунення несправностей. Пасивні оптичні розгалужувачі не можуть передавати інформацію про несправності в центр управління мережею. Цим обумовлюється складність виявлення несправності оптоволоконної лінії між перехідником і точкою терміналу оптичної мережі (ONT) абонента. Це ускладнює знаходження і усунення несправностей в мережах GPON і підвищує витрати їхньої експлуатації. При пошкодженні точки терміналу оптичної мережі (ONT) вона може передавати в оптоволоконній лінії постійний світловий сигнал, що призводить до порушення зв'язку всіх абонентів цієї мережі, причому знайти пошкоджені пристрій дуже важко[1].

Не дивлячись на вищесказані недоліки, є значні переваги:

– необхідність електроживлення елементів мережі відсутня;

– на обслуговування витрачаються невеликі витрати;

– без перерви зв'язку, можливість підключення абонентів;

– розширення смуги можна застосовувати динамічно;

– економія оптичних випромінювачів.

При розгортанні мережі GPON, перераховані її недоліки являються менш значущі, ніж її переваги[2].

ВИСНОВКИ

На сьогодні GPON являється розвиваючою оптичною мережевою технологією. У розвинених країнах світу кількість абонентів GPON зростає. Вартість активного обладнання єдиний фактор, що стримує впровадження GPON технології. Будівництво мереж GPON рекомендовано для багатоповерхової забудови або для приватного сектору і офісних центрів.

ЛІТЕРАТУРА

1. Принципы построения сетей абонентского доступа [Електронний ресурс] – Режим доступу до ресурсу: islitelnye-sistemy-seti-i-

- telekommunikatsii/14-printsipy-postroeniya-setej-abonentskogo-dostupa#14.6.2.
2. Практика внедрения пассивных оптических сетей (GPON) [Электронный ресурс] – Режим доступа до ресурсу: <https://deps.ua/knowegable-base-ru/articles/item/462-praktika-vnedrenija-passivnyh-opticheskikh-setej-pon.html>.
 3. Технология PON: оптимальное решение для построения сетей в частном секторе [Электронный ресурс] – Режим доступа до ресурсу: <http://mediasat.info/2015/12/15/pon/>.

*Первой Д.В.,
Одеський національний політехнічний університет
Кривоконь А.С.,
Одеський національний політехнічний університет
Богданова Л.О.,
ст. викладач, Одеський національний політехнічний університет
Ситніков В.С.,
д.т.н., проф., Одеський національний політехнічний університет*

АВТЕНТИФІКАЦІЯ КОРИСТУВАЧІВ НА WEB-ДОДАТКАХ

В роботі проведено аналіз модель автентифікації, взаємодії серверів з даними користувачів. Запропонована модель безпечної автентифікації, обробки даних, яка може використовуватися багато разів у мережі.

Авторизація в Інтернеті присутня практично на всіх сайтах. Ми стикаємося з нею регулярно: на форумах, в соціальних мережах, на сайтах банків або в коментарях до новин. Але які технології ховаються за кнопкою «Увійти», знають далеко не всі.

Механізм авторизації починається з ідентифікації користувача. Користувач вводить своє ім'я користувача та пароль в форму авторизації і відправляє їх системі. Використовуючи ці два параметри, система шукає обліковий запис користувача у своїй базі даних і, якщо користувач знайдений, створює для нього сесію.

Слід пам'ятати, що паролі користувачів не зберігаються в базах даних в чистому вигляді. Замість них в базах зберігаються спеціальні буквено-цифрові ключі. Ці ключі - результат застосування

до паролів спеціальних криптографічних функцій. Такі функції незворотні, і відновити пароль за їх результатами вкрай складно. Це зроблено для того, щоб в разі витоку даних зловмисники не могли скористатися паролем користувача для спроби входу в його обліковий запис на інших сайтах адже у багатьох користувачів на всіх сервісах встановлено однаковий пароль. Це крайні небезпечно, тому що, якщо один із цих сервісів буде дискредитований, пошта та пароль одразу попадуть до рук зловмисників і отримають доступ до багатьох сервісів жертв.

Всі типи автентифікації можна розділити за наступними методами: за логіном та паролем; за сертифікатом; за одноразовим паролем; за ключем доступу; за токеном.

Деякі з цих методів чимось схожі один на одного, але мають різний набір переваг і недоліків. При цьому усі крім першого методу ніяк не обходяться без початкової реєстрації та авторизації. Разом, сучасні методи двоетапної автентифікації наприклад, OTP та SMS допомагають захиститися від загроз повторного використання паролів і витоку бази хешів паролів з сервера, але все-таки показують себе неефективними проти ретельно продуманого фішингу або направленої атаки, що включає в себе крадіжку самого телефону.

Відомо, що найбезпечніший спосіб автентифікації, це відсутність передачі логіну та паролю та делегування на інші фізичні пристрої. Для делегування процесу запропоновано використовувати месенджер Telegram адже він вважається найбезпечнішим на даний момент, через своїх методів шифрування. Оскільки для авторизації необхідні певні заходи безпеки, буде використаний Telegram API для доступу до можливостей його ботів. Структура системи авторизації буде розділеною на частини, клієнтської бібліотеки яка буде відображати кнопки «Увійти через Телеграм» після натискання на яку скрипт сформує запит до боту на авторизацію. Серверної частини, котра відповідає за логіку роботи самого бота а також обробляє запити від веб хука відправлені з боку серверів Telegram.

Процес автентифікації проходить наступним чином. Користувач заходить до веб додатку на якому в формі авторизації підключений скрипт, що додає кнопку «Увійти через Телеграм». Коли користувач натискає на неї, відобразиться поле для вводу логіну Телеграм, після вводу, відправиться запит до серверу на якому знаходиться бот і шукає облікові записи користувача, якщо користувач

не встановив бота, тоді повернеться відповідь с повідомлення про те що необхідно додати бота. Коли знайдені налаштовані записи, бот присилає повідомлення у месенджер с запитом чи дійсно користувач хоче увійти на сайт а через який обліковий запис. Після підтвердження, бот отримує необхідні данні та посилає запит до веб хуку сайту, і передає необхідні данні, веб хук обробляє данні, реєструє користувача якщо він не зареєстрований та автентифікує його, без необхідності вводити та зберігати пароль. Клієнтський скрипт посилає запит методом POST якій містить у тілі логін Телеграм ведений користувачем, url адресу веб хуку, ip адреса, useragent та токен якій генеруються при ініціалізації скрипту кожен раз різний, формується він з випадкової комбінації латинських символів за цифр. Веб хук отримує запит, та шукає у бази користувача за логіном, якщо користувача не має у базі, формується відповідь з повідомленням про це, та користувачу пропонується додати бота. Після того як користувач знайден, бот відправляє йому повідомлення яке містить сайт з якого прийшов запит на автентифікацію, ip адреса, та useragent. Наступне повідомлення містить вибір список акаунтів даного користувача та пропонує підтвердити дію обравши акаунт чи відхилити дію. Після підтвердження бот звертається до сервісу якій посилає запит до веб хуку сайту клієнту за url який був у тілі запиту. Запит містить у тілі ім'я, фамілію, поштову скриньку та номер телефону, для верифікації запиту, генерується цифровий підпис за форматом усіх полів та токєну через роздільник; ця строчка шифрується методом HmacSHA256.

В ході експерименту було визначено, що використання даної моделі дозволило скоротити час реєстрації та автентифікації на 37%, в порівнянні з аналогічною роботою інших моделей автентифікації. Також слід зазначити, що розроблені моделі дозволяють використовувати декілька типів облікових записів для різних типів сайтів.

*Банделюк О.С.
бакалавр, Одеський національний політехнічний університет
Донцов О.Ю.
бакалавр, Одеський національний політехнічний університет
Ситніков В.С.
д.т.н., проф., Одеський національний політехнічний університет
Богатова О.О.
ст. викл, Одеський національний політехнічний університет*

ПРОГНОЗУВАННЯ НАВАНТАЖЕННЯ КОМП'ЮТЕРНИХ МЕРЕЖ НА ОСНОВІ МОДЕЛІ ARMA

Більшість локальних комп'ютерних мереж не застраховані від раптових навантажень які можуть спричинити за собою втрати даних, порушення їх цілісності, порушення безпеки мережі і безліч інших проблем. Запропоновано застосувати модель ARMA для прогнозування у реальному часі пікового навантаження що може допомогти системним адміністраторам локальних мереж вжити необхідних заходів, або надати інформацію автоматизованим системам моніторингу та адміністрування мереж для подальшої обробки.

Через локальну мережу постійно проходить трафік, у деякий момент кількість трафіку може стати значно більше ніж деякий вузол може витримати, тоді він може вийти з ладу.

В роботі розглянуто ряд моделей прогнозування навантаження комп'ютерної мережі, показані їх переваги та недоліки та запропоновано використати модель ARMA. Оскільки модель ARMA об'єднує в собі авторегресійну модель та модель ковзного середнього.

Модель ARMA записується у формі ARMA(p, q) та має наступний вигляд

$$y_t = \alpha_1 y_{t-1} + \alpha_2 y_{t-2} + \dots + \alpha_p y_{t-p} + \varepsilon_t - \beta_1 \varepsilon_{t-1} - \beta_2 \varepsilon_{t-2} - \dots - \beta_q \varepsilon_{t-q},$$

де, ε_t – білий шум; p – порядок авторегресійної складової; q – порядок ковзних середніх.

Така модель визначається як лінійна модель множинної регресії, в котрій у якості пояснюючих змінних виступають минулі

значення залежної змінної, а у якості регресійного залишку – ковзке середнє з елементів білого шуму. Правильно налаштована модель ARMA(p, q) дає більш точний прогноз на більший інтервал прогнозування, ніж авторегресійна модель та модель ковзного середнього, тому для вирішення задачі було обрано саме її.

За результатами експерименту визначено, що зі збільшенням інтервалу прогнозу його точність зменшується, крім того на інтервалі до 20 хвилин прогноз у межах допустимої точності. Отже, при застосуванні моделі ARMA дає змогу у реальному часі на інтервалі до 20 хвилин вперед прогнозувати навантаження мережі, що також обумовлене досить не стабільною сутністю мережевого трафіку.

Баев В.Ф.

бакалавр, Одесский национальный политехнический университет

Бреставец В.В.

бакалавр, Одесский национальный политехнический университет

Ситников В.С.

д.т.н., проф., Одесский национальный политехнический университет

Богатова О.А.

ст. преп., Одесский национальный политехнический университет

ФАКТОРЫ ВЛИЯЮЩИЕ НА РАНЖИРОВАНИЕ ВЕБ-САЙТА

В работе рассматриваются факторы, влияющих на его продвижение. Каждый фактор имеет свой вес в алгоритме поисковых систем, таким образом используя определенные факторы можно повысить позицию сайта в результате поиска, а также количество его пользователей

Влияние разных факторов каждый раз меняется с обновлением алгоритма ранжирования поисковой системы. В связи с этим возникает необходимость выявить основные факторы, которые будут весомыми на долгосрочную перспективу.

На основании исследований 2017 года, проводимыми специалистами в области оптимизации сайтов, был выявлен ряд

факторов, которые могут существенно влиять на позиции сайта и число пользователей. В таком случае возникает ряд задач анализа факторов: провести классификацию факторов, для упрощения анализа; определить те факторы, которые остаются актуальными в данный момент времени и на перспективу; рассмотреть возможность добавить ряд других актуальных факторов, влияющих на продвижение сайта.

Согласно алгоритму ранжирования поисковых систем определяется - какой документ из базы данных (а именно какая интернет страница) будет находиться на первых и последующих позициях, по ключевому запросу, который ввел пользователь. Каждая поисковая система имеет свой уникальный алгоритм ранжирования, а точная формула и вес всех факторов, влияющих на выдачу, держится в секрете и является главной коммерческой тайной. Таким образом, воздействовать на ранжирование сайта можно благодаря конкретным факторам.

Для того, чтобы сайт начал ранжироваться, он предварительно индексируется. Под индексацией понимается сканирование страниц сайта поисковым роботом, а также последующее занесение их в общую базу. В основном используется два известных вида индекса: прямой и инвертированный.

Наиболее весомые факторы не теряют своей актуальности. Однако есть и те, которые приобрели больший вес по сравнению с прошлым периодом времени. Таким образом, факторы, влияющие на контент, остаются весомыми и сегодня. К таким можно отнести: наличие тега-заголовка (а также ключевого слова в нем); наличие тега-описания (а также ключевого слова в нем); наличие тега H1 (а также ключевого слова в нем).

Каждый из этих факторов влияет на такое понятие как кликабельность. Например, поисковая система Bing учитывает клики пользователей при составлении выдачи. Чем лучше каждый из них оптимизирован, тем выше этот показатель. Помимо этого, сам контент может быть как качественным, так и полезным. В понимании поисковой системы, это два абсолютно разных понятия. Для качественного контента важны такие факторы: плотность ключевой фразы в тексте ; оптимизация изображений; грамматика и правописание; использование мультимедиа; ключевое слово в первых 100 словах контента.

В свою очередь полезный контент характеризуется следующими факторами: глубокое освещение темы; LSI в контенте; суть контента; новизна контента; обновление контента; коммерческие страницы (страницы связи, доставки, способов оплаты и т.д.)

При увеличении качества и полезности контента, прямо пропорционально увеличиваются наиболее важные факторы, как показатель отказов, время пребывания, и циклический трафик.

Помимо этого, на итоговую оценку влияет ссылочный профиль сайта. Под ссылочным профилем подразумеваются ссылки, которые ведут на сайт от других сайтов. В качестве основных факторов можно выделить следующие: число ссылающихся доменов; авторитет ссылающейся страницы; авторитет; ссылающегося домена; ссылки с социальных сетей

Ключевым фактором являются оценки других сайтов и их страниц. Тем выше оценка той или иной страницы, тем существеннее будет ссылка. Чем выше этот параметр у страницы, тем весомее будет ссылка с такой страницы. Следует отметить, что чем выше PageRank страницы-донора, тем выше будет влияние ссылки.

Помимо факторов контента и ссылочного профиля необходимо также упомянуть факторы технической оптимизации сайта и его страниц. Эти факторы напрямую влияют на три вышеупомянутых фактора: время пользователя на странице, показатель отказов и на цикличность трафика. К наиболее важным стоит отметить следующие: загрузка страниц через HTML; наличие и корректно составленный файл robots.txt; наличие sitemap; мобильная оптимизация; дублированный контент на сайте.

Данный анализ позволил проанализировать несколько сайтов и предложить их оптимизацию по полученным факторам, что позволило значительно продвинуть рабочие сайты и увеличить число пользователей.

Таким образом, анализ известных факторов ранжирования, актуальных в данный период времени, и выявление наиболее значимых факторов, которые в большей степени влияют на позиции сайта в поисковых системах, позволило улучшить рейтинг ряда сайтов, что подтверждено экспериментальной проверкой и выработанными рекомендациями.

Букреев А.В.
бакалавр, Одесский национальный политехнический университет
Тютюник Д.Г.
бакалавр, Одесский национальный политехнический университет
Ситников В.С.
д.т.н., проф., Одесский национальный политехнический университет
Маруцак Ю.В.
ст. преп., Одесский национальный политехнический университет

АНАЛИЗ ПСИХОЛОГИИ И ЮЗАБИЛИТИ ВЕБ-ДИЗАЙНА

Веб-дизайн упрощает процесс взаимодействия пользователя и машины посредством удобного, легко читаемого интерфейса, а также позволяет с головой погрузиться в виртуальную среду, созданную с целью комфортного пребывания в ней пользователя. В работе рассматриваются основные понятия юзабилити и психологическая сторона веб-дизайна, правильное использование которых влияет на решения пользователя.

Со времени возникновения сети интернет-пользовательская виртуальная среда претерпевает непрерывные изменения в своём внешнем виде и назначении. Одной из проблем во все времена была проблема грамотной компоновки элементов на странице, построения композиции, сочетания цветов и т.д. На помощь современным дизайнерам приходит психология и юзабилити дизайна, позволяющие повысить удобство, дружелюбность, интуитивность и понятность разрабатываемых сайтов.

Понятие веб-дизайн появилось сравнительно недавно. Его развитие до текущего состояния происходило в несколько этапов начиная табличной и заканчивая адаптивной вёрстками.

Юзабилити – это оценка удобства и простоты использования продукта или веб-сайта. Он основывается на трех принципа дизайна:

- *Firmitas*(прочность — лат.) — долговечность и устойчивость дизайна;

- Utilitas(практичность — лат.) — полезность дизайна и пригодность для заинтересованных в нём людей;

- Venustas(привлекательность — лат.) — красота дизайна.

По современным меркам юзабилити - удобный дизайн должен:

- быть простым в использовании настолько, чтобы стать привычным для пользователя во время первого же контакта с сайтом;

- быть простым в достижении пользователем своих целей во время использования сайта;

- быть прост в запоминании, чтобы пользователь мог вспомнить как его использовать при последующих визитах.

Для достижения описанных требований необходимо уделять внимание визуальным факторам:

- размерность и цвет шрифта - использование цветов удобных для чтения, хорошо различимых на фоне;

- брендинг - логотип компании, согласно модели движения глаз при чтении следует размещать в левом верхнем углу, так как это первое место, куда посмотрят люди, читающие слева направо;

- навигация - чтобы пользователь мог получить максимальную отдачу от сайта, нужно обеспечить ему быстрое попадание из точки А в точку Б путём использования навигации;

- цвета - цвета должны быть совместимы согласно палитре комплементарных цветов для повышения читаемости и эстетической привлекательности;

- абзацы - абзацы должны быть чёткими и легко узнаваемыми, контент разбит на небольшие блоки, чтобы предотвратить перегрузку пользователя информацией.

Говоря о психологии веб-дизайна, следует особым образом выделить три компонента: цвет; шрифт; работа с пространством.

Существует понятие "белое пространство" — области дизайна, не требующие внимания пользователя. Оно играет важную роль в дизайне, так как даёт визуальное место для отдыха посетителя сайта. Например, если каждый сантиметр страницы заполнен графиками, рисунками и текстом, пользователь может почувствовать дискомфорт. Для предотвращения этого используется работа с пространством, в частности стоит выделить важный принцип — принцип близости. Согласно ему набор

объектов расположенных близко друг к другу воспринимается пользователем как группа.

Экспериментальная проверка эффективности юзабилити была проведена путём юзабилити-тестирования. Группа из пяти человек с разным уровнем навыка владения компьютером выполнила ряд заданий на одном и том же сайте до оптимизации и после. По завершении им следовало оценить сайт по ряду критериев по шкале от 1 до 5. В ряд критериев входили такие:

- Сайт в общем привлекателен?
- Графика сайта приятная?
- На сайте хороший баланс графики и текста?
- Цвета, используемые на сайте, привлекательны?
- Типография (надписи, заголовки) привлекательна?
- Легко найти свой путь по сайту?
- Я могу быстро получить информацию?
- Легко запомнить, где найти вещи?
- Информация легко читается?
- Сайт отражает прогрессивный, передовой дизайн?

Обработка результатов тестирования показало, что оптимизация юзабилити послужила увеличению эффективности работы дизайна и восприятия его пользователем.

Таким образом, проанализировав и выделив основные принципы построения дизайна и юзабилити веб-страницы, позволило улучшить восприятие сайта посетителями, что подтверждено в процессе тестирования.

*Підгаєцький О.С.,
Державний університет «Житомирська політехніка»
Левківський В.Л.
ст. викладач кафедри комп'ютерних наук,
Державний університет «Житомирська політехніка»*

ОБҐРУНТУВАННЯ ВИБОРУ ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ ТА ВИМОГ ДО АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ

В роботі проведено аналіз програмного забезпечення, на основі якого було обґрунтовано вибір технологій для розробки веб додатка.

Інтернет програмування розвивається кожного дня. Розглянемо програми, в яких будемо створювати цей програмний продукт.

Adobe Photoshop: продукт компанії Adobe Systems. Це графічний редактор є лідером ринку в області комерційних засобів редагування растрових зображень, і найвідомішим продуктом фірми Adobe. Часто цю програму називають просто Фотошоп. У наш час Photoshop доступний на платформах Mac OS X/Mac OS і Microsoft Windows. Підтримується обробка зображень, як з традиційною глибиною кольору (8 біт, 256 градацій яскравості на канал), так і з підвищеною (16 біт, 65536 відтінків в кожному каналі). Можливе збереження у файлі додаткових елементів, як то: напрямних (Guide), каналів (наприклад, каналу прозорості — Alpha channel), шляхів обтравки (Clipping path), шарів, що містять векторні і текстові об'єкти. Файл може включати колірні профілі (ICC), функції перетворення кольору (transfer functions). У програмному продукті за допомогою Фотошопу було створено декілька макетів, з яких був обраний макет з зручним інтерфейсом та стильним дизайном

WebStorm: інтегрована середа розробки у якій була розроблена бакалаврська робота. В порівнянні зі своїми конкурентами набагато простіший в інтерфейсі та зручніший у використанні, завдяки підсвітці коду та можливості роботі на два екрани.

Основні можливості:

Інтеграція з системами управління версіями Subversion, Git, GitHub, Perforce, Mercurial, CVS підтримуються з коробки з можливістю побудови списку змін і відкладених змін.

Інтеграція з системами відстеження помилок.

Модифікація файлів .css, html, .js з одночасним переглядом результатів (Live Edit, в деяких джерелах ця функціональність називається «редагування файлів на льоту» або «в реальному часі» або «без перезавантаження сторінки»).

Віддалене розгортання за протоколами FTP, SFTP, на монтюваних мережевих дисках тощо з можливістю автоматичної синхронізації

Можливості Zen Coding і Emmet.

HTML5 (HyperText Markup Language) – мова гіпертекстової розмітки сторінки. З Cascading Style Sheets(CSS) і JavaScript, вона утворює триаду основних технологій для World Wide Web. HTML 5 — це на даний час остання переробка стандарту HTML. Робота над створенням специфікації, відома під назвою «Web Applications 1.0», розпочата WHATWG в червні 2004 року.

HTML 5 спрямований на скорочення використання заснованих на плагінах RIA-технологій, таких як Adobe Flash, Microsoft Silverlight і Sun JavaFX, хоча досягнення цієї мети займе багато років.

Специфікація HTML 5 зводиться до надання семантичного рівня мови розмітки і пов'язаних з ними семантичних рівнів API для сценаріїв задля авторизації доступних сторінок у Всесвітній мережі інтернет, від статичних документів та аж до динамічних застосунків. HTML 5 дозволяє використовувати багато нових елементів і атрибутів, які відображають типову архітектуру сучасних веб-сторінок. Деякі з них є семантичними замінами загально-використовуваних блочних (div) і вбудованих (span) елементів, наприклад елемент content(раніше потрібно було для цього використовувати div) і header . Інші елементи забезпечують нові функціональні можливості через стандартизований інтерфейс, наприклад елементи audio і video.

Наразі специфікація має статус «у розробці», та, як очікується, матиме його ще протягом трьох років, хоча розробка частин HTML 5 буде завершена і реалізована в браузерах ще до того, як специфікація отримає остаточний статус.

Cascading Style Sheets(CSS) – каскадна таблиця стилів. Мова, яка використовується для візуалізації презентації сторінки написаних на HTML.

Один і той самий HTML документ може бути по іншому відображений залежно від використаного CSS. Стили для відображення сторінки можуть бути:

Стили автора (інформацію надає автор сторінки):

- зовнішні таблиці стилів , окремих файл або файли .css
- внутрішні таблиці стилів, включені як частина документа або блоку
- стилі для окремого елемента.

Стили користувача: локальний css-файл, вказаний користувачем для використання на сторінках і вказаний в налаштуваннях браузера (наприклад Google Chrome, Mozilla Firefox).

Великою перевагою каскадної таблиці стилів є те, що інформація про стиль може знаходитись в одному файлі.

Отже, нами обгрунтовано вибір технологій для розробки веб-додатку сайту автосалону. Найкраще для реалізації даного проекту підходить мова розміти HTML 5, для стильового оформлення веб-додатку будемо використовувати CSS3. Середою розробки буде програма Web Storm, дизайн макету буде розроблений у програмі Adobe Photoshop.

ЛІТЕРАТУРА

1. W3SCHOOLS [Електронний ресурс] – Режим доступу до ресурсу: <https://www.w3schools.com/JavaScript>:
2. Дуглас Крокфорд. Java Script The Good Parts 2016. – 264 с.
3. Front-end Developer [Електронне джерело] – Режим доступу до ресурсу: <https://developer.frontend.com/library/archive/documentation/General/Conceptual/DevPedia-CocoaCore/MVC.html>

Бондаренко Р. Г.
Криворізький національний університет
Музика І. О.
к. т. н, доцент, Криворізький національний університет

ІНФОРМАЦІЙНА СИСТЕМА Е-СТУДЕНТ

Проаналізовано стан інформаційних систем університету в сфері комунікації та дистанційного контролю студентів. Виявлено аналоги, найважливіші компоненти системи, дано алгоритмічну модель системи, з урахуванням недоліків аналогів та потреб, що необхідно задовольнити.

Облік відвідування пар студентами – важлива статистична інформація, але зараз він проводиться вручну педагогом, а інколи й студентами самостійно. Витрати часу, неповна визначеність та деякі інші проблеми змушують впроваджувати технологічні рішення для задоволення інформаційних потреб в достатніх обсягах.

Станом на 2020 рік в Україні не існує систем, що задовольняли б високі стандарти та потреби університету в задачах цієї галузі.

Великий приватний банк пропонував систему контролю відвідування учнями школи, а зараз пропонує систему контролю співробітників [1], що була найближчим реалізованим аналогом в Україні, але вона потребувала специфічних карткових продуктів, встановлення турнікетів та додаткового обладнання, не була захищена від “шахрайства” з підміною карток.

Система контролю відвідування пар, яку запропоновано сьогодні не тільки позбавлена таких суттєвих недоліків, але й відповідає багатьом іншим, не менш важливим викликам - безпека, простота, доступність, дешевизна впровадження.

Прогресивний технологічний стек – Golang, VueJS (+ PWA) та широкоеживане, перевірене досвідом обладнання (ESPXX) дозволяє сконцентруватися на важливих реалізаційних викликах та не відволікатися на непередбачувані збої, як на етапі розробки, так і на етапі експлуатації.

На рис. 1 розглянуто процес реєстрації відвідування заняття за допомогою інтеграційної схеми базових компонентів системи.

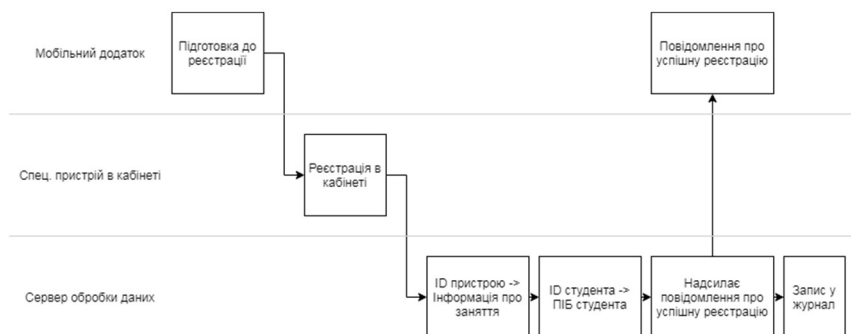


Рис. 1. Схема процесу реєстрації відвідування заняття

ВИСНОВКИ

На жаль в Україні розвиток технологічної складової освітньої системи проходить достатньо повільно, існуючі системи обліку відвідування пар застарілі, не працюють, або ж мають істотні недоліки. Тому розробка такої системи, що врахувала б усі проблеми та досвід європейських аналогів є не просто доцільною, а такою, що мала бути розроблена набагато раніше.

ЛІТЕРАТУРА

1. Учет рабочего времени сотрудников [Електронний ресурс] – Режим доступу до ресурсу: <https://privatbank.ua/ru/business/uchet-rabochego-vremeni>.
2. Vue JS [Електронний ресурс] – Режим доступу до ресурсу: <https://vuejs.org/>.
3. The Go Programming Language [Електронний ресурс] – Режим доступу до ресурсу: <https://golang.org/>.

СЕКЦІЯ 4. PROGRAMMING. СИСТЕМНЕ ТА ПРИКЛАДНЕ ПРОГРАМУВАННЯ В КОМП'ЮТЕРНИХ СИСТЕМАХ ТА МЕРЕЖАХ

*Прокоп Ю. В.,
к.і.н., Одеська національна академія зв'язку ім. О.С. Попова
Трофименко О. Г.
к.т.н., доцент, Національний університет «Одеська юридична
академія»*

АНАЛІЗ ПОПУЛЯРНИХ НА ІТ-РИНКУ ПРАЦІ МОВ ПРОГРАМУВАННЯ

Проведено дослідження українського ринку ІТ-праці, що дозволило сформувати певний рейтинг затребуваних мов програмування. Проаналізовано дослідження закордонних ринків ІТ-праці і відповідні рейтинги популярних мов програмування. Зроблено порівняння та виявлено відмінні риси внутрішнього та закордонного ІТ-ринків.

Розвиток сучасних інформаційних технологій стимулює зростання попиту на певні мови програмування і водночас зменшує використання інших. Постійні змінення в рейтингу мов програмування кидають виклик вищим навчальним закладам, які готують ІТ-фахівців, змушуючи їх гнучко адаптувати навчальні програми до потреб ринку праці. Адже ключовим фактором підготовки ІТ-фахівців є затребуваність на ринку праці і відповідність знань та вмінь майбутніх програмістів, набутих під час навчання, до вимог роботодавців.

Проведені нами дослідження [1-3] українського ринку ІТ-праці дозволили сформувати рейтинг популярних мов програмування. Так, перші дві позиції посіли мови JavaScript та PHP, які використовуються у вебпрограмуванні. Розширення можливостей мови JavaScript за останні роки збільшило сферу її використання та значно підвищило її популярність. Третє місце має мова Java, яка використовується, зокрема, для програмування під Android та для серверних застосунків у великих корпораціях. Четверта позиція закріпилась нині за Python, сферами використання якої є веброзробка,

машинне навчання та написання сценаріїв для автоматизації. Мова C#, яка є основним конкурентом Java на практиці, з невеликим відривом посіла п'яту позицію. Наступною за популярністю йде C++, завдяки тому, що саме цією мовою розроблено більшість редакторів і швидкісних серверів. Крім того, C++ зберігає першість у ігровій індустрії. Такі результати були здобуті унаслідок статистичного аналізу результатів пошукових запитів на сайтах rabota.ua та work.ua.

Суттєвою є відмінність між рейтингами мов програмування, актуальними на українському ринку IT-праці, та попитом на мови програмування за кордоном. Дослідження [4] та аналіз вебсайтів zaista.com та dice.com безперечним лідером за затребуваністю за кордоном признали мову Java. Друге місце нині посідає JavaScript. Мови Python, C++ і C# займають з третього по п'яте місця, дещо змінюючись між собою в різних рейтингах [2, 5]. РНР, на відміну від українських показників, посідає сьому та восьму сходинки у закордонних рейтингах.

Відмінні риси внутрішнього і закордонного IT-ринків зумовлені різного роду задачами та різними вимогами у розробленні програмного забезпечення, а отже і різними вимогами до знань мов програмування. Лідерські позиції мов вебпрограмування у нас пояснюються повсюдними становленням вітчизняного інтернет-бізнесу та переведенням на рейки інтернет-комунікацій інфраструктури державних і комунальних підприємств, а отже потребою у розробці великої кількості сайтів різного рівня складності та взаємодії. Для європейського інформаційного суспільства цей етап розвитку дещо позаду, а тому на ринку IT-праці затребуваність у фахівців, які застосовують мови вебпрограмування вже не така велика.

Позаяк український ринок праці має певні відмінності від закордонного, нерозумно повною мірою переймати досвід університетів інших країн при розробленні навчальних програм для підготовки майбутніх програмістів. Проте деякі українські викладачі виступають за безпосереднє залучення досвіду своїх закордонних колег, не усвідомлюючи, зокрема, специфіки українського ринку IT-праці. Позаяк наші вищі мають готувати фахівців саме для розбудови нашої країни, важливо враховувати специфічні регіональні тенденції IT-індустрії. Проте варто регулярно контролювати

змінення попиту на мови програмування в Україні та відповідно до них гнучко коригувати навчальні освітні програми в університетах України.

ЛІТЕРАТУРА

1. Prokop Yu., Trofimenko E., Severin N., Bukata L. An analysis of criteria for choosing a first programming language in universities. *ICT-TERI-2019: The 15th International Conference on ICT in Education, Research, and Industrial Applications*. Kherson State University, Kherson. June 12-15, 2019, pp. 420-425.
2. Prokop Y.V., Trofymenko O.G., Kapustin M.M. A study of software development tools that are required in the job market in Ukraine and the world. *Proceedings of the O.S. Popov ONAT*. 2018, Vol. 2, pp. 101-108. DOI 10.33243/2518-7139-2018-1-2-101-108.
3. Prokop Yu., Trofimenko E., Zadereyko O., Loginova N., Gerganov M. Multivariate analysis when choosing the first programming language studied in universities. *Advancing Society Through Applied Physics, Electrical and Computer Engineering: 2nd Ukraine Conference on Electrical and Computer Engineering (IEEE UKRCON-2019)*. Lviv, July 2-6, 2019, pp. 1224-1228.
4. Top programming languages in use 2000-2018. [Online]. Available: <https://yourbasic.org/top-programming-languages/>.
5. Putano B. Most Popular and Influential Programming Languages of 2018. [Online]. Available: <https://stackify.com/popular-programming-languages-2018/>.

*Нерода С. І.,
Житомирська політехніка
Вакалюк Т. А.,
к.пед.н., доцент, Житомирська політехніка*

БЛОКЧЕЙН-ТЕХНОЛОГІЯ: ХАРАКТЕРИСТИКА ТА СФЕРИ ЗАСТОСУВАННЯ

Коротко описано алгоритм роботи технології блокчейн. Розглянуто області в яких є перспектива застосування даної технології.

Блокчейн – доволі молода технологія, що в більшості випадків асоціюється з криптовалютою. Однак, сфера застосування блокчейн-технології не обмежується лише фінансовим сектором.

Для початку потрібно розібратись, що представляє собою дана технологія. Блокчейн можна порівняти з електронним реєстром відомостей, що присутній у всіх учасників вибраного процесу і дані в ньому постійно оновлюються [8]. В цей реєстр може бути записана будь-яка інформація у вигляді ланцюга блоків: від персональних даних клієнтів до криптовалютних операцій [8]. При цьому оновлення даних у кожному блоці проходить постійно і можна переглянути історію змін. Кожен блок містить посилання на попередній, тому, якщо хтось вирішить втрутитись у цілісність ланцюга, порушити порядок включення або змінить набір даних, то в системі одразу виконається масовий запит інформації у інших учасників системи, який покаже, що нова інформація не відповідає попередній, отже порушено структуру і означає, що такий запит не буде оброблятися. Орієнтовна схема роботи такої технології зображена на рис. 1.



Рис. 1. Спрощена схема роботи алгоритму блокчейн

Оскільки в блоках може зберігатись інформація різного типу, тому технологія блокчейн має досить широкий спектр застосування. Розглянемо де саме можна впроваджувати дану технологію.

Банкові структури. Банкові структури служать для безпечного зберігання та передачі цінних активів. Блокчейн, будучи захищеним цифровим реєстром, може виконувати ці функції. Саме тому в найближчі роки варто очікувати сильного впливу блокчейна на банківську галузь, вважають експерти. На думку деяких учасників ринку, така технологія дозволить банкам заощаджувати близько \$ 20 млрд за рахунок скасування посередників в транзакціях [7].

Платежі та грошові перекази. У відповідності до звіту Світового економічного форуму, децентралізовані платіжні технології можуть створити відкритий, швидкий і безкомісійний потік платежів по всьому світу [7].

Державний сектор. Різноманітність урядових функцій забезпечує широке поле діяльності для блокчейн-технологій. Приклади використання блокчейну в держорганах включають голосування, автоматизовану реєстрацію даних, ведення обліку, транзакції і зві-

рку рахунків, соціальну і гуманітарну допомогу, ринки активів і надання окремим особам контролю над своїми конфіденційними даними і медичними записами.

За словами аналітиків, блокчейн може підвищити ефективність децентралізованих процесів [7]. Наприклад, застосування блокчейн-технологій здатне прискорити збір документації від кількох державних структур, що не мають спільних систем або тісних організаційних відносин. Подібне використання може вирішити проблему розосереджених процесів, таких як видача дозволів і реєстрація.

Застосування в області дотримання нормативних вимог.

Як тільки дані збережено, вони не можуть бути змінені або видалені. Це дозволяє використовувати блокчейн як документальне свідчення або підтвердження передачі цифрових активів. З тим же успіхом він може бути використаний для зберігання даних про власника фактичної власності. За допомогою експериментальної системи на основі блокчейна планується оцифрувати процеси в сфері купівлі-продажу нерухомості. Незмінність дозволяє використовувати блокчейн і як метод підтвердження відповідності процесів нормативним вимогам - запис всіх дій і отриманих результатів в ланцюгу блоків може служити аудиторським журналом для регулюючих органів.

Розумні контракти. Смарт-контракт – комп'ютерний алгоритм, що базується на технології блокчейн, призначений для укладання і підтримки комерційних контрактів [6]. Сторони підписують розумний контракт, для забезпечення автоматизованого виконання зобов'язань договору.

Всі умови контракту повинні мати математичний опис і ясну логіку виконання. У зв'язку з цим, перші розумні контракти мають завдання формалізації найбільш простих взаємин, що складаються з невеликої кількості умов. Розумні контракти, наприклад, можуть стежити за виконанням умов довгострокових кредитів.

Маючи безперешкодний доступ до об'єктів контракту, розумний контракт відстежує за вказаними умовами досягнення або порушення пунктів і приймає самостійні рішення, ґрунтуючись на за-програмованих умовах.

Таким чином, основний принцип розумного контракту полягає в повній автоматизації і достовірності виконання договірних відносин між людьми.

Освіта. Якщо освітні установи можуть перейти на прозору документацію (дипломи, академічні довідки тощо), використовуючи блокчейн або подібні захищені технології, шахраїв в цій сфері стане менше, а також можна буде позбутися від ручних перевірок паперових документів.

Голосування. Зараз іде активна робота над застосуванням побудованої на блокчейні платформи для голосування. Завдяки їй з'явилася б можливість віддавати свої голоси через мобільні пристрої, а підсумки виборів були б чесними.

Лізинг і продаж автомобілів. У 2015 році компанії Visa і DocuSign створили спеціальне програмне забезпечення на базі блокчейна для підвищення зручності оплати оренди автомобіля і здійснення лізингових платежів за нього. Платежі автоматично списуються в міру експлуатації транспортного засобу відповідно до договору. Подібне ж рішення може використовуватися при продажу машин і постановки їх на облік.

Мережеві технології та Інтернет речей. IBM і Samsung працюють над концепцією Adept, яка передбачає використання подібної блокчейну технології для створення децентралізованої мережі з величезної кількості різних пристроїв Інтернету речей, що зможуть взаємодіяти один з одним для автономної установки оновлень, усунення помилок і управління живленням [4].

Торгівля акціями. Overstock.com за допомогою дочірньої фірми TØ побудував систему, що дозволяє компаніям випускати акції та брати кредити під їх заставу через блокчейн [5], а Chain спільно з біржею Nasdaq працює над торгівлею акціями приватних компаній за допомогою блокчейна [5].

Ланцюги поставок. 22 серпня 2017 року IBM анонсувала проект, метою якого є вивчення можливостей використання технології блокчейн для контролю поставок продуктів харчування і підвищення безпеки їжі [2].

Передбачається, що блокчейн, здатний відстежувати роботу сотень учасників масового виробництва і ланцюги поставок продуктів харчування, дозволить швидше визначати джерело зіпсованої їжі, не дозволяючи їй опинитися на прилавках магазинів.

Медицина. В кінці січня 2019 року IBM оголосила про створення блокчейн-екосистеми для охорони здоров'я у співпраці з американськими страховими компаніями Aetna, Anthem і Healthcare Service Corporation (HCSC), а також з великою фінансовою корпорацією PNC Bank [1].

Мета проекту полягає в тому, щоб підвищити прозорість процесів і функціональну сумісність різних технологій, що застосовуються в галузі охорони здоров'я [1]. Блокчейн-технології повинні допомогти розробникам створити безпечну загальну середу для передачі даних. Якщо їх зусилля увінчаються успіхом це, на думку творців, скоротить кількість адміністративних помилок і дозволить більш ефективно обмінюватися інформацією між системами і установами [1].

ВИСНОВКИ

Отже, як показує дослідження, технологія блокчейну має широке застосування у різних галузях, і розробка різних програмних продуктів, що базуватимуться на даній технології, буде актуальним у найближчі роки.

ЛІТЕРАТУРА

1. Aetna, Anthem, Health Care Service Corporation, PNC Bank and IBM announce collaboration to establish blockchain-based ecosystem for the healthcare industry [Електронний ресурс] – Режим доступу до ресурсу: <https://newsroom.ibm.com/2019-01-24-Aetna-Anthem-Health-Care-Service-Corporation-PNC-Bank-and-IBM-announce-collaboration-to-establish-blockchain-based-ecosystem-for-the-healthcare-industry>
2. IBM Food Trust. Новая эпоха для цепочки поставок продуктов питания [Електронний ресурс] – Режим доступу до ресурсу: <https://www.ibm.com/ru-ru/blockchain/solutions/food-trust>
3. Блокчейн для финансовых регуляторов [Електронний ресурс] – Режим доступу до ресурсу: <https://dataart.ua/news/blokchein-dlia-finansovykh-regulatorov/>
4. Криптовалюта под прицелом: биткоин, блокчейн и все, что с этим связано [Електронний ресурс] – Режим доступу до ресурсу: https://www.securitylab.ru/blog/personal/Informacionnaya_bezopasnost_v_detalyah/325954.php

5. Разработка на Blockchain (Блокчейн) [Електронний ресурс] – Режим доступу до ресурсу: <https://zavode.ru/blog/useful/blockchain-blokchejn/>
6. Смарт-контракт [Електронний ресурс] – Режим доступу до ресурсу:
<https://ru.wikipedia.org/wiki/%D0%A1%D0%BC%D0%B0%D1%80%D1%82-%D0%BA%D0%BE%D0%BD%D1%82%D1%80%D0%B0%D0%BA%D1%82>
7. Список крупнейших отраслей, где может использоваться блокчейн [Електронний ресурс] – Режим доступу до ресурсу: <https://mcem.ru/articles/cpisok-krupnejshikh-otraslej-gde-mozhet-ispolzovatsya-blokchejn>
8. Технология блокчейн: что это такое и как используется в Украине [Електронний ресурс] – Режим доступу до ресурсу: <https://newsone.ua/news/tekhnohija-blokchejn-cto-eto-takoe-i-kak-ispolzuetsja-v-ukraine.html>

*Нерода С. І.,
Державний університет «Житомирська політехніка»
Вакалюк Т. А.,
д.пед.н., доцент,
Державний університет «Житомирська політехніка»*

ЗАГРОЗИ І РИЗИКИ ВПРОВАДЖЕННЯ ТЕХНОЛОГІЇ БЛОКЧЕЙН

Розглянуто основні проблеми застосування технології блокчейн та ризики, які виникають при впровадженні даної технології.

Вважається, що транзакції криптовалюти захищені, а схеми шифрування з відкритим ключем майже неможливо зламати. Однак, це не говорить про те, що немає вразливих місць.

Наявність слабких місць може бути обумовлено також платформою блокчейн, яка може виявитися небезпечною в силу використаного середовища розробки або наявності вразливостей в ІТ-архітектурі системи.

16 серпня 2018 року компанія, що спеціалізується на управлінському консалтингу Boston Consulting Group (BCG) представила дослідження [1], в якому розглянуті проблеми впровадження блокчейна в торгову мережу. Незважаючи на те, що компанії і банки з 2016 року активно використовують блокчейн, автори доповіді [1] вважають, що нова технологія не підходить для торгових операцій з фізичними товарами.

Вважається, що блокчейн, спочатку використовувався як платформа для криптовалюти, повинен підвищити прозорість транзакцій і знизити ризик шахрайства. Однак товарообіг із застосуванням блокчейну поки занадто малий, щоб можна було з упевненістю стверджувати, коли його обсяг досягне критичної маси.

Блокчейн, на думку багатьох фінансових інститутів, підвищує продуктивність, робить операції більш прозорими і зводить до мінімуму ризик шахрайства

Однак основна проблема використання блокчейну полягає в тому, що криптовалюта не призначена для фізичних угод - відстежити фізичний об'єкт у віртуальному світі неможливо. Крім того, впровадження нових технологій повинно супроводжуватися узгодженням термінології, і поки неясно, чи виправданий з економічної точки зору перехід на блокчейн-платформу.

Дослідники також відзначають, що неясно, якою мірою продавці готові прийняти блокчейн, здатний знизити і без того незначну частку прибутку. У доповіді BCG [1] підкреслюється, що в міру формування платформ неефективність ціноутворення і нерівномірне поширення інформації, які лежали в основі посередницьких відрахувань, зникнуть, а з ними розчиниться і прибуток продавців.

У той же час, на думку аналітиків, використання блокчейна здатне підвищити прозорість операцій, створити більш ефективний і ліквідний ринок. Торгові угоди перестануть обмежуватися двосторонніми договорами і перейдуть до більш гнучких транзакцій на основі електронних платформ між безліччю продавців і покупців. Чим ширше будуть впроваджуватися нові технології, тим меншою буде роль посередників, і поки неясно, чи захочуть вони активно використовувати блокчейн.

Серйозною перешкодою на шляху широкого впровадження блокчейна часто називають швидкість роботи. За продуктивністю блокчейн значно поступається традиційним базам даних, і на те є

причина: криптографічний компонент, який, власне, і дає блокчейну настільки важливі переваги, має на увазі складні обчислення. Для прикладу, пропускна здатність системи біткоін становить всього лише сім транзакцій в секунду. Ясна річ, цей показник не йде ні в яке порівняння з двома тисячами операцій в секунду, які в середньому проходять через платіжну систему VISA, максимальна продуктивність якої і зовсім досягає 56 тисяч транзакцій в секунду [2].

Звичайно, над підвищенням продуктивності блокчейну активно працюють. Зокрема, для криптоплатформи BitShares заявлена можливість обробляти до 100 тисяч транзакцій в секунду [3], що, звичайно, звучить зовсім непогано. Правда, потрібно врахувати, що способи визначення продуктивності, які використовуються в BitShares, відрізняються від загальноприйнятих. Власне, розподілений характер блокчейна в принципі ускладнює будь-які порівняння, але поки різниця в продуктивності залишається величезною і рахунок не на користь нової технології.

Налагодження та управління інфраструктурою для підтримки блокчейн-рішень - ще одна проблема для організацій, які експериментують з технологією. Команди, які займаються інформаційною безпекою, транзакціями, хмарними обчисленнями і багатьма іншими процесами, розпочинають впровадження блокчейну, позиціонуючи технологію в своїх компаніях як нове рішення для роботи з даними і / або кодом. При цьому згаданий процес може бути дуже руйнівним - зокрема, тому що поки не існує напрацьованих методик його успішної реалізації.

У розмовах про модний тренд бізнесмени найчастіше не враховують головне: блокчейн - це нова технологія роботи з даними, а зовсім не новий товар, який можна перепродати з прибутком. Безумовно, блокчейн-технології допоможуть скоротити витрати при роботі з документами, але спочатку за розробку відповідного продукту потрібно заплатити сторонньому підряднику.

Ще одна причина не поспішати з впровадженням блокчейн-технологій відсутність надійних і ефективних технологічних рішень, доступних всім бажаючим.

Використання блокчейн-технологій на знаходиться поза правовим полем. Це можна проілюструвати на прикладі ICO (Initial Coin Offering). Творці стартапів випускають власні цифрові монети (токени), використовуючи технології блокчейну, і обмінюють їх на

криптовалюту, щоб залучити необхідне фінансування. Правовий статус ICO сьогодні не визначений ні в одній країні світу, а тому у таких інвестицій немає ніяких гарантій повернення.

Розробки принципово нових технологічних рішень ведуться, проте масового їх впровадження не варто очікувати до того моменту, коли технології блокчейн не отримують конкретний юридичний статус.

Якщо ж проявити зайве завзяття вже сьогодні, можна отримати продукт, який не виправдає витрачених на нього сил і засобів, оскільки може бути використаний тільки для внутрішнього вживання в конкретній компанії.

Отже, як показує дане дослідження, технологія блокчейну має слабкі місця пов'язані з продажем реальних товарів, нестачею надійних технічних рішень та відсутністю регулювання на законодавчому рівні.

ЛІТЕРАТУРА

1. A Reality Check for Blockchain in Commodity Trading [Електронний ресурс] – Режим доступу до ресурсу: <https://www.bcg.com/publications/2018/reality-check-blockchain-commodity-trading.aspx>
2. Как работает Lightning Network, будущее Bitcoin [Електронний ресурс] – Режим доступу до ресурсу: <https://hype.ru/@id123/kak-rabotaet-lightning-network-budushchee-bitcoin-m14rbwlv>
3. Технічна інформація BitShares [Електронний ресурс] – Режим доступу до ресурсу: <http://uacoin.club/coin/BTS>

*Саламатов Д. І.,
Державний університет “Житомирська політехніка”
Левківський В.Л.
ст. викладач кафедри комп’ютерних наук,
Державний університет “Житомирська політехніка”*

МОВА ПРОГРАМУВАННЯ PYTHON - ПОТУЖНИЙ ІНСТРУМЕНТ ДЛЯ РОЗРОБКИ ДОДАТКІВ ЗІ ШТУЧНИМ ІНТЕЛЕКТОМ

Обґрунтовано використання мови програмування Python, як інструменту для розробки додатків що містять “Machine learning” та “Artificial Intelligence”.

В сучасному світі технології не стоять на місці. На сьогоднішній день важко уявити хороший та сучасний додаток без використання штучного інтелекту. Все більшої і більшої популярності набирають додатки, які спроможні розпізнавати та виокремлювати різні об’єкти на фото або відео.

Головний напрямок сфери Artificial Intelligence - розробка технологічних рішень, які працюють за принципом людського інтелекту або близько до нього. Тобто, навчаючись, вдосконалюючись, все більше опановуючи звичайним розмовною мовою (адже ряд провідних теорій розвитку каже, що саме мова і праця зробили з мавпи людину). Зараз флагмани AI - це нейронні мережі, машинне навчання, data science [1].

Python - одна з основних мов програмування, які застосовують в області машинного навчання і штучного інтелекту (Machine Learning та Artificial Intelligence). Наприклад, бібліотека з відкритим вихідним кодом TensorFlow, створена дослідницької командою Google Brain, написана з використанням Python. Google використовує цю бібліотеку для програмування і навчання нейронних мереж, які використовуються для вивчення штучного інтелекту [2].

На сьогоднішній день Python є досить перспективною мовою програмування, яка, на мою думку, має велике майбутнє. Вона має кілька потужних і популярних бібліотек, які призначені для роботи з великими даними: аналізу, візуалізації, прогнозування тенденцій.

Один із способів оцінки популярності мови програмування - індекс ТЮВЕ. Він розраховується на основі кількості пошукових запитів в Google та інших пошукових системах. Враховуються запити, що включають назву мов програмування.

Згідно з індексом ТЮВЕ, в серпні 2019 року Python займає третє місце в списку найбільш популярних мов програмування. Він випереджає JavaScript, PHP, Swift та інші популярні мови.

На рис. 1 наведено рейтинг мов програмування станом на серпень 2019 року.

Aug 2019	Aug 2018	Change	Programming Language	Ratings	Change
1	1		Java	16.028%	-0.85%
2	2		C	15.154%	+0.19%
3	4	▲	Python	10.020%	+3.03%
4	3	▼	C++	6.057%	-1.41%
5	6	▲	C#	3.842%	+0.30%
6	5	▼	Visual Basic .NET	3.695%	-1.07%
7	8	▲	JavaScript	2.258%	-0.15%
8	7	▼	PHP	2.075%	-0.85%
9	14	▲	Objective-C	1.690%	+0.33%
10	9	▼	SQL	1.625%	-0.69%

Рис. 1. Рейтинг мов програмування станом на серпень 2019 року

Таким чином Python є відносно новою але дуже перспективною технологією для розробки з використанням AI. Вона має багато вбудованого функціоналу, який дозволяє створювати додатки в цій сфері.

ЛІТЕРАТУРА

1. Революція шоста: штучний інтелект [Електронний ресурс] – Режим доступу до ресурсу: https://project.liga.net/projects/tech_revolutions_artificial_intelligence/.
2. Програмування на Python [Електронний ресурс] – Режим доступу до ресурсу: <https://ru.hexlet.io/blog/posts/programmirovanie-na-python-osobennosti-obucheniya-perspektivy-situatsiya-na-rynke-truda>.

*Кропачов В.І.,
Державний університет «Житомирська політехніка»
Сфремов Ю.М.,
к.т.н., доцент,
Державний університет «Житомирська політехніка»*

ОСОБЛИВОСТІ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ ВЕБ-СЕРВІСУ АВТОМАТИЗАЦІЇ СТО

Проаналізовано застосування та вимоги щодо веб-сервісу, проаналізовано та обґрунтовано вибір використаних веб-технологій, було розібрано особливості реалізації веб-системи.

Веб-система автоматизації СТО реалізована з метою покращення процедури запису в авто сервіс та ведення клієнтського обліку керуючим персоналом станції.

Метою дослідження є висвітлення особливостей програмної реалізації веб-орієнтованої системи автоматизації СТО.

PHP – це широко використовувана мова сценаріїв загального призначення з відкритим вихідним кодом. Говорячи простіше, PHP це мова програмування, спеціально розроблена для написання веб-додатків (сценаріїв), що виконуються на стороні Web-сервера.

Apache – є найбільш широко використовуваним програмним забезпеченням веб-сервера. Розроблений і підтримуваний Apache Software Foundation, Apache є відкритим програмним забезпеченням, доступним безкоштовно. Він працює на 67% всіх веб-серверів у світі. Він швидкий, надійний і безпечний.

MySQL – система керування реляційними базами даних з відкритим вихідним кодом Oracle, що базується на структурованій мові запитів (SQL). MySQL працює практично на всіх платформах, включаючи Linux, UNIX і Windows.

Схема бази даних веб-орієнтованої системи автоматизації СТО представлена на рисунку 1.

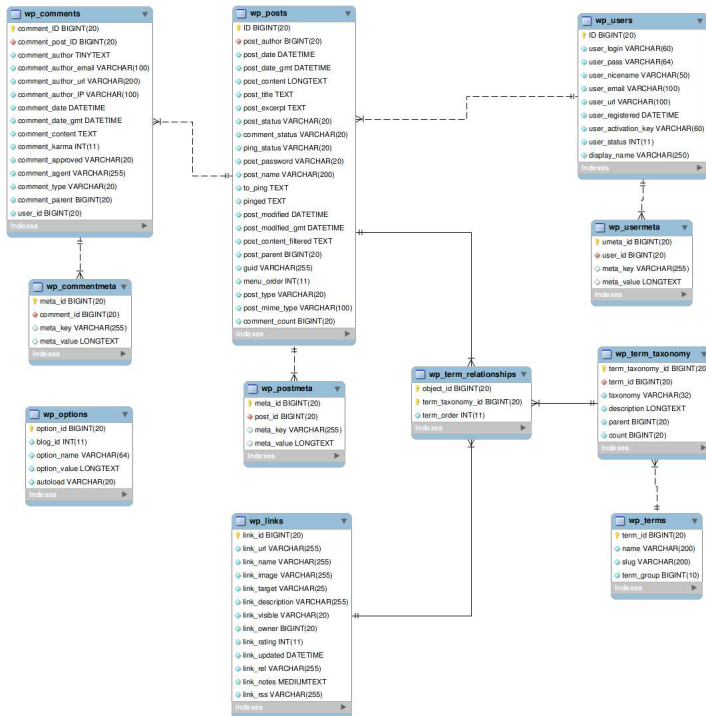


Рис 1. Схема бази даних веб-орієнтованої системи автоматизації СТО

Для розробки була обрана безкоштовна платформа з відкритим кодом CMS WordPress. Написана на мові PHP використовуючи патерн MVC, в якості бази даних використовує MySQL. Сфера застосування - від блогів до досить складних новинних ресурсів. У той же час, вбудована система «тем» і «плагінів» разом з вдалою архітектурою дозволяє конструювати практично будь-які проекти.

CMS(Content Management System) – Система Управління Вмістом дозволяють централізовано керувати усім на сайті – записами, категоріями, файлами та інше. Все виконується з однієї сторінки – панелі адміністратора (або по простому – адмінки). Системи керування, окрім додавання записів, мають ще дуже багато можливостей, які включені в їх стандартну комплектацію, або додаються за допомогою так званих плагінів(додаткових підпрограм), які розширюють функціональність цих систем.

Розроблений веб-сервіс використовуватиме такі ролі користувачів:

1. Гість може:
 - переглядати графік роботи;
 - відправляти заявки на зворотній дзвінок;
 - відправляти заявку на отримання консультації;
 - реєструвати обліковий запис;
 - проходити авторизацію;
 - переглядати сторінку послуг, головну та сторінку контактних даних.
2. Авторизований користувач може:
 - переходити у власний кабінет;
 - записуватись на послугу;
 - переглядати власні замовлення та сервісну історію;
3. Менеджер може:
 - оформлювати замовлення;
 - переглядати заявки;
 - змінювати графік роботи;

Аналіз функціональних вимог дозволив виділити наступні сутності, що забезпечать реалізацію веб-орієнтованої системи автоматизації СТО. На рисунку 2 наведено діаграму класів, які будуть використовуватися.

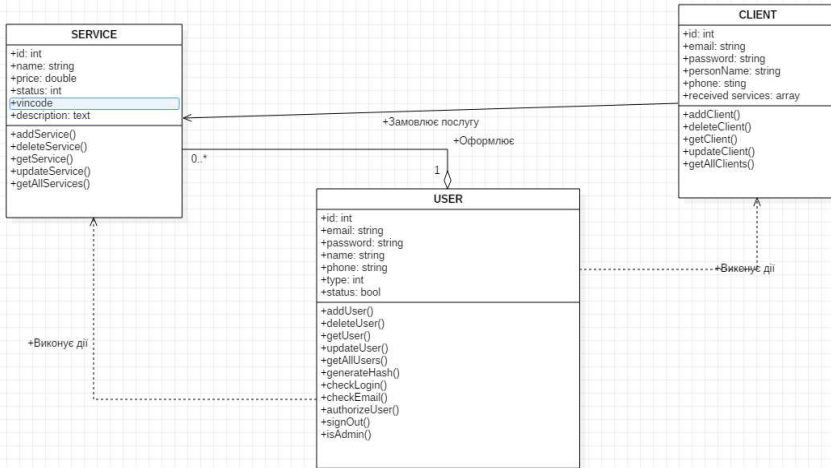


Рис. 2 Діаграма класів веб-орієнтованої системи автоматизації СТО

ВИСНОВКИ

Отже, для створити програмна реалізація веб-орієнтованої системи автоматизації СТО передбачає використання веб-серверу Apache, мови розмітки гіпертекстових документів HTML, каскадних таблиць стилів CSS, мов програмування JS та PHP, системи керування базами даних MySQL, а також застосування такого програмного продукту як CMS WordPress.

ЛІТЕРАТУРА

1. WP Кама – вордпрес как на долони [Електронний ресурс] – Режим доступу до ресурсу: https://wp-kama.ru/function/get_posts.
2. Всё о PHP, MySQL и не только ! [Електронний ресурс] – Режим доступу до ресурсу: <http://www.php.su/>.
3. Jon D. HTML and CSS: Design and Build Websites / Jon Duckett. – NY: John Wiley & Sons, 2011. – 490 с.

Саламатов Д. І.,
Державний університет “Житомирська політехніка”
Левківський В.Л.
ст. викладач кафедри комп’ютерних наук,
Державний університет “Житомирська політехніка”

АКТУАЛЬНІСТЬ РОЗРОБКИ СИСТЕМИ РОЗПІЗНАВАННЯ ВІЛЬНИХ МІСЦЬ ДЛЯ ПАРКУВАННЯ АВТО

Проаналізовано рівень “автомобілізації” в Україні станом на 2019 рік. Доведено доцільність розробки та впровадження системи розпізнавання вільних місць для паркування авто в містах України.

Аналізуючи кількість авто в містах України, можна побачити, що ця кількість зростає з кожним роком. Це породжує ряд проблем для жителів міста. Зокрема, найбільшою проблемою є пошук вільного паркувального місця біля великих супермаркетів, багатоповерхових житлових будинків чи інших місць скупчення людей. Для пошуку вільного паркувального місця зазвичай використовується чимало часу, що нерідко вносить свої корективи в розпорядок дня власника авто[1].

У табл. 1 наведено порівняльну статистику реєстрації легкових автомобілів за осінь 2018 та 2019 років.

Таблиця 1-Статистика реєстрації легкових авто за 2018/2019

Місяць	Кількість реєстрацій 2018	Кількість реєстрацій 2019	Більше на, %
Вересень	6768	7200	6
Жовтень	6750	9000	25
Листопад	7109	8565	17

Впровадження системи розпізнавання вільних місць для паркування автомобілів надасть можливість користувачу легко та швидко визначити, чи має конкретна парковка вільні місця та зорієнтуватися в якій частині парковки ці місця розміщені. Функціональні можливості системи значно полегшать паркувальний процес для жителів міста, які застосовують авто як засіб пересування.

Таким чином, зважаючи на тенденцію росту кількості автомобілів в містах України, було підтверджено доцільність розробки та впровадження системи розпізнавання вільних місць для паркування авто. Ця система забезпечить швидкий пошук паркувальних місць в місцях масового скупчення людей.

ЛІТЕРАТУРА

1. Автоматизована система моніторингу паркомісць [Електронний ресурс] – Режим доступу до ресурсу: <http://www.copa-data.com.ua/realizovannye-proekty/avtomatizovana-sistema-monitoringu-parkomists>

*Самунь С. Й.,
Магістр 1 року навчання
Спеціальності 126 «Інформаційні системи та технології»
Державний університет "Житомирська політехніка"
Вакалюк Т.А.
д.п.н., доц., проф. кафедри
інженерії програмного забезпечення*

СИСТЕМИ ЕЛЕКТРОННИХ ТИРІВ: ПЕРЕВАГИ ТА НЕДОЛІКИ

Проаналізовано наявні технології реалізації систем електронних тирів, їх недоліки та переваги. До основних переваг електронних тирів віднесено доступність, безпека, мала собівартість експлуатації, різноманітність сцен для виконання вправ, також в залежності від ПЗ можливість моделювання оперативної ситуації та поведінки цілей під конкретні задачі.

Можливість точного та швидкого ураження цілей на відстані завжди було і є пріоритетним розвитком людини, яка володіє зброєю.

Для вдосконалення та розвитку навичок володіння вогнепальною зброєю використовують комплекс вправ по стрільбі у спеціально обладнаних тирах або полігонах. Для навчання необхідно дотримуватись суворих заходів безпеки та витратити значні кошти, як на саму зброю так і набої до неї, а також на обслуговування та переобладнання тиру. Для мінімізації цих витрат, зменшення ризиків безпеки, для легкого переобладнання тиру під конкретні задачі при виконанні вправ, а також для впровадження в ігрову індустрію і доступність широкому колу, на сьогоднішній день все більшу популярність набувають електронні тири (лазерні, мультимедійні, інтерактивні) [1].

Сучасні електронні тири дозволяють проводити навчання стрільбі:

- по статичних цілях;
- по цілям, які рухаються, як по заданій траєкторії руху так і хаотично;
- по цілям, які встають;
- по групових цілях;
- по вибіркових цілях.

Електронні тири можуть будуватися на різних технологіях: формування і проведення пострілу; виготовлення зброї, мішені; знаходження, фіксації та обробки зроблених пострілів. А саме:

- зброя з променевим випромінювачем та мішень з світлочутливою матрицею, засоби зв'язку, комп'ютер, ПЗ;
- окуляри доповненої реальності, зброя, що відслідковує рух у 3D-просторі, комп'ютер, ПЗ.
- зброя з відеокамерою, у якості оптичного датчика, МК та бездротовим передавачем, ноутбук або планшет з ПЗ, можливо проектор для транслявання відео зображення з монітору комп'ютера на вертикальну поверхність; [3]
- зброя з променевим випромінювачем, статична відеокамера з світофільтром та направленням на мішень, комп'ютер з ПЗ, яке у реальному часі знаходить та фіксує відмітки пострілів, проектор; [1, 3]

- зброя з променевим випромінювачем, статична відеокамера, комп'ютер з ПЗ, яке за допомогою алгоритмів розпізнавання образів та комп'ютерного зору визначає мішені та фіксує відмітки пострілів у реальному часі [4].

Променевий випромінювач (лазерний або світлодіодний з системою лінз) або оптичний датчик (відеокамера) можуть бути, у вигляді набою, вбудовані у ствол зброї або йти окремим модулем і кріпитися на ньому. Зброя – іграшкова, імітуюча по розміро-ваговим характеристикам, бойова. Мішені – паперові, з світлочутливих елементів, віртуальні [1, 3].

При навчанні стрільбі з вогнепальної зброї на влучність пострілу також впливають наступні фактори:

- сполох від пострілу;
- шум пострілу;
- порохові гази;
- складна віддача та різке відведення зброї з лінії вогню, внаслідок неї;
- вага зброї та магазину з набоями;
- дистанція до мішені;
- різкість спуску куркового механізму.

Врахування цих факторів, в обладнанні електронного тиру значно збільшує ефективність імітування реального пострілу.

Основні переваги електронних тирів це – доступність, безпека, мала собівартість експлуатації, різноманітність сцен для виконання вправ, також в залежності від ПЗ можливість моделювання оперативної ситуації та поведінки цілей під конкретні задачі.

Навчання стрільбі у електронному тирі ніколи не замінить навичок отриманих з використанням реальної вогнепальної зброї, але сучасні електронні тирі завдяки низькій собівартості пострілу, безпеці, різноманітністю сцен для виконання стрілецьких вправ часто-густо використовуються у навчанні, як новачків, так і у спецпідрозділах.

ЛІТЕРАТУРА

1. ООО «НПЦ «Алекс-рубин» [Електронний ресурс] – Режим доступу до ресурсу: <https://alex-rubin.pro/>
2. Домашний тир на Raspberry [Електронний ресурс] – Режим доступу до ресурсу: <https://habr.com/ru/post/248181/>

3. Профессиональные стрелковые тренажеры [Электронный ресурс] – Режим доступа до ресурсу: <https://www.scatt.ru/>
4. Customize your Sirt [Электронный ресурс] – Режим доступа до ресурсу: <https://nextleveltraining.com/>

*Мякишин А. Д.,
Донецкий национальный технический университет
Ярош И. В.,
ст. преп. каф. ПМИ, Донецкий национальный технический университет*

РАЗРАБОТКА RНР-ФРЕЙМВОРКА ДЛЯ СОЗДАНИЯ ВЕБ-ПРИЛОЖЕНИЙ

Описано разработанный RНР-фреймворк для создания веб-приложений, его базовый функционал и архитектуру в сравнении с архитектурой фреймворка, использующего паттерн MVC.

На современном этапе веб-приложения ассоциируются с программными комплексами достаточно сложной структуры, создание и последующая поддержка работоспособности которых – специфическая задача. В связи с этим достаточно актуально и распространено использование специализированных систем для облегчения разработки веб-приложений, так называемых web-фреймворков. Фреймворк представляет собой комплекс уже готовых программных модулей, которые избавляют разработчиков от необходимости создавать новые проекты «с нуля»; организует архитектуру всего проекта и уже содержит в себе базовый функционал, который решает тривиальные задачи.

Разработанный фреймворк состоит из механизмов, которые предоставляют соответствующий базовый функционал. При этом он использует файловую архитектуру вместо паттерна MVC, который популярен в данной области. Основной причиной использования MVC является разделение кода RНР и шаблона страницы, потому что при их сочетании код становится менее читабельным [1]. На практике эти части все-равно смешиваются, но в очень малых количествах, код остается «чистым». Идея данного фреймворка – не

разделять эти части на контроллеры и виды, но выделять весь сложный PHP-код во множество переиспользуемых классов, большинство которых являются уже встроенными во фреймворк, и уже далее использовать их в видах. Использование классов не требует большого количества строк, поэтому код шаблона остается все еще достаточно «чистым» и понятным. Разработчикам не нужно дополнительно создавать еще контроллеры и базовые модели.

В предложенном фреймворке на каждую страницу приходится один файл-вид страницы. Маршрутизация выполняется в соответствии с иерархией видов страниц в определенной директории фреймворка. Например, страница «site.com/company/about» определяется файлом вида страницы с путем «views/pages/company/about.php» (относительно корневой директории проекта).

Механизм видов предоставляет шаблоны страниц и их частей, выдаваемые в ответ на запросы. Типы видов – страницы, блоки, виджеты, страницы ошибок, компоновщики. Также предусмотрены специальные динамические страницы для возможности выстраивания человеко-понятного URL.

Механизм действий, используя паттерн «Шаблонный метод», позволяет удобно реализовывать обработку запросов типа CRUD, которые относятся к созданию, обновлению и удалению, предоставляя средства их пошаговой реализации. Он может определить разработчику методы, которые отвечают за валидацию, действие при успешной/неудачной валидации. Также позволяет определить редирект на другой адрес при завершении действия. При этом фреймворк кодирует все спецсимволы в строках, пришедших из формы от пользователя, избавляя разработчика необходимости помнить об этом. В действия также встроена защита от CSRF (Cross-Site Request Forgery).

Механизм базы данных дает возможность подключаться и работать с ней, выполняя любые SQL-запросы. В качестве СУБД поддерживается MySQL.

Все ошибки и замечания любого уровня, всплывающие на самый верх приложения, преобразуются в классические PHP-исключения, на которые можно ставить обработчики. Механизм обра-

ботки ошибок предоставляет удобный способ подключения пользовательских обработчиков ошибок, возникающих во время работы приложения.

Механизм пользовательских модулей позволяет подключать программные части, специализирующиеся на отдельных частях проекта, для которого используется фреймворк. В нем уже есть такие встроенные модули, как пользователи, статьи, комментарии. К модулям также привязывается механизм прав пользователей. Контроль доступа осуществляется с помощью подхода, схожего с подходом ABAC (Attribute-based access control) [2]. В данной реализации основными атрибутами доступа являются права пользователей, но права можно и расширять, добавляя к ним дополнительные проверки.

Механизм кэширования позволяет кэшировать какие-либо значения, инициализация которых требует ресурсов и/или времени, чтобы каждый следующий раз не выполнять инициализацию этих значений повторно, а выдавать уже созданные ранее значения.

События вызываются при выполнении внутренних процессов фреймворка, чтобы можно было прикреплять обработчики и создавать плагины к нему.

Конфигурация фреймворка хранится в нескольких небольших JSON-файлах и в файле с входной точкой приложения (индексном файле) в виде создания объектов ООП и вызова их конфигурационных методов, что удобно для настройки глобальных частей – подключения модулей и плагинов.

ВЫВОДЫ

Предложенный и описанный PHP-фреймворк позволяет создавать веб-приложения любой сложности и полагает на себя множество базовых операций, связанных с обработкой запросов. Набор функциональных возможностей разработки подлежит развитию и расширению в дальнейшем.

ЛИТЕРАТУРА

1. PHP 7 / Д. В. Котеров, И. В. Симдянов. – СПб.: БХВ-Петербург, 2016. – 1088 с.
2. Guide to Attribute Based Access Control (ABAC) Definition and Considerations [Электронный ресурс] – Режим доступа к ресурсу: <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-162.pdf>.

Адамова В.С.,
Криворізький національний університет
Кузнєцов Д. І.
к.т.н., доцент, Криворізький національний університет

ПОРІВНЯЛЬНИЙ АНАЛІЗ СУЧАСНИХ JAVASCRIPT ФРЕЙМВОРКІВ

Розглянуто сучасні JavaScript фреймворки: React, Angular, Vue.js для створення Web сторінок. Коротко охарактеризовано дані фреймворки та проведено їх порівняльний аналіз.

На сьогоднішній день перед програмістом, котрий починає розробку нового проекту, постає актуальне питання вибору кращого JavaScript фреймворку. Найбільш гостро це питання постає при розробці великих та складних додатків.

Для порівняння було обрано такі JavaScript фреймворки як React, Angular та Vue.js. Всі вони використовуються в основному для створення SPA (Single Page Application). На рисунку 1 наведено статистику завантажень фреймворків за останні 2 роки згідно npm trends [1]. За наведеними даними видно, що на сьогодні найпопулярнішим серед завантажень є React. Друге та третє місце займають Vue.js та Angular відповідно.

Downloads in past 2 Years ▾

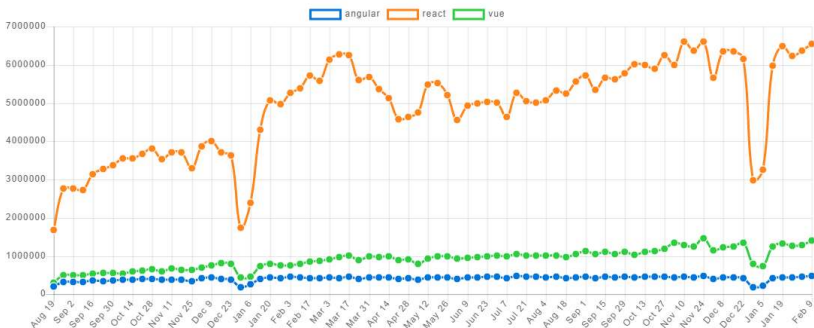


Рисунок 1 – Статистика завантажень фреймворків за останні 2 роки

Порівнюючи дані фреймворки, першим показником з якого слід почати порівняння фреймворку є рендеринг сторінки. Фреймворки, що розглядаються використовують різні підходи для обробки DOM (Document Object Model), що в свою чергу впливає на рендеринг сторінки, відображуваної на екрані користувача. React та Vue.js використовують схожі методи обробки, а саме створюють копію DOM, так званий Virtual DOM, та відображають в кінцевому документі тільки ті частини сторінки, що відрізняються від результатів обробки. Такий підхід скорочує об'єм трафіку, а отже значно прискорює завантаження та рендеринг сторінки, що немало важливо при використанні мобільних пристроїв. У свою чергу Angular вкрай відрізняється від своїх конкурентів. Він використовує підхід розділення «обов'язків», покладаючи на клієнтську частину, тобто браузер користувача, відображення DOM, а на серверну частину створення директив, завантаження коду і сервісів [2 с. 97].

Другий показник на який хочеться звернути увагу це зворотна сумісність, тобто можливість оновлення архітектури для підключення нових модулів і бібліотек. Серед розглядаємих фреймворків повною сумісністю між версіями володіє React та Vue.js. До додатку можна підключити бібліотеки різних версій та оновити застарілі зі успадкуванням властивостей. Angular в свою чергу повністю залежний від попередніх версій і компонентів, тобто прямий перехід між версіями неможливий.

Третім, не менш важливим показником є зручність використання певного фреймворку, а саме наявність відповідної документації та підтримка спільноти програмістів. На рисунку 2 наведено статистику розглядаємих фреймворків на GitHub згідно npm trends [1].

Stats			
	stars 🌟	forks 🍴	issues 🚩
angular	59648	28811	465
react	143737	27604	581
vue	156941	23651	444

Рисунок 2 – Статистика фреймворків на GitHub

З точки зору наявності детальної документації, Vue.js та Angular переважають над React. Але з рисунку 2 видно, що Vue.js та React є лідерами серед спільноти програмістів, а отже розробники в будь-який момент можуть отримати відповіді на запитання.

ВИСНОВКИ

Таким чином, було проведено огляд сучасних JavaScript фреймворків з точки зору різних показників. Кожен фреймворк має свої переваги та недоліки. Оптимальним для використання, враховуючи всі показники є React та Vue.js. Вони більш швидші за рахунок використання Virtual DOM, найбільш доступні для вивчення та розробки завдяки наявності відповідної документації та підтримки з боку інших програмістів, а також є можливість розширення функціоналу та оновлення за рахунок зворотної сумісності.

ЛІТЕРАТУРА

1. Angular vs react vs vue [Електронний ресурс] – Режим доступу до ресурсу: <https://www.npmtrends.com/angular-vs-react-vs-vue>.
2. Матеріали XXIV науково-технічної конференції студентів, магістрантів, аспірантів, молодих вчених та викладачів. Енергетика, електроніка та інформаційні технології. Том III / II ЗНУ. – Запоріжжя: П ЗНУ. - 2019 р.- 128 с.

*Клименко Я. Ю.,
Донецький національний технічний університет
Ярош І. В.,
ст. викл. каф. ПМІ,
Донецький національний технічний університет*

РОЗРОБКА ТА ВПРОВАДЖЕННЯ ПРОГРАМНОГО ЗАСОБУ ДЛЯ КОНТРОЛЮ ДОСЯГНЕНЬ УЧНІВ ЗАГАЛЬНООСВІТНЬОЇ ШКОЛИ

Розкрито проблеми традиційного контролю навчальних результатів учнів за допомогою ведення журналів успішності та запропоновано їх рішення за допомогою використання реалізованого програмного засобу.

На сьогодні широко спостерігається впровадження інформаційних комп'ютерних технологій у всі сфери людської діяльності.

Не оминула ця тенденція і навчально-виховний процес загальноосвітніх шкіл [1].

Необхідно відмітити, що модернізація діяльності навчальних закладів виконується за рахунок застосування сучасного обладнання (комп'ютерів, мультимедійних дошок, широкоформатних моніторів) та спеціального програмного забезпечення, причому як для реалізації адміністративних цілей, так і в межах процесу навчання [2].

Доволі раціональним підходом інформатизації шкільного навчального процесу є використання програмних продуктів для здійснення контролю досягнень учнів. Традиційні методи контролю результатів навчання учнів загальноосвітніх шкіл – паперові класні журнали – вже стають пережитками минулих часів [3]. Такий вид зберігання даних вже не є доцільним і зручним: заповнення журналу виконується вручну, потребується високий рівень уваги, а підрахунок поточного контролю результатів навчання вимагає великих затрат часу.

Вже багато країн перейшли на системи електронних щоденників. В Україні ж на такий спосіб контролю успішності учнів наважилися лише окремі школи. Одними з причин, через які заклади освіти не впроваджують електронні щоденники, є їх дорога вартість, невеликий вибір програмних продуктів на території країни та не завжди достатня укомплектованість програмного продукту. Проаналізувавши існуючі електронні щоденники та потреби шкіл був визначений перелік необхідних для реалізації функцій.

Засобами мови C#, з використанням СУБД MS MySQL, був розроблений програмний продукт для контролю результатів навчання учнів школи.

Основний функціонал запропонованої розробки передбачає:

- систематизацію оцінок школярів:
 - 1) складання звітності щодо успішності учнів за різними підкатегоріями навчання (початкова/середня/старша школа);
 - 2) складання звітності щодо успішності учнів конкретних класів;
 - 3) складання звітності щодо успішності учнів школи в цілому;
- створення інших звітних документів:
 - 1) складання табелів успішності;

- 2) складання звітності для місцевих підрозділів МОН України;
- 3) формування похвальних листів, грамот;
- централізоване зберігання додаткових досягнень учнів у єдиній базі;
 - 1) внесення та зберігання даних про відмінників;
 - 2) внесення та зберігання даних про медалістів;
 - 3) внесення та зберігання даних про участь в олімпіадах, конкурсах;
- генерація індивідуальних вибірок оцінок:
 - 1) формування вибірок по конкретному учню;
 - 2) формування вибірок за навчальними предметами;
 - 3) формування вибірок за класами;
- зберігання інформації в електронному вигляді:
 - 1) внесення даних журналів, що належать архівуванню;
 - 2) внесення поточних оцінок в електронні журнали успішності учнів.

Використання запропонованого програмного засобу для контролю досягнень учнів загальноосвітньої школи вирішить множину проблем, пов'язаних з використанням паперової документації, і дозволить:

- полегшити заповнення журналу: список учнів класу легко дублювати і не виникає потреби повторного його запису;
- уникнути помилок при заповненні полів журналу, а у разі їх виникнення – легко їх виправити;
- автоматизувати підрахунок поточних і підсумкових оцінок учнів;
- зробити зберігання даних компактним та довговічним.

ВИСНОВКИ

У сучасному світі використання інформаційних технологій в освітньому процесі є невід'ємною частиною розвитку та прогресу суспільства. Впровадження програмних засобів для контролю успішності учнів є сучасним і доцільним рішенням багатьох застарілих проблем шкільного навчального процесу.

ЛИТЕРАТУРА

1. Федотова Е. Л. Информационные технологии в науке и образовании : учебное пособие / Е. Л. Федотова, А. А. Федотов. – М. : ИД ФОРУМ; НИЦ ИНФРА-М, 2013. – 336 с.
2. Использование информационных технологий в образовании / Л. В. Красовская, Т. И. Исабекова // Научный результат. Сер. Педагогика и психология образования. – 2017. – Т.3, №4. – С. 29-36.
3. Сулла Р. В., Красовская Л. В. Информационные технологии в школьном образовании // Сборник статей Международной научно-практической конференции. – Белгород, 2017. – С. 521-523.

Бугаенко А.П.

Одесский национальный политехнический университет

Нестерюк А.Г.

к.т.н., доцент, Одесский национальный политехнический университет

СПОСОБЫ ПОВЫШЕНИЯ СКОРОСТИ ОБРАБОТКИ ЗАПРОСОВ БАЗ ДАННЫХ

Проведен обзор и анализ различных способов уменьшения времени, затрачиваемого сервером базы данных для выполнения запросов, показаны их достоинства и недостатки. Рассмотрен метод и алгоритм кеширования запросов как наиболее рациональный.

По мере технического развития наблюдается увеличение значимости интернет-технологий в жизни современного общества. В связи с этим увеличиваются объемы передаваемой и обрабатываемой информации. Что, в свою очередь, выдвигает повышенные требования к мощности серверов либо к нахождению новых эффективных методов, ускоряющих обработку запросов к базам данных. Основными методами, применяемыми при этом являются нормализация [2] и денормализация [3] таблиц, а также кеширование запросов [1, 4, 5]. В докладе проводится обзор современных методов ускорения обработки запросов, приводятся их достоинства и недостатки.

Как наиболее эффективный с точки зрения затраты/производительность предлагается использовать метод кеширования запросов баз данных. Приводится предлагаемый алгоритм кеширования и разбирается его работа. Приводится пример применения данного алгоритма.

ВЫВОДЫ

В результате исследования показана актуальность задачи разработки методов и алгоритмов ускорения обработки запросов баз данных. применение предлагаемого алгоритма кеширования позволяет ускорить выполнение запросов от 20% до 80% в зависимости от размера тестовой таблицы, что позволяет избежать критических нагрузок на сервер.

ЛИТЕРАТУРА

1. Query Optimization in MySQL [Электронный ресурс]. – Режим доступа: <https://www.internet-technologies.ru/articles/optimizaciya-zaprosov-v-mysql.html>
2. Normalizing the pros and cons [Электронный ресурс]. – Режим доступа: <https://studfile.net/preview/895413/page:9/>
3. Denormalization of the database. What for? When? How? [Электронный ресурс]. – Режим доступа: <https://habr.com/ru/post/64524/>
4. Memcached and Grails [Электронный ресурс]. – Режим доступа: <https://www.ibm.com/developerworks/ru/library/j-memcached1/index.html>
5. Data caching [Электронный ресурс]. – Режим доступа: <https://ruhighload.com/search?q=/Кэширование+данных/>

*Добель О. А.,
ГВУЗ «Криворожский национальный университет»
Музыка И. О..*

к.т.н., доцент, ГВУЗ «Криворожский национальный университет»

ИНТЕРАКТИВНАЯ ПОЗНАВАТЕЛЬНАЯ ИГРА НА ПЛАТФОРМЕ UNITY 2D

Проанализировано этапы создания 2D игры. Дано характеристику вспомогательному ПО для разработки будущего приложения, а также сделано сравнение с другими средами разработки.

На сегодняшний день индустрия развлечений является одной из самых масштабных и включает себя множество категорий, в том числе и компьютерные игры.

Заинтересованность людей в данной сфере привела к появлению игр, что основываются на возможности выбора развития сюжета, то есть игр с нелинейным повествованием. Подвидом таких игр являются визуальные новеллы.

Основной платформой для легального распространения подобных игр является Steam. Платформа, на которой любой инди-разработчик может выставить на продажу свой проект.

Для создания визуальных новелл используются как специально спроектированные движки, так и движки общего назначения.

Самым распространенным движком для создания визуальных новелл является Ren-py. Он спроектирован на базе языка Python. Имеет простой интерфейс и ограничен набором команд, которые полностью охватывают нужный функционал для разработки ВН. Основным преимуществом данного движка является простота в использовании. Простые для понимания команды помогают создать визуальную новеллу даже начинающим разработчикам.

Ren-py является межплатформенным движком, так что ВН можно проектировать сразу для нескольких платформ одновременно, в частности для Windows, Linux, Mac OS, iOS и Android.

В тоже время свою популярность в данной индустрии начинает приобретать движок общего назначения Unity. Данная среда разработки имеет более гибкий функционал, что помогает внедрять в разрабатываемую визуальную новеллу элементы из игр других

жанров. Такое разнообразие позволит более интерактивно и интересно подать нужный материал.

Для работы с Unity 2D, в отличие от Ren-ру, понадобятся навыки программирования на языке C#, что значительно усложняет процесс создания ВН для рядового пользователя. Однако, в контексте разработки ВН знакомым с данной средой программистом, процесс создания игры не будет вызывать трудностей.

Unity, так же, как и Ren-ру является межплатформенной средой разработки. Это позволит выпустить будущую игру на различных ОС, ориентировочно это будут Windows, Mac OS, iOS и Android.

Кроме того, Unity имеет графический интерфейс, что позволяет куда быстрее разрабатывать дизайн будущей ВН, а также уже на первых этапах видеть примерный интерфейс игры.

Так же известен тот факт, что наиболее продаваемыми инди-играми на платформе Steam, являются игры, разработанные с помощью движка Unity.

Принимая во внимание все плюсы и минусы рассмотренных выше движков, было принято решение выбрать именно Unity, для создания ВН.

Графическую составляющую будущей ВН будет создано с помощью Adobe Photoshop CC 2019. Этот графический редактор наиболее удобен для реализации всех запланированных нужд.

Так как визуальная новелла базируется на повествовании, а главную роль в передаче информации играет именно текст, встал вопрос о выборе языка будущей игры.

Проведя исследования о статистическом количестве пользователей, исходя из официальной информации предоставленной Steam`ом, было обнаружено, что 38,64% пользователей являются носителями английского языка.

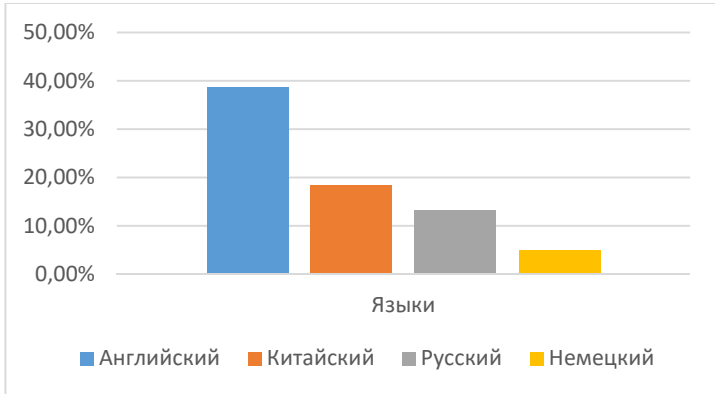


Рисунок 1 – Диаграмма процентного соотношения пользователей Steam

Исходя из этого, было принято решение выбрать английский язык в качестве языка повествования, для большего охвата аудитории.

ВЫВОДЫ

Таким образом, было определено основные этапы разработки игры, а также выбрано сопутствующее этой разработке ПО.

ЛИТЕРАТУРА

1. Официальный сайт Steam [Электронный ресурс] — Режим доступа к ресурсу: <https://store.steampowered.com>

Біляєва К.Є.

Криворізький національний університет

Рибальченко О.Г.

ст. викл., Криворізький національний університет

ВИКОРИСТАННЯ МОВ ПРОГРАМУВАННЯ GOLANG ТА JAVA В ЯКОСТІ СЕРВЕРНИХ МОВ ДЛЯ РОЗРОБКИ WEB-ЗАСТОСУНКІВ

У доповіді були порівняні основні риси мов програмування Go та Java, проаналізовано можливість їх використання як серверних мов програмування.

Java є однією з найпопулярніших мов програмування для серверів та веб-застосунків, її сильні сторони роблять Java майже ідеальною для таких задач. Але у кожній мові є недоліки, що дають поштовхи до постійної модернізації старих мов програмування та створення нових. Таким чином при участі Google Робертом Гризмером, Робом Пайком і Кеном Томпсоном була розроблена нова мова програмування Go (або Golang). Вона була представлена у листопаді 2009 року.

Обидві мови належать до сім'ї мов C, тому мають дещо схожий синтаксис. Ще одна спільна особливість – наявність збирача сміття. Ця риса робить програмування на Java та Go безпечнішим та простішим, адже тепер про цю функцію не треба турбуватися самим програмістам. Розглянуті мови також вважаються крос-платформними, хоча для реалізації цієї особливості Java для інтерпретації коду потрібна наявність Java Virtual Machine (JVM), а Go створює окремий бінарний файл для кожної платформи.

На відміну від Java, Go не є об'єктно-орієнтованою мовою [1]. Тому в Go немає об'єктів та спадкування, наявні тільки структури. Також в Go недоступні перевизначення методів та арифметичні операції над покажчиками. Хоча деякі паттерни об'єктно-орієнтованого програмування можуть бути симульовані, ця особливість впливає на стиль програмування. Таким чином, Go добре підходить для низькорівневого програмування, а от Java не вистачає для цього деяких функцій.

Один з найголовніших принципів Go – «Якщо без чогось можна обійтись, то без цього потрібно обійтись». Мається на увазі, в першу чергу, лаконічність мови – немає нічого зайвого. В Go дуже мало мовних конструкцій (наприклад, тільки один оператор циклу), використовується автоматичне форматування коду на рівні самої мови, в результаті чого існує тільки один канонічний вид коду на Go. Все це реалізовано задля економії часу та простоти розуміння коду іншими програмістами. Завдяки такому підходу програми на Go на 20-30% коротші програм на Java (проста відсутність зайвих слів, крапки з комою в кінці кожної команди, дужок в операторах умови та циклах тощо), а великий проект компілюється за декілька секунд.

Великі переваги Java – масштабованість, багатопотоковість та потужне управління пам'яттю. Мова досить гнучка і при зростанні I/O запитів нескладно додати нові ресурси та перевизначити загрузку. Компоненти Java легкодоступні, тому неважко виконати масштабування великих web-застосунків. При багатопоточних процесах на кожен потік виділяється ізольований стек пам'яті та загальна куча на всі потоки. В кучі зберігаються лише об'єкти, а змінні стека посилаються на них. Сама куча має декілька частин, що полегшує «збирання сміття» в Java.

Важко остаточно визначити, яка з розглянутих мов програмування швидша та продуктивніша, адже швидкість виконання коду буде залежати від типу задач. Якщо вимірювати швидкість виконання еталонних тестів (bin-trees, regex-redux, mandelbrot, fannkuch-redux, n-body), то лідером є Java (Рис. 1).

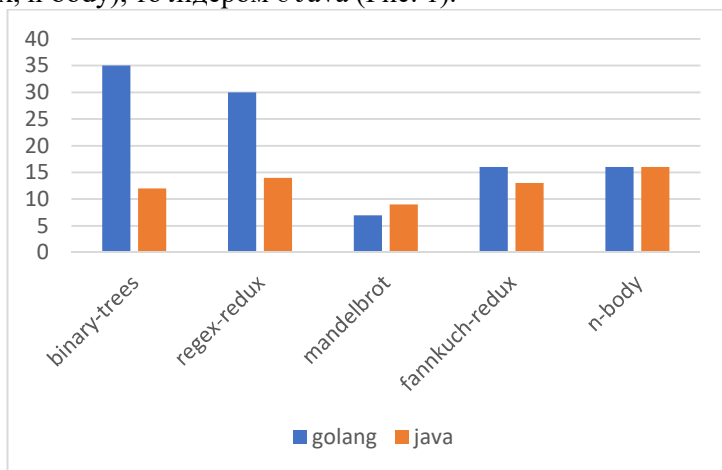


Рис. 1 - Еталонні тести на швидкість виконання

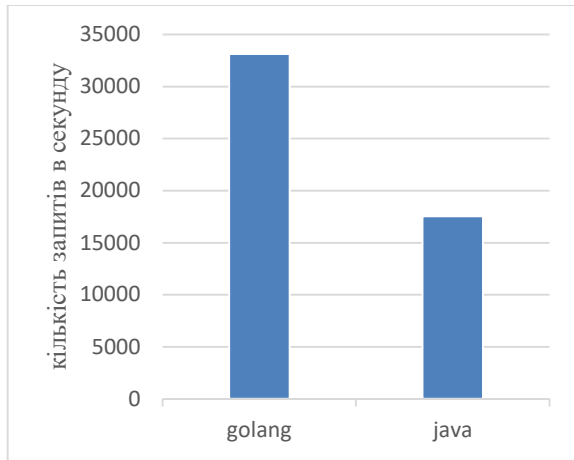


Рис. 2 - Продуктивність http-сервера

Але, якщо мова йде про роботу http-сервера, то Go може обробляти більшу кількість запитів за секунду (Рис. 2) [2].

ВИСНОВКИ

Проведене порівняння показує, що вибір між Go та Java для backend-розробки буде залежати від типу виконуваного проекту, часових вимог та уподобань програмістів. Мова Go виявляється продуктивнішою для обробки запитів до сервера та робить процес програмування легшим завдяки вбудованим інструментам для розробки серверів. Мова Java надає динамічне користування бібліотеками, що може компенсувати відсутність деяких інструментів. Також завдяки існуючим фреймворкам для мови Java, помітна частина роботи може бути автоматизована.

ЛІТЕРАТУРА

1. Educba.com [Електронний ресурс] – Режим доступу до ресурсу: <https://www.educba.com/go-vs-java/>
2. fullstackfeed.com [Електронний ресурс] – Режим доступу до ресурсу: <https://fullstackfeed.com/http-performance-java-jersey-vs-go-vs-node-js/>

Захарова К. І.,
Криворізький національний університет
Кузнєцов Д.І.,
к. т. н., доцент, Криворізький національний університет

МЕТОДИ ТА ЗАСОБИ ПОШУКОВОЇ ОПТИМІЗАЦІЇ САЙТІВ НАВЧАЛЬНИХ УСТАНОВ

Проаналізовано методи аналізу пошукової оптимізації веб-сайтів закладів освіти та приведено показники, сервіси та самі методи веб-аналітики.

Web-сайт - це своєрідний інтерфейс між закладом освіти та його оточенням – студентами та абітурієнтами. Тому створення сайта є одним з головних завдань діяльності навчальних установ, в тому числі і в мережі Internet. [1]

Нині все більша кількість ЗО починає використовувати Internet для обміну інформацією та оптимізації роботи зі студентами. Найпоширенішою формою присутності у мережі є створення сайта (Web-представництва).

Структурована інформація сайта дає змогу студентам і абітурієнтам отримати повне уявлення про заклад освіти і його діяльність.

Такий спосіб передачі повідомлень потенційним студентам ефективніший за звичайні засоби комунікації: пошту, телефон, факс. За допомогою таких обов'язкових елементів, як інтерфейс або навігація, студент/абітурієнт може самостійно вибрати необхідну для вивчення інформацію. Сайт є доповненням до рекламної кампанії, забезпечуючи можливість зворотного зв'язку й інтерактивної роботи з даними ЗО. [1]

Веб-аналітика (з англійської "Web analytics") - це вимірювання, збір, аналіз та впорядкування звітів про веб-дані для розуміння та оптимізації веб-ресурсів.

Веб-аналітика дозволяє зрозуміти, скільки відвідувачів у веб-ресурсу та які сторінки вони переглядають. Ця інформація допомагає оцінити тенденції трафіку та популярність контенту, оптимізувати бюджет. [2]

Веб-аналітика може використовуватись не лише для оцінки та підвищення ефективності веб-сайту, але й як інструмент для дослідження бізнесу та ринку загалом.

Сервіси веб-аналітики дозволяють компаніям виміряти результати традиційних друкованих або широкомовних рекламних кампаній. Це допомагає оцінити, як змінюється трафік на сайт після запуску нової рекламної кампанії.

Методи аналітики бувають такі: [2]

- Аналіз відвідуваності ресурсу.
- Діагностика зручності сайту.
- Вивчення поведінки користувачів.
- Бенчмаркінг (еталонне тестування, порівняння з тенденціями і з конкурентами).
- Наскрізна аналітика (метод ROMI, на основі різних даних простежується повний шлях відвідувача, починаючи від перегляду реклами до покупки).

Показники аналітики: [2]

- Глибина перегляду веб-сторінок – це кількість переглянутих веб-сторінок одним користувачем за один раз.
- Час, проведений на ресурсі – береться до уваги час, коли відвідувач здійснював якісь дії на сайті, переходив на різні сторінки.
- Відмови – аналізується те, як швидко користувач покинув сайт (відмовився від подальшого перегляду сторінок).
- Конверсія – співвідношення загальної кількості візитів до кількості відвідувань, під час яких користувачем було здійснено певну дію (покупка, дзвінок, заповнення лід-форми).
- Досягнення цілей – це певні дії користувачів ресурсу, що цікавлять бізнес загалом.

Найпопулярнішим сервісом веб-аналітики є Google Analytics. А також є інші, такі як: [2]

- Majestic;
- Ahrefs;
- Piwik;
- Clicky;

- StatCounter;
- Woopra;
- FireStats;
- CrazyEgg;
- E-commerce;
- Коллтрекінг;
- Hotjar.

ВИСНОВОК

Веб-аналітика – важливий інструмент інтернет-маркетингу, що допомагає відслідковувати ефективність сайтів, оптимізувати бюджет та поліпшити бізнес-показники загалом. Для навчальних установ веб-сайт – один із головних інструментів залучення абітурієнтів.

Здійснити веб-аналітику можна за допомогою різних онлайн-інструментів.

ЛІТЕРАТУРА

1. Навчальні матеріали онлайн. URL: https://pidruchniki.com/1331090747799/informatika/sayt_zasib_pid_trimki_rozvitku_pidpriyemstva (дата звернення 20.02.2020)
2. Digital-агентство Ланет Click. URL: <https://lanet.click/services/web-analytics/> (дата звернення 20.02.2020)

*Краснобокий Р. С.,
Криворізький національний університет
Музика І. О.,
к.т.н., доцент, Криворізький національний університет*

ОРГАНІЗАЦІЯ РОБОТИ СТУДЕНТА НА БАЗІ DISCORD API

Проведено аналіз популярності чат додатків Skype, Telegram, Discord та обрано платформу для власної розробки організатора.

Сьогодення студента неможливо уявити без інформаційних технологій. Однак, проблема вибору відповідного програмного забезпечення для оптимізації роботи студента все ще вирішена не до

кінця, оскільки широко ІТ технології стали впроваджувати в освіту відносно недавно. Розглянемо варіант власної реалізації програми-помічника.

Оскільки розробка власного додатка з початку може виявитися трудомісткою і ресурсомісткою задачею, то рішенням може стати використання за основу інших додатків, які надають власне API для розробки додатків на їх базі.

API (Application programming interface) в свою чергу це набір процедур і функцій, що дають можливості, дані і засоби для створення додатків на основі іншої програми [2]. Потенційними для розробки можуть бути такі популярні платформи: Telegram, Discord, Skype. Об'єднує ці платформи те, що це чат додатки із підтримкою голосового зв'язку (VoIP).

Skype – у першу чергу проектувався саме для відеодзвінків та конференцій, де функція чату є додатковою. На даний момент додаток широко використовують у сферах бізнесу за його простий та інтуїтивно сприйнятливий інтерфейс.

Telegram – навпроти, створювався як текстовий чат, але із підтримкою дзвінків. Високу популярність здобув завдяки безпеці переписки, яка забезпечується алгоритмом end-to-end шифрування MTProto 2.0.

Discord – додаток, який дозволяє створювати сервери, де можна додавати як текстові, так і голосові канали. Значної популярності здобув серед людей, які грають в комп'ютерні ігри і молоді у цілому за сучасний інтерфейс і великий обсяг функціоналу.

На рисунку 1 за даними Statista [1] на липень 2019 зображено топ-8 найпопулярніших мобільних месенджерів у світі.

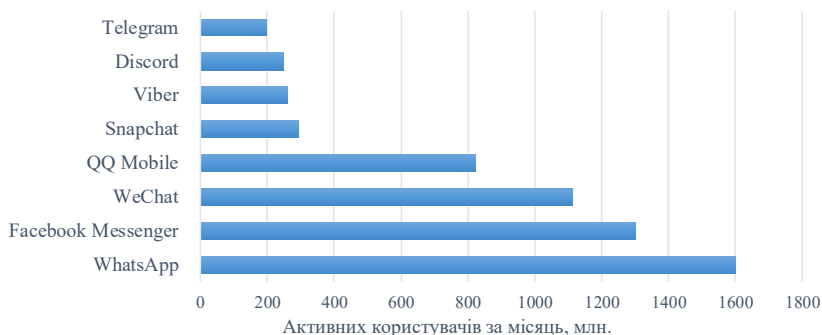


Рис. 1. Популярність месенджерів у світі

ВИСНОВКИ

Таким чином, для розробки було обрано платформу Discord, тому що крім функціоналу, у цієї платформи дуже зручно побудована робота із API. Discord-бот – додаток, який може бути створено на будь-якій мові програмування, однак приєднання до платформи виконується за рахунок ключа авторизації OAuth 2. Функціонал бота є досить різноманітним та реалізується через команди.

ЛІТЕРАТУРА

1. Мессенджер для геймерів Discord обошёл Skype и занял четвертое место по аудитории на ПК в России в апреле 2019 года [Електронний ресурс] – Режим доступу до ресурсу: <https://vc.ru/services/73312-messenzher-dlya-geymerov-discord-oboshel-skype-i-zanyal-chetvertoe-mesto-po-auditorii-na-pk-v-rossii-v-aprele-2019-goda>
2. Discord Developer Portal [Електронний ресурс] – Режим доступу до ресурсу: <https://discordapp.com/developers/docs>

Кузьменко А. І.,

Криворізький національний університет

Музика І. О.

к.т.н., доцент, Криворізький національний університет

ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ ОРГАНІЗАЦІЇ КОРПОРАТИВНОЇ РОБОТИ НА ОСНОВІ TELEGRAM API

Проаналізована доцільність розробки інформаційної системи в месенджері за допомогою ботів. Наведена порівняльна характеристика Telegram, Viber та WhatsApp.

Кожному доводилося стикатися з недоліками організації корпоративної роботи, тому вирішення цього питання є важливим та актуальним. Для цього потрібен новітній підхід, тому доцільно розробити бота у месенджері Telegram [1].

Бот – це обліковий запис, котрий є інтерфейсом для програми, яка працює на сервері. За допомогою HTTPS-запитів до API Telegram здійснюється керування ботом. Використовуючи API Telegram можна отримувати індивідуальні повідомлення та новини, виконувати інтеграцію з іншими сервісами, отримувати платежі від

користувачів Telegram, створювати користувальницькі інструменти, будувати соціальні сервіси тощо.

У таблиці 1 наведена порівняльна характеристика месенджерів, а саме Telegram, Viber та WhatsApp [2].

Таблиця 1 – Порівняльна характеристика месенджерів

Характеристика	Telegram	Viber	WhatsApp
Бізнес модель	некомерційний	комерційний	комерційний
Політика конфіденційності	чітка і прозора	середньої складності	досить складна
Документація	відкрита і докладна	відкрита і докладна	відсутня
Протокол шифрування	відкритий MTProto	напівзакритий протокол	відкритий Signal
Кількість пристроїв	необмежена	один первинний і кілька вторинних пристроїв	2
Швидкість відправлення повідомлень	150 мс	900 мс	550 мс

ВИСНОВКИ

Таким чином, налаштування інформаційної системи у середовищі Telegram буде зручним для кожного споживача системи, надійним та нескладним у розробці.

ЛІТЕРАТУРА

1. Офіційний сайт Telegram [Електронний ресурс] – Режим доступу до ресурсу: <https://core.telegram.org/api>.
2. Порівняння месенджерів [Електронний ресурс] – Режим доступу до ресурсу: <https://jayxt.github.io/MessengerComparison/ru/>

*Турчик Є. Л.
Криворізький національний університет
Рибальченко О. Г.
старший викладач, Криворізький національний університет*

ВИКОРИСТАННЯ ХМАРНИХ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ ДЛЯ РОЗРОБКИ WEB-ЗАСТОСУНКІВ

У доповіді були порівняні основні аспекти хмарних обчислювальних систем Amazon elastic compute cloud, Microsoft Azure та Google compute engine, наведені приклади їх використання.

За останні 10 років стрімкий зріст попиту на створення web-застосунків спонукав масштабний розвиток технологій розробки відповідних додатків та їх введення в роботу. Оскільки купівля та підтримка власних серверів із можливістю розширення обчислювальної потужності для більшості стартапів є нерентабельною, популярності набули хмарні обчислювальні системи.

Amazon Elastic Compute Cloud (Amazon EC2) – це веб-сервіс, що надає безпечні хмарні обчислювальні ресурси із можливістю подальшого масштабування [1]. Він допомагає розробникам програмного забезпечення, спрощуючи проведення обчислень в хмарі в масштабі всього Інтернету. Amazon EC2 дозволяє отримати доступ до обчислювальних ресурсів і їх налаштувань з мінімальними зусиллями та надає користувачам повний контроль над потужностями, а також перевірене обчислювальне середовище Amazon для роботи. Цей сервіс був застосований при розробці таких відомих проєктів як Airbnb, Netflix, Allergan та ін..

Microsoft Azure – це набір хмарних служб, що постійно розширюється та допомагає організаціям різної потужності вирішувати свої бізнес-завдання [2]. Він дозволяє вільно створювати та розгортати застосунки, управляти ними у глобальній мережі з можливістю використання великої кількості зручних інструментів і спеціалізованих платформ. Цей сервіс широко використовується такими корпораціями як Adobe, Hewlett Packard, BMW та ін..

Google Compute Engine надає користувачам віртуальні машини із великою кількістю налаштувань та можливістю масштабу-

вання [3]. Вони працюють у центрах обробки даних Google та становлять частину високоефективної мережевої інфраструктури Google. Сервіс надає можливість інтегрувати обчислення з іншими службами Google Cloud, такими як штучний інтелект, машинне навчання та аналітика даних. До його найвідоміших клієнтів можна віднести PayPal, Twitter, Airbus та AirAsia.

Після проведення порівняльної характеристики хмарних обчислювальних систем за критеріями популярності, потужності та ціни можна виділити такі переваги розглянутих сервісів [5]:

- Amazon Elastic Compute Cloud – це найбільш використовувана платформа, популярність якої підтримується довірою до бренду Amazon;
- Microsoft Azure – має перевагу в потужності, оскільки надає для використання реальні ядра процесорів, а не віртуальні vCPU, як у Google та Amazon;
- Google Compute Engine – має відмінну систему дисконтування, яка включається автоматично та дозволяє економити більше коштів, ніж при використанні сервісів Amazon або Microsoft.

Масштаби сучасного використання хмарних сервісів наведено на Рис.1 [4].



Рис.1. Використання хмарних сервісів у світі станом на 2018 рік

ВИСНОВКИ

Після порівняння хмарних обчислювальних систем Amazon elastic compute cloud, Microsoft Azure та Google compute engine, можна зробити висновок, що кожен із перелічених сервісів дозволяє розробникам зручно та ефективно створювати web-застосунки, вести контроль над ними та проводити модифікації із можливістю розширення обчислювальної потужності.

ЛІТЕРАТУРА

1. Amazon EC2 [Електронний ресурс] / Режим доступу: <https://aws.amazon.com/ru/ec2/>
2. Microsoft Azure [Електронний ресурс] / Режим доступу: <https://azure.microsoft.com/>
3. Google Compute Engine [Електронний ресурс] / Режим доступу: <https://cloud.google.com/compute/>
4. Roundup of cloud computing forecasts and market estimates 2018 [Електронний ресурс] / Режим доступу: <https://www.forbes.com/sites/louiscolombus/2018/09/23/roundup-of-cloud-computing-forecasts-and-market-estimates-2018/>
5. Тестируем облачные платформы из Топ-3 [Електронний ресурс] / Режим доступу: <https://habr.com/ru/post/328916/>

Барановський В.

*Криворізький економічний інститут Київського національного
університету ім. Вадима Гетьмана*

Вдовиченко І.Н.

К.т.н., доцент, Криворізький національний університет

РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СОЦІАЛЬНОЇ МЕРЕЖІ З ВИКОРИСТАННЯМ NODE.JS І REACT NATIVE

Метою дослідження є створення програмного забезпечення соціальної мережі з використанням Node.js і React Native. Задачами

дослідження є створення програмного забезпечення соціальної мережі. Об'єктом дослідження є соціальна мережа. Предметом дослідження є алгоритм створення контролю соціальної мережі. Результатами дослідження є програмний комплекс, розроблений на мові JavaScript з використанням технологій Node.js та React Native.

З розвитком технологій і появою мобільних телефонів, Інтернету вже мало хто може уявити своє життя без можливості перебувати в онлайн-режимі. Саме потреби користувачів і диктують свої правила в створенні нових софтів. Якщо раніше перші програмні забезпечення для мобільних телефонів - це були вбудовані додатки, необхідні для виконання певних функцій самого пристрою, наприклад, телефонна книга, можливість писати, редагувати смс-повідомлення, то зараз, з появою нових гаджетів, все більшою популярністю користується розробка мобільних додатків для Android і iOS (iPhone). Поява мобільного Інтернету неймовірно полегшила життя багатьом користувачам. Як правило, звичними супутниками подорожей, поїздок стали телефони, планшети та інші пристрої, через які можна просто підключитися до мережі.

Соціальні мережі – це соціальна структура, що утворена індивідами або організаціями. У соціальних мережах існують можливості вподобати або поширити опубліковану новину. Кількість вподобань дозволяють оцінити актуальність новини серед користувачів.

Метою розробки власної соціальної мережі було невдоволення незручністю і зайвим функціоналом існуючих на той момент подібних сервісів.

Сьогодні Node по праву вважається однією з лідируючих платформ для веб-розробки, і велика частина веб-інструментів, серверних і клієнтських, працюють з цим інструментом. Один з найважливіших його плюсів - це асинхронність в поєднанні з подієвим підходом. На відміну від потокового, подієво-орієнтоване програмування засноване на якихось зовнішніх діях. Таке рішення спрощує програмування інтерактивних додатків, тобто заснованих на роботі з використанням введення-виведення [1].

React Native - це JS-фреймворк для створення відображаємих iOS-і Android-додатків. В його основі лежить розроблена в Facebook

JS-бібліотека React, призначена для створення користувацьких інтерфейсів. Але замість браузерів вона орієнтована на мобільні платформи. Іншими словами, якщо ви веб-розробник, то можете використовувати React Native для написання чистих, швидких мобільних додатків, не залишаючи комфорту звичного фреймворка і єдиної кодової бази JavaScript. У порівнянні зі стандартною розробкою під iOS і Android, React Native має набагато більше переваг. Оскільки додаток здебільшого складається з JavaScript, ви можете користуватися численними перевагами веб-розробки. Наприклад, щоб побачити внесені в код зміни, можна миттєво «оновити» додаток замість тривалого очікування завершення традиційного ребілд. Крім того, React Native надає «розумну» систему повідомлень про помилки і стандартні інструменти налагодження JavaScript, що сильно полегшує процес розробки [2].

MongoDB - документо-орієнтована система керування базами даних (СКБД) з відкритим вихідним кодом, яка не потребує опису схеми таблиць. MongoDB займає нішу між швидкими і масштабованими системами, що оперують даними у форматі ключ/значення, і реляційними СКБД, функціональними і зручними у формуванні запитів. MongoDB підтримує зберігання документів в JSON-подібному форматі, має досить гнучку мову для формування запитів, може створювати індекси для різних збережених атрибутів, ефективно забезпечує зберігання великих бінарних об'єктів, підтримує журналювання операцій зі зміни і додавання даних в БД, може працювати відповідно до парадигми Map/Reduce, підтримує реплікацію і побудову відмовостійких конфігурацій. У MongoDB є вбудовані засоби із забезпечення шардінгу (розподіл набору даних по серверах на основі певного ключа), комбінуючи який з реплікацією даних можна побудувати горизонтально масштабований кластер зберігання, в якому відсутня єдина точка відмови (збій будь-якого вузла не позначається на роботі БД), підтримується автоматичне відновлення після збою і перенесення навантаження з вузла, який вийшов з ладу. Розширення кластера або перетворення одного сервера на кластер проводиться без зупинки роботи БД простим додаванням нових машин.

ВИСНОВКИ

Для реалізації задачі було створено базу даних, розробленою на MongoDB, призначену для зберігання вхідної інформації.

Відповідно до розробленої моделі, був визначений алгоритм побудови програми. Для реалізації алгоритму задачі було обрано мову програмування JS, з використанням фреймворку React Native.

Результатом роботи стала розробка мобільного додатку соціальної мережі, яка вирішує поставлену задачу. Програмний комплекс, розроблений на мові JavaScript з використанням технології React Native, яка призначена для створення користувацьких інтерфейсів. Але замість браузерів вона орієнтована на мобільні платформи.

ЛІТЕРАТУРА

1. MongoDB Documentation [Електронний ресурс] - Режим доступу: <https://docs.mongodb.com/>
2. React Native - A framework for building native apps using React [Електронний ресурс] - <https://facebook.github.io/react-native/>

Вінівітін І. В.,

Криворізький національний університет

Музика І. О.

к.т.н., доцент, Криворізький національний університет

РОЗРОБКА МЕСЕНДЖЕРУ З ВИКОРИСТАННЯМ СУЧАСНИХ ВЕБ-ТЕХНОЛОГІЙ

Проаналізовано перспективи застосування ReactJS, WebRTC, Electron для створення месенджера.

Проведений аналіз показав, що на сьогодні існує велика кількість різних месенджерів та програм для обміну інформацією. Для розробників є чимало технологій, бібліотек та фреймворків для організації миттєвої передачі інформації. Одна з таких технологій WebRTC (Web Real-Time Communication) – це інтернет-протокол із відкритим кодом, призначений для організації голосового та відеозв'язку через інтернет у режимі реального часу [1].

Впровадження однорангового транспорту з низькою затримкою є нетривіальним інженерним завданням: потрібно налаштувати обхід таблиць NAT (Network Address Translation), перевірити підключення, забезпечити сигналізацію, безпеку, контроль перевантаження і безліч інших деталей, про які необхідно подбати. WebRTC

обробляє все вищеперераховане і багато іншого від нашого імені, тому це одне з найбільш значних доповнень до веб-платформи з моменту її створення. Фактично, WebRTC – це не просто набір окремих функцій, а злагоджена інформаційна система, всі компоненти якої працюють разом, щоб надати простий і уніфікований API для побудови додатків у браузері.

Надання багатофункціональних високоякісних додатків RTC, таких як аудіо і відеоконференції, обмін даними між рівноправними вузлами, вимагає багато нових функцій в браузері: можливості обробки аудіо та відео, нових API-інтерфейсів додатків і підтримки півдюжини нових мережевих протоколів. Браузер абстрагує більшу частину цієї складності від трьох основних API:

- `MediaStream`: отримання аудіо та відео потоків;
- `RTCPeerConnection`: передача аудіо і відео даних;
- `RTCDataChannel`: передача довільних даних додатка.

Для створення додатку необхідна незначна кількість програмного коду JavaScript, і будь-який веб-додаток може забезпечити багатий досвід проведення телеконференцій за допомогою передачі даних. На рис. 1 наведено схему узгодження сеансу між користувачами.



Рис. 1. Сигналізація і узгодження сеансу

В основі роботи алгоритму узгодження сеансу лежить одно-рангове рукостискання (рис. 2).

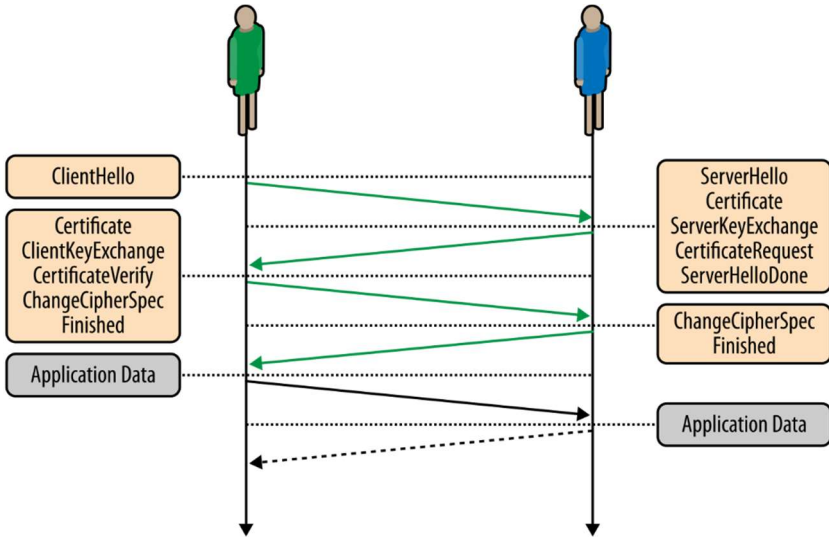


Рис. 2. Однорангове рукоштовання по DTLS

ВИСНОВКИ

WebRTC – сучасна та відносно нова технологія для обміну даними між браузерами. Вона забезпечує безпечну та швидку передачу даних, має зрозумілий API. Із впровадженням протоколів на базі IPv6 технологія WebRTC зможе забезпечити високий рівень якості відео та аудіо трансляції.

ЛІТЕРАТУРА

1. Офіційний сайт WebRTC [Електронний ресурс] – Режим доступу до ресурсу: <https://webrtc.org>.

Лісневський Д. Є.,
Криворізький національний університет
Музика І. О.,
к.т.н., доцент, Криворізький національний університет

РОЗВАЖАЛЬНА ОНЛАЙН ПЛАТФОРМА З ЗАСТОСУВАННЯМ ТЕХНОЛОГІЇ NODE.JS

Проаналізовано перспективи застосування засобів для створення веб-додатків, таких як JavaScript та Flash. Описано технологію Node.js та принцип роботи деяких ігор, які входять до складу створюваної розважальної онлайн платформи.

Браузерні ігри хоч і не такі круті, як ігри для ПК та консолей, проте вони не гірше приковують до себе увагу гравця. Причин цього доволі багато: вправна робота гейм-дизайнерів, котрі створюють «дозований» ігровий процес, відсутність необхідності завантажувати та встановлювати гру на пристрій, можливість грати будь-де, будь-коли, та з будь-якого пристрою, котрий має веб-браузер та підключення до мережі Інтернет.

Багато браузерних ігор було створено на Flash, який працює на скриптовій мові програмування ActionScript. Flash – це програмна платформа для створення анімацій, ігор та різних інтернет-застосунків. Проте наразі популярність цієї платформи сильно знижується з декількох причин:

- припинення підтримки та розповсюдження платформи у 2020 році;
- проблеми з безпечністю продукту та недовіра користувачів до платформи Flash [1];
- наявність на ринку платформ та мов програмування, які забезпечують більш гнучкий функціонал, вищу продуктивність та мають меншу кількість вразливостей.

Тому наразі набирає популярності створення веб-ігор за допомогою мови JavaScript, котра дає можливість зберігати інформацію про гравця в змінних та обробляти графіку за допомогою елемента HTML5 Canvas. Проте без бази даних та розробки серверної частини така гра не матиме можливості постійно зберігати прогрес

гравця, а продовжити грати з будь-якого збереження на іншому пристрої без спеціальної синхронізації буде неможливо. Також буде відсутня можливість грати через мережу з кимось, що одразу зменшить популярність такої гри. Тому розробники використовують серверні мови програмування для вирішення цих проблем.

Враховуючи те, що основною грою створюваної платформи буде клікер, потрібно використовувати таку технологію, котра буде пристосована для створення високопродуктивних мережевих застосунків, щоб максимальна кількість клієнтів могли синхронізувати дані з БД за найменший проміжок часу, адже для створення комфортного процесу гри необхідно буде часто зберігати прогрес гравця. Також треба врахувати, що на платформі будуть доступні такі ігри: «Покер», «Аркада» та «Платформер», котрі також потребують частий обмін інформацією з сервером.

Гра, котра найменше потребує високої продуктивності сервера є «Покер», адже для початку гри необхідно мінімум два гравці, для котрих сервер повинен випадковим чином обрати по 2 карти на гравця та 5 на стіл з 52 доступних у колоді, а потім надсилати результат свого вибору клієнтам у відповідний час.

У свою чергу гравці мають можливість робити ставки, мінімальне та максимальне значення яких має контролюватись сервером, та не перевищувати кількість наразі доступних очок, зароблених у грі «Клікер», значення яких сервер повинен розіслати всім клієнтам, котрі знаходяться в виділеній кімнаті (гравцям та спостерігачам).

Отже, сервер повинен обрати від 9 до 23 випадкових карт та розіслати відповідні дані про 7 карт (2 карти гравця та 5 карт на стіл) кожному клієнту з урахуванням виділеного часу на ходи гравців.

Грою, яка дуже вибаглива до продуктивності сервера є «Платформер», адже сервер повинен згенерувати рівень, розставити ворогів та контролювати їх. Також сервер повинен контролювати виконання дій персонажу гравця, щоб унеможливити нечесний вплив на ігровий процес.

Враховуючи вищезазначене, можна зробити наступний висновок: необхідно використовувати таку платформу, котра надасть функціонал для реалізації подієвої системи синхронізації даних з сервером. На цю роль ідеально підійде платформа Node.js [2], яка є

асинхронно подієвим JavaScript оточення, котре дозволяє програмувати клієнтську та серверну частини однією мовою. До того ж ця технологія забезпечує майже на чверть вищу швидкодію в порівнянні з PHP [3].

ВИСНОВКИ

Отже: функціонал мови програмування JavaScript в поєднанні з HTML5 дають можливість створювати клієнтські веб-застосунки будь-якої складності, а при додаванні технології Node.js, котра призначена для створення високопродуктивних мережових додатків та надає можливості використання операцій неблокуючого операційного введення/виведення інформації, можливо створити будь-який клієнт-серверний веб-додаток.

ЛІТЕРАТУРА

1. Про безпечність Flash Player [Електронний ресурс] – Режим доступу до ресурсу: <https://ichip.ru/tekhnologii/pravda-ili-mif-rochemu-flash-player-nebezopasen-225982>.
2. Офіційний сайт Node.js [Електронний ресурс] – Режим доступу до ресурсу: <https://nodejs.org>.
3. PHP vs Node.js [Електронний ресурс] – Режим доступу до ресурсу: <https://habr.com/ru/post/273259/>.

Пузіно М. В.

Криворізький національний університет

Рибальченко О. Г.

старший викладач, Криворізький національний університет

АВТОМАТИЗАЦІЯ РОЗГОРТАННЯ WEB-ДОДАТКІВ З ВИКОРИСТАННЯМ КОНТЕЙНЕРІВ DOCKER

У доповіді були порівняні основні методи розгортання веб-додатків та висвітлені переваги автоматизації цього процесу за допомогою програмного забезпечення Docker.

Розвиток технологій створення веб-додатків, їх супроводу та необхідність автоматизації цих процесів призвели до виникнення низки актуальних питань, пов'язаних з розробкою, доставкою та експлуатацією готового продукту. Основною з них є проблема передачі додатку клієнту та користувачеві, а також забезпечення його

широкої експлуатації. Саме задля її вирішення був створений Docker – технологія та програмне забезпечення з відкритим кодом для автоматизації розгортання додатків з підтримкою контейнеризації.

Контейнеризація – це підхід до розробки програмного забезпечення, при якому додаток і його конфігурація (абстрактні файли маніфесту розгортання) упаковуються разом в образ контейнера. Контейнерний додаток може тестуватися як єдине ціле і розгортатися як екземпляр образу контейнера в операційній системі (ОС) вузла. Тобто контейнер – це стандартна одиниця програмного забезпечення, в якій упакований додаток з усіма необхідними для його роботи залежностями – кодом програми, середовищем запуску, системними інструментами, бібліотеками і налаштуваннями. Контейнеризація програмного забезпечення дозволяє розробникам та ІТ-фахівцям розгортати його в різних середовищах без будь-яких змін або з мінімальними змінами.

Docker запускає свої процеси у власній, заздалегідь налаштованій операційній системі. Треба зазначити, що при цьому всі процеси Docker працюють на фізичному host-сервері, поділяючи всі процесори і всю доступну пам'ять з іншими процесами, запущеними в host-системі. Контейнери надають такі переваги як ізоляція, кросплатформеність, гнучкість, масштабованість і контроль, протягом усього життєвого циклу програми. Найважливішою перевагою є ізоляція середовища розробки від робочого середовища.

Розглянемо інші можливі шляхи розгортання веб-додатків. Установчий скрипт дозволяє розгорнути все необхідне для роботи проекту на потрібних серверах. Недоліком цього способу є нестійкість до помилок, які можуть виникнути під час запуску, та неможливість повернення до початкової точки в разі їх виникнення. Інший спосіб розгортання – це хмарні сервіси, на які розробник власноруч установлює всі необхідні програми, інструменти, бібліотеки, а потім виконує клонування проекту задану кількість разів. Проте, через використання хмарних сервісів запуск рішення за межами хмари стає неможливим та спостерігається різке зниження швидкості роботи. Останній спосіб розгортання – це віртуальні машини, образи яких використовуються із заздалегідь встановленим та налаштованим додатком. Недоліками такого підходу є розмір образу, що може

бути занадто великим, а також необхідність повторного завантаження і встановлення образу при будь-якій зміні в додатку.

Наразі Docker стає все більш популярним завдяки простоті використання та надійності. Контейнери на базі Docker стають фактичним стандартом в цій області і підтримуються більшістю постачальників в екосистемі Windows і Linux, такими як Майкрософт, Amazon AWS, Google і IBM. Дуже скоро Docker, ймовірно, буде широко використовуватися як в хмарних, так і локальних центрах обробки даних. Статистика поширення використання Docker приведена на Рис.1.

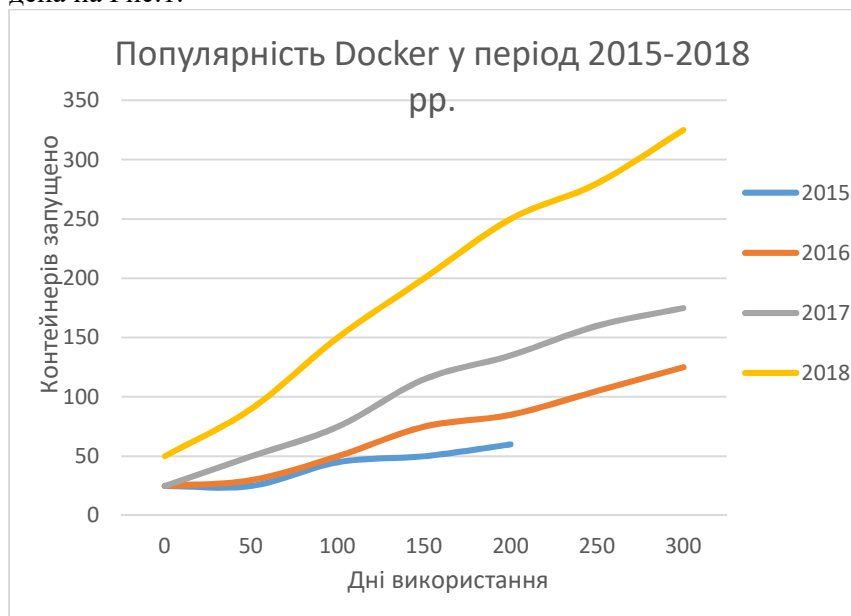


Рис.1. Популярність Docker у період 2015-2018 рр.

ВИСНОВКИ

Після порівняння шляхів автоматизації розгортання web-додатків було зроблено висновок, що Docker є зручнішим для вирішення проблем передачі серверного проекту клієнту, забезпечення тиражування серверів та оновлення раніше створених конфігурацій. Рішення на основі контейнерів надають важливі переваги з боку скорочення витрат, тому що дозволяють усувати проблеми, викликані відсутністю залежностей у робочому середовищі.

ЛІТЕРАТУРА

1. Docker Customers [Електронний ресурс] / Режим доступу: <https://www.docker.com/customers>
2. Docker [Електронний ресурс] / Режим доступу: <https://ru.wikipedia.org/wiki/Docker>
3. Docker. Зачем и как [Електронний ресурс] / Режим доступу: <https://habr.com/ru/post/309556/>
4. Usage Among Adopters [Електронний ресурс] / Режим доступу: <https://www.datadoghq.com/docker-adoption/>

*Вайдалаускас П. В.,
Державний університет «Житомирська політехніка»
Вакалюк Т. А.,
д.п.н., доцент, професор,
Державний університет «Житомирська політехніка»*

ВИКОРИСТАННЯ HIBERNATE ДЛЯ РОЗРОБКИ ПРОГРАМНИХ ДОДАТКІВ

Дано визначення технології Hibernate. Описано архітектура фреймворку та візуально представлено компоненти, що входять до складу технології. Проаналізовано переваги використання Hibernate технології для створення великих enterprise додатків.

Hibernate - це інструмент реляційного відображення об'єктів з відкритим кодом (ORM), який забезпечує основу для представлення об'єктно-орієнтованих моделей доменів у реляційних базах даних для веб-додатків.[1]

Об'єктивне реляційне відображення базується на контейнеризації об'єктів та абстракції, що забезпечує цю ємність. Абстракція дає змогу звертатися до об'єктів та керувати ними, не враховуючи, як вони пов'язані зі своїми джерелами даних. Hibernate ORM керує представленням класів Java до таблиць баз даних та типів даних Java на типи даних SQL та забезпечує запити та пошук.

Будь-які внесені зміни інкапсульовані у самому джерелі даних, так що коли ці джерела чи їх API змінюються, додатки, які використовують ORM, не повинні вносити зміни або навіть усвідом-

лювати цю інформацію. Так само програмісти можуть мати послідовний перегляд об'єктів у часі, хоча джерела, які їх доставляють, раковини, які їх отримують, та програми, які отримують доступ до них, можуть змінюватися.

Hibernate ORM дозволяє розробникам легше писати програми, дані яких переживають процес подання заявки. Як структура об'єктно-реляційного картографування (ORM), Hibernate переймається стійкістю даних, оскільки це стосується реляційних баз даних (через JDBC).

Окрім власного "рідного" API, Hibernate також є реалізацією специфікації Java Persistent API (JPA). Таким чином, його можна легко використовувати в будь-якому середовищі, що підтримує JPA, включаючи додатки Java SE, сервери додатків Java EE, контейнери Enterprise OSGi тощо.

Hibernate дозволяє розробляти стійкі класи за природними об'єктно-орієнтованими ідіомами, включаючи спадкування, поліморфізм, асоціацію, склад та бібліотеку колекцій Java. Hibernate не потребує інтерфейсів або базових класів для стійких класів і дозволяє будь-якому класу або структурі даних бути стійкими.

Hibernate підтримує ліниву ініціалізацію, численні стратегії отримання та оптимістичне блокування з автоматичною версією та тимчасовим штампуванням. Hibernate не вимагає спеціальних таблиць баз даних або полів і генерує велику частину SQL під час ініціалізації системи, а не під час виконання.

Hibernate послідовно пропонує кращі показники порівняно з прямим кодом JDBC, як з точки зору продуктивності розробника, так і в умовах виконання.

Hibernate був розроблений для роботи в кластерному сервері прикладних програм і забезпечити високомасштабну архітектуру, як показано на рис. 1. Hibernate добре масштабується в будь-якому середовищі: використовуйте її для управління внутрішньою мережею, що обслуговує сотні користувачів, або для критично важливих програм, які обслуговують сотні тисяч.

Hibernate добре відомий своєю чудовою стабільністю та якістю, що підтверджено прийняттям та використанням десятками тисяч розробників Java.

Hibernate легко настраюється та масштабується.

Hibernate Architecture

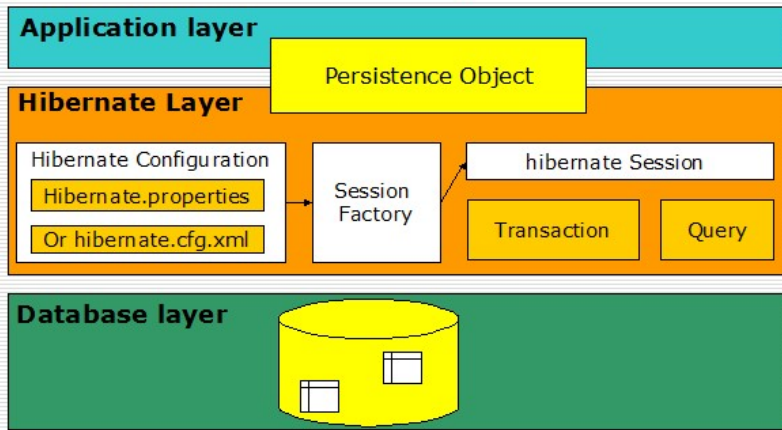


Рис. 1. Схема архітектури компонентів технології Hibernate [2]

ВИСНОВКИ

Таким чином, Hibernate це зручна технологія для розробки додатків, що потребують взаємодії з базою даних. Фреймворк має багатий інтерфейс використання та застосування для вирішення будь-яких задач.

ЛІТЕРАТУРА

1. Офіційний сайт Hibernate [Електронний ресурс] – режим доступу до ресурсу: <https://hibernate.org/>
2. Офіційний сайт блогів на тему технологій [Електронний ресурс] – режим доступу до ресурсу: <https://habiletechnologies.com/blog/reasons-to-choose-hibernate-over-jdbc/>

*Вайдалаускас П. В.,
Державний університет «Житомирська політехніка»
Вакалюк Т. А.,
д.п.н., доцент, професор,
Державний університет «Житомирська політехніка»*

АРХІТЕКТУРА ТА ОПИС ТЕХНОЛОГІЇ SPRING FRAMEWORK

Представлено технологію Spring Framework для розробки додатків. Описано архітектуру фреймворку та вказано проблеми, що можуть бути вирішені використовуючи інтерфейс. Продемонстровано короткий опис кожного модуля.

Spring Framework - платформа Java, яка забезпечує всебічну підтримку інфраструктури для розробки програм Java. Spring обробляє інфраструктуру, щоб можна було зосередитися на розробці свого додатку. [1]

Spring Framework складається з функцій, організованих приблизно в 20 модулів, як показано на рис. 1. Ці модулі згруповані в: базовий контейнер, доступ та інтеграція даних, веб, АОР (орієнтоване на аспектно програмовання), бібліотеки та інструменти, обмін повідомленнями та тести.

Модулі spring-core та spring-beans забезпечують основні частини основи, включаючи функції IoC та Injection Dependency. BeanFactory - це складне втілення фабричного зразка. Це знімає потребу в програмних синглтонах і дозволяє від'єднати конфігурацію та специфікацію залежностей від фактичної логіки програми.

Модуль spring-context побудований на міцній основі, що надається модулями Core та Beans: це інструмент доступу до об'єктного стилю, подібний до реєстру JNDI.

Модуль spring-express забезпечує потужну мову виразів для запитів та маніпулювання графіком об'єкта під час виконання. Це розширення уніфікованої мови вираження (уніфікована EL), як зазначено в специфікації JSP. Мова підтримує налаштування та отримання значень властивостей, призначення властивостей, виклик методів, доступ до вмісту масивів, колекцій та індексів, логічних та арифметичних операторів, названих змінних та отримання об'єктів

за назвою з контейнера IoC Spring Spring. Він також підтримує проєкцію та вибір списків, а також спільні агрегації списків.

Модуль `spring-aop` забезпечує реалізацію програмного забезпечення, орієнтовану на аспект AOP Alliance, що дозволяє визначати, наприклад, перехоплювачі методів та точкові вирізки для чистої розв'язки коду, який реалізує функціональність, яку слід розділити. Використовуючи функціональні можливості метаданих на вихідному рівні, ви також можете включати інформацію про поведінку у свій код таким чином, як атрибуту `.NET`.

Модуль `spring-jdbc` забезпечує рівень абстракції JDBC, який позбавляє від необхідності виконувати втомливе кодування JDBC та розбір специфічних кодів помилок для постачальника бази даних.

Модуль `spring-orm` забезпечує інтеграційні шари для популярних API-об'єктів реляційного відображення, включаючи JPA та Hibernate. Використовуючи модуль `spring-orm`, ви можете використовувати ці рамки відображення O / R у поєднанні з усіма іншими функціями Spring Spring, такими як простий декларативний елемент управління транзакціями, згаданий раніше.

Модуль `spring-web` надає основні функції інтеграції, орієнтовані на веб-сторінки, такі як функція завантаження файлів з декількома частинами та ініціалізація контейнера IoC за допомогою слухачів сервлетів та веб-орієнтованого контексту програми. Він також містить клієнт HTTP та пов'язані з Інтернетом частини підтримки віддаленої підтримки Spring.



Spring Framework Runtime

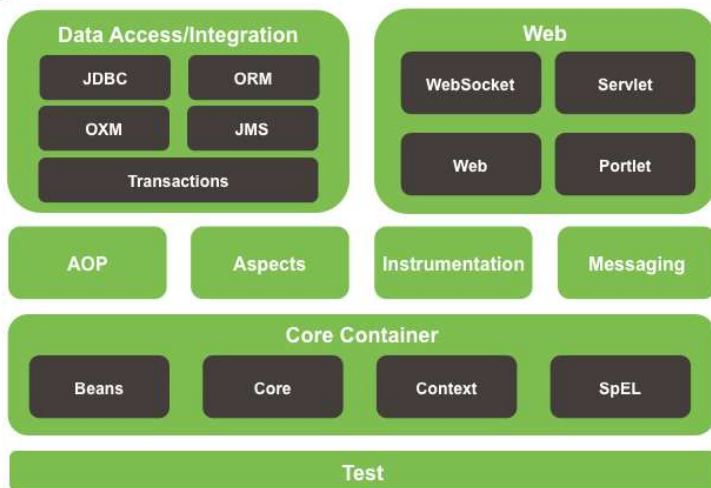


Рис. 1. Схема структури Spring Framework [2]

ВИСНОВКИ

Spring Framework – найпопулярніший фреймворк для розробки додатків використовуючи мову програмування Java. Технологія містить широкий стек бібліотек, що можна використовувати для вирішення різноматніх проблем.

ЛІТЕРАТУРА

1. Офіційний сайт Spring Framework [Електронний ресурс] – режим доступу до ресурсу: <https://spring.io>
2. Офіційний сайт документації Spring Framework [Електронний ресурс] – режим доступу до ресурсу: <https://docs.spring.io>

*Вайдалаускас П. В.,
Державний університет «Житомирська політехніка»
Вакалюк Т. А.,
д.п.н., доцент, професор,
Державний університет «Житомирська політехніка»*

ОСНОВНІ КОНЦЕПЦІЇ РЕАКТИВНОГО ПРОГРАМУВАННЯ

Дано визначення реактивному програмуванню. Обґрунтована роль функціонального програмування у поєднанні з реактивним. Описано основні концепції реактивного програмування.

Реактивне програмування використовується разом із функціональним програмуванням для вирішення задач, які базуються на подіях. [1]

З одного боку, функціональне програмування - це процес створення програмного забезпечення шляхом складання чистих функцій, уникання загального стану, змінних даних та побічних ефектів.

З іншого боку, реактивне програмування - це асинхронна парадигма програмування, що стосується потоків даних та поширення змін.

Спільно функціональне реактивне програмування утворює поєднання функціональних та реактивних прийомів, які можуть представляти елегантний підхід до програмування, орієнтованого на події - зі значеннями, які змінюються в часі і де споживач реагує на дані в момент надходження.

Ця технологія поєднує в собі різні реалізації основних її принципів, існують властивості, які визначають загальну лексику для опису нового типу додатків.

Реактивне програмування (рис. 1) охоплює в собі такі властивості [2]:

Responsive - системи повинні реагувати своєчасно

Message Driven - системи повинні використовувати асинхронну передачу повідомлень між компонентами, щоб забезпечити вільне з'єднання

Elastic - системи повинні залишатися чутливими при великому навантаженні

Resilient - системи повинні залишатися чутливими, коли деякі компоненти виходять з ладу

Під час роботи з реактивним програмуванням слід розуміти два ключових типи:

Observable являє собою будь-який об'єкт, який може отримувати дані з джерела даних і стан якого може представляти інтерес таким чином, щоб інші об'єкти могли реєструвати його

Observer є будь-який об'єкт, про який хочеться повідомити, коли стан іншого об'єкта змінюється

Observer підписується на послідовність спостереження. Послідовність посилає елементи спостерігачеві по одному. Observer обробляє кожен з них, перш ніж обробити наступний. Якщо багато подій надходять асинхронно, їх потрібно зберігати в черзі або опускати. Observer ніколи не буде викликаний з компонентом поза порядком або викликаний до того, як зворотний виклик повернеться для попереднього пункту.

Оператор - це функція, яка приймає один observable (джерело) як перший аргумент і повертає інший observable (призначення). Тоді для кожного елемента, який видає спостережуване джерело, застосовується функція для цього елемента, а потім створюється результат в компоненті Observable.

Операторів можна зв'язати між собою, щоб створити складні потоки даних, які фільтрують подію на основі певних критеріїв. До одного і того ж спостережуваного можна застосувати кілька операторів.

Не важко потрапити в ситуацію, коли Observable випромінює компонент швидше, ніж оператор чи спостерігач може їх споживати.

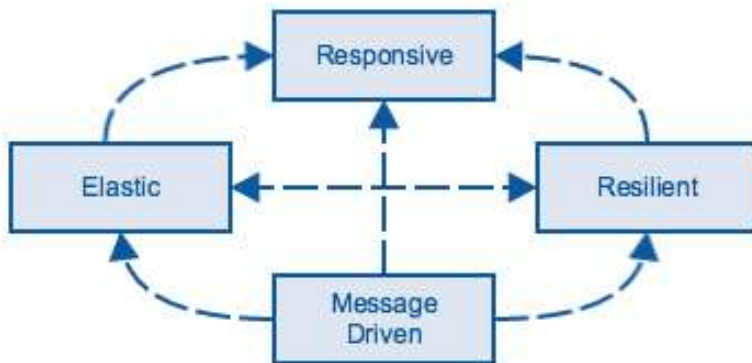


Рис. 1. Властивості реактивного програмування [2]

ВИСНОВКИ

Отже, реактивне програмування дуже корисно і зручно використовувати разом з функціональним для вирішення задач, які можна розв'язати за допомогою використання подій.

ЛІТЕРАТУРА

1. Офіційний сайт з описом реактивного програмування [Електронний ресурс] – режим доступу до ресурсу: <https://medium.com/ngx/introducing-to-reactive-programming-984c1a390e2d>
2. Офіційний сайт про реактивне програмування на Java [Електронний ресурс] – режим доступу до ресурсу: <https://www.baeldung.com/rx-java>

*Пехтерев І.В.
Криворізький національний університет
Купін А.І.
Д-р техн. наук, професор,
Криворізький Національний Університет*

СИСТЕМА ВЕБ-МОНІТОРИНГУ ВИПУСКНИКІВ УНІВЕРСИТЕТУ У СОЦІАЛЬНИХ МЕРЕЖАХ

Проаналізовано системи інтеграції модулів соціальних мереж у середу Веб-сайту університету на базі системи управління РНР. Обрано оптимальні засоби реалізації для подальшої роботи.

На теперішній час, як на етапі створення сайту на базі РНР, так при його модернізації, можлива інтеграція додаткових модулів та систем.

Розглянемо інтеграцію модуля від соціальної мережі типу Facebook для моніторингу її користувачів, а саме, наприклад, випускників університету. Для цього нам потрібна така прикладна програма, як RHP SDK Facebook. Створена вона розробниками Facebook саме для інтеграції модуля цієї соціальної мережі. Тому вона немає аналогів від сторонніх розробників, та у подальшому використовувати ми будемо саме її (рис. 1). [1]

Зазначена розробка має повний комплекс API для реалізації найбільш поширених типів взаємодій, кооперації та запитів.

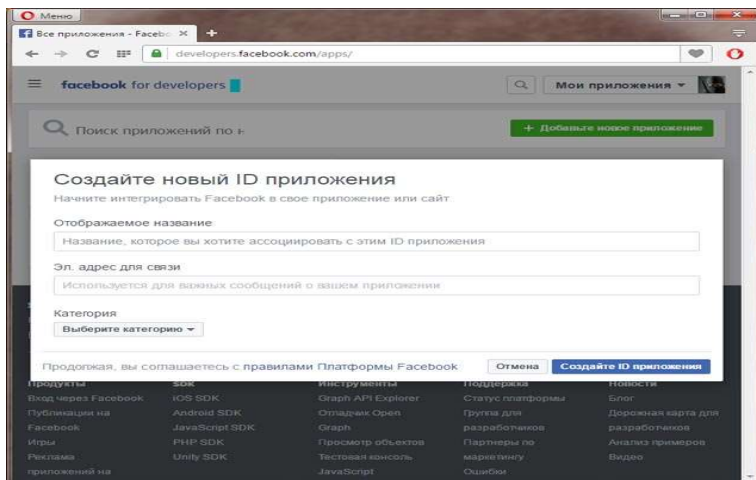


Рис. 1 – Приклад створення облікового запису для доступу к API соціальних мереж

ВИСНОВКИ

Таким чином, PHP SDK Facebook є, на нашу думку, найбільш вдалим рішенням для інтеграції модуля у сайт на базі PHP. Програма розроблена спеціалістами Facebook та практично немає гідних аналогів від сторонніх розробників.

ЛІТЕРАТУРА

1. Сайт shra.ru [Електронний ресурс] –Режим доступу до ресурсу: <https://shra.ru/>

*Трохліб Д.Г.
Державний університет “Житомирська політехніка”
Вакалюк Т.А.
Державний університет “Житомирська політехніка”*

ОПТИМІЗАЦІЯ ДИНАМІЧНОГО КОНТЕНТУ З ВИКОРИСТАННЯМ СУЧАСНИХ ВЕБ-ТЕХНОЛОГІЙ

Проаналізовано перспективи застосування JavaScript-бібліотек: ReactJS, AngularJS, Vue.js для інтерактивних веб-додатків. Дано характеристику Vue.js та зроблено порівняння його швидкодії з іншими технологіями.

Впровадження динамічного контенту на сайтах і персоналізація контенту - одна з важливих інтернет-маркетингових тенденцій останніх років. Постає питання оптимізації динамічного контенту та його швидкодії.

Vue.JS створювався з оглядкою на кращі практики сучасних технологій. З React.JS команда Vue запозичила ідею віртуального DOM. Цей підхід виключає пряму взаємодію з вузлами інтерфейсу. Початкова робота ведеться з його копією (virtual DOM). І тільки після цього зміни застосовуються до реальних вузлів інтерфейсу. Паралельно відбувається порівняння реального DOM дерева і його віртуальної копії [1]. Таким чином виявляється різниця і перемальовується тільки те, що зазнало змін.

Ядро Vue.JS, подібно React, містить лише необхідний функціонал для роботи з інтерфейсом. Тому воно компактно та легко інтегрується з іншими технологіями.

Vue.JS була розроблена своя технологія контролю стану програми - Vuex. Vuex повністю запозичує ідеї Redux, але ступінь інтеграції цієї бібліотеки з Vue набагато вище, ніж в разі React і Redux. А це трансформується в швидкодію і зручність.

У табл. 1 подано аналіз часу виконання операцій з DOM-ом сторінки різними JavaScript-бібліотеками.

Таблиця 1-Час виконання операцій з використанням різних JavaScript-бібліотек

Тривалість для виконання операції	JavaScript-бібліотека		
	Angular	React	Vue
Створити рядки, створити 1000 рядків у таблиці	143.6мм	165.7мм	162.3мм
Часткове оновлення, оновлення кожного 10-го ряду на 1000 рядків у таблиці	130.1мм	127.8мм	128.2мм
Вибір та виділення вибраного рядка	137.9мм	162.2мм	130.9мм
Заміна рядків, заміна 2 рядків для таблиці зі 1000 рядків	28.5мм	31.1мм	30.7мм

Видалити рядок, видалити один рядок	422.5мм	429.6мм	66.6мм
Створення рядків, створити 10 000 рядків	47.2мм	48.7мм	44.6мм
Додавання рядків до таблиці, додати 1000 рядків та таблиці зі 10000 рядками.	131.5мм	159.4мм	125.1мм
Очищення рядків, очищення рядків таблиці зі 1000 рядками	268.5мм	292.2мм	262.6мм

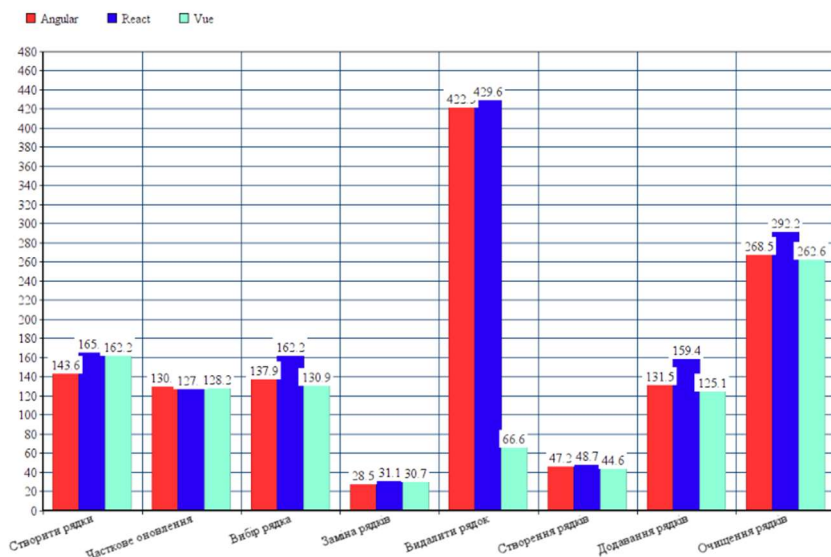


Рис. 1. Діаграма тривалості виконання операцій

Таким чином, Vue.JS являє собою нову технологію, що включила у себе кращі практики і реалізації з інших аналогічних бібліотек. Вона має привабливі функціональні можливості, такі як Vuex, two-way-data-binding та Virtual DOM. Бібліотека Vue.js забезпечує більш швидке оновлення сторінки, але це молода бібліотека, тому на даний час Вона програє у поширеності згаданим бібліотекам.

ЛІТЕРАТУРА

1. Vue.JS: особливості, переваги та недоліки [Електронний ресурс] – Режим доступу до ресурсу: <https://jetruby.com/ru/blog/vue-js-preimuschestva-i-nedostatki/>

Фоменченко В.В.

Державний університет «Житомирська політехніка»

Марчук Г.В.

*ст. викладач кафедри інженерії програмного забезпечення,
Державний університет «Житомирська політехніка»*

ОБҐРУНТУВАННЯ ВИБОРУ ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ РОЗРОБКИ ФІТНЕС-ДОДАТКУ

Автором в даній роботі було досліджено переваги та недоліки інструментальних засобів розробки програмного забезпечення та зроблено обґрунтування вибору засобів для розробки фітнес-додатку.

Створювати мобільні додатки – не тільки чудова професія, а й прибутковий бізнес. За даними AppAnnie ринок мобільних додатків в 2016 році оцінювався в \$50,9 мільярда доларів. Вже через 3 роки ця цифра зросла в 2.5 рази. Темпи росту не зменшуються, а навпаки збільшуються кожного кварталу. Вже до кінця 2021 року валовий годовий дохід може перевищувати \$189 мільярдів.

На ринку смартфонів найпопулярнішими операційними системами (надалі ОС) є iOS (розробка компанії Apple) та Android (розробка компанії Google). Для створення мобільного фітнес-додатку була обрана iOS. Вибір впав на неї по наступним причинам:

- З економічної точки зору проектування під платформу iOS значно вигідніше ніж під Android. Так як за даними першого кварталу 2019 року дохід від додатка для мобільних пристроїв під iOS майже в 2 рази більший ніж під Android.
- Тестування бета версій, написаних для iOS, значно легше через існування такого сервісу від Apple як TestFlight.

- iOS являється більш надійною ОС. На дану платформу практично не здійснюються жодні хакерські атаки.

Основною мовою програмування був обраний Swift. Це швидко сучасна, безпечна та зручна мова. З її допомогою процес створення програм стає дуже гнучким і продуктивним, так як Swift увібрав в себе найкраще з таких мов, як C, Objective-C та Java. Ця мова має вкрай перспективне майбутнє [1].

Головною перевагою Swift є те, що саме за допомогою неї можна створювати повністю нативні мобільні додатки. Це зумовлено тим, що мова була створена саме компанією Apple для своїх продуктів. Крім цього є ще ряд переваг:

- Швидкість роботи програм написаних на мові Swift майже на одному рівні з C++

- Swift викликає помилку при виконанні коду при використанні змінної nil. Це дозволяє полегшити процес усунення багів, оскільки програміст вирішує проблеми відразу

- Є можливість працювати в Swift Playgrounds, де програмісти можуть відразу перевірити новий алгоритм не створюючи повноцінну програму

- Досить нова мова, а отже відповідає усім потрібним вимогам

Для розробки програмного продукту був використаний фреймворк UIKit. Він визначає основні компоненти програми iOS, від ярликів і кнопок до табличних уявлень і контролерів навігації. Він базується на таких фреймворках як The Foundation та Cocoa Touch. Завдяки цьому ми маємо змогу працювати не тільки з елементами, що відповідають за графічне відображення, а й з усіма іншими базовими рівнями функціональності, включаючи зберігання та збереження даних, обробку тексту, обчислення дати та часу, сортування та фільтрацію.[2]

Завдяки фреймворку UIKit ми маємо змогу створювати повністю нативні мобільні додатки, що будуть працювати значно краще, ніж кросплатформені рішення такі як, наприклад, Xamarin чи ReactNative.

Для зберігання та відображення даних, мобільному додатку знадобиться база даних(надалі БД). Під час вибору найкращого рішення в даному питанні були розглянуті дві найпопулярніші БД: Firebase та Realm. Кожне з цих рішень має свої плюси та мінуси. Firebase це BaaS (Backend-as-a-service) сервіс, що надає змогу користуватися базою даних в реальному часі, мати власне сховище для мета-даних(Firebase Storage), та вбудована аунтіфікація користувачів(Firebase Authentication). Дані в Firebase зберігаються в JSON. Завдяки тому, що Firebase має Realtime Database, ми можемо бачити зміни, які відбуваються в базі даних в реальному часі.

Другим сервісом, що був розглянутий є Realm – це комбінація серверних і клієнтських компонентів на основі NoSQL, підключених за допомогою швидкого і ефективного протоколу синхронізації. Даний сервіс більш спрямований на надання змоги створення локальної бази даних. Це не підходить для створення мобільного фітнес додатку, так як не буде змоги зберігати дані користувача після видалення програми зі смартфона та оновлення даних для тренувань чи списку доступних страв в реальному часі. Realm має платформу, яка виступає в ролі BaaS сервісу, проте вона поступається по швидкості Firebase та немає такого великого функціоналу. Отже було обрано Firebase як основну БД.

ЛІТЕРАТУРА

1. Усов В. Swift. Основы разработки под iOS и MacOS / Василий Усов. – СПб: Питер, 2017. – 368 с.
2. Matt N. Programming iOS 11 / Neuburg Matt. – Beijing, Boston, Farnham, Sebastopol, Tokyo: O'Reilly, 2018. – 1172 с.

Ярмоленко Д.А.
ДУ «Житомирська політехніка»
Марчук Г.В.
ст.викладач кафедри інженерії програмного забезпечення
ДУ «Житомирська політехніка»

ОБҐРУНТУВАННЯ ВИБОРУ ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ ДЛЯ РОЗРОБКИ СИСТЕМИ ОБЛІКУ ВІДВІДУВАННЯ ЗАНЯТЬ СТУДЕНТАМИ

В даній роботі проведено аналіз операційних систем, мов програмування та баз даних, виявлено їх переваги та недоліки, на основі чого обґрунтовано вибір інструментальних засобів для розробки системи обліку відвідування занять студентами.

Порівнявши більшість операційних систем (ОС) на ринку, можна виділити Android, так як він має більш суттєві переваги. Визначальним є те, що пристроїв на даній ОС найбільша кількість і це можливість для охоплення більшої аудиторії користувачів. Важливий момент, що Android постійно оновлюється і в нові версії додають все більше і більше можливостей як для користувачів, так і для розробників.

Також є досить зручне середовище для розробки – Android Studio. В даному середовищі є всі необхідні інструменти для простого створення програм, інтерфейсів.

Вибір мови програмування додатку – відповідальний крок. Від вибору мови залежить подальше використання технологій та продуктивність роботи даної системи.

На сьогоднішній день існує великий вибір мов програмування для мобільних пристроїв. Але зважаючи на рейтинги компанії Gartner найбільш популярними мобільними ОС є Android, IOS, Windows Mobile, частка ринку яких дорівнює 59%, 38% та 3% відповідно.

Для збереження даних користувачів потрібно використання бази даних.

Вибір бази даних також є важливим етапом, так як від цього залежить швидкість роботи додатку з даними які там зберігаються. Серед варіантів використання є Firebase, MySQL.

Android

Ця ОС має найбільшу частку на ринку мобільних пристроїв. Через це компанія Google в останніх версіях ОС Android намагається максимально оптимізувати систему для зручного використання на безлічі телефонів незважаючи на їх продуктивність.

Основними перевагами є:

- найпопулярніша мобільна ОС;
- зручне середовище для розробки;
- велика кількість допоміжних технологій;
- допомога в створенні інтерфейсів завдяки MDK;

До недоліків можна віднести:

- існує потреба в максимальній оптимізації програми через те, що велика кількість пристроїв;
- складність написання коду;
- переважна більшість бюджетних пристроїв на даній ОС, використання старих версій ОС, що призводить до їх меншої продуктивності.

Дана ОС є найкращим варіантом для використання при необхідності максимального поширення додатків, які не є дуже вибагливими до продуктивності телефонів.

IOS

Ця ОС є другою за популярністю. Використовується лише на пристроях однієї марки, що робить її максимально швидкою та продуктивною. Є конкурентом Android.

До переваг можна віднести :

- швидкість роботи;
- легкість написання коду;
- постійна актуальність системи навіть на старих пристроях;
- велика кількість допоміжних технологій;
- просте створення інтерфейсів для користувача та їх взаємодії між собою;

До недоліків віднесемо:

- висока вартість пристроїв;
- можливість створювати програми лише з продуктів компанії Apple;
- складніше розповсюдження додатків;

Ознайомившись з необхідним стеком інформації було визначено основні акценти системи обліку відвідування студентів. Було враховано основні недоліки та переваги програм-конкурентів для створення максимально зручного продукту. Для цього було розроблено зручний інтерфейс, оптимально необхідний функціонал програми.

База даних була обрана Firebase, так як це програмний продукт компанії Google, як і ОС Android. Дана БД є легкою в використанні, дозволяє проводити аналітику використання програми, має зручний зовнішній вигляд. Мова написання обрана нативна – Java. Ця мова програмування рекомендована до використання компанією Google, що забезпечує максимальну швидкодію в роботі ПП.

ЛІТЕРАТУРА

1. Книги по программированию: как читать и что именно [Електронне джерело] - Режим доступу до ресурсу: <https://proglib.io/p/programming-books/>
2. Material Design Guide [Електронне джерело] – Режим доступу до ресурсу: <https://material.io/design/>
3. Android SDK Guide [Електронне джерело] – Режим доступу до ресурсу: <https://developer.android.com/studio/intro>

*Mohammed Azad Sabgha
Zubov Dmytro, PhD
VŠB - Technical University of Ostrava*

FAST OBJECT DETECTION WITH ROTATION AND SCALE INVARIANCE VIA SYMMETRIC POSITIONS OF A PIXEL: A CONCEPT FOR IMAGE PROCESSING

A new approach of the fast object detection with rotation and scale invariance via symmetric positions of a pixel with low computational complexity is discussed for image processing. The information in the pixel might be presented using the gradient magnitude only since the angle is known. This work is a part of the project “Smart City Mesh Network of eHealth Intelligent Agents for Visually Impaired and Blind People”.

Key words: *Image processing, object detection, rotation and scale invariance.*

INTRODUCTION

Visual correspondence, object detection, decomposition of an image into local regions of interest, and many other vision applications depend on stable and representative features in the image. The main difficulty is the compact and robust description of keypoints invariant to scale, rotation, affine transformation, and noise. The past decades showed that there is no universal algorithm to efficiently describe keypoints and match them.

The most popular descriptor is the histogram of oriented gradient proposed by Dr. Lowe to represent the Scale Invariant Feature Transform (SIFT) keypoints. Most of the efforts in the last years were to speed up the SIFT approach and make it with lower computational complexity. Speeded Up Robust Features (SURF), Binary Robust Invariant Scalable Keypoints (BRISK), Features from Accelerated Segment Test (FAST), and Fast Retina Keypoint (FREAK) [1-3] are good examples of that. They were proposed to make the objects detection faster and less computationally expensive.

The above-stated algorithms do not consider specificities of detected objects. For instance, FAST detects the keypoint via a certain number of surrounding pixels, e.g. a minimum of nine neighbor pixels out of sixteen must have intensities greater or less than the central one. In this work, we propose a new approach with low computational complexity. The main idea is to analyze symmetric positions of a pixel around the center and specified axis.

FAST OBJECT DETECTION WITH ROTATION AND SCALE INVARIANCE VIA SYMMETRIC POSITIONS OF A PIXEL

In this algorithm, we use a feature of the image descriptor that the pixel goes from one position to another if a specific angle is applied. Figure 1 shows an example of two pixels rotation with angle 22.6° . In this figure, pixels with coordinates (2,3) and (1,5) go to positions with coordinates (3,2) and (-1,5), respectively. We can easily extend this example to pixels with coordinates (2,-3) and (-1,-5) – they go to positions (3,-2) and (1,-5) if rotation angle 22.6° , counterclockwise. Another example of rotation angle 12.7° is as follows: pixels with coordinates (1,9) and (5,4) – they go to the positions (-1,9) and (4,5), respectively.

The scaling is done with specific factors. For instance, if the factor is 2, the pixel (1,9) goes to the position (2,18). The information in the

pixel might be presented using the gradient magnitude [4] only since the angle is known.

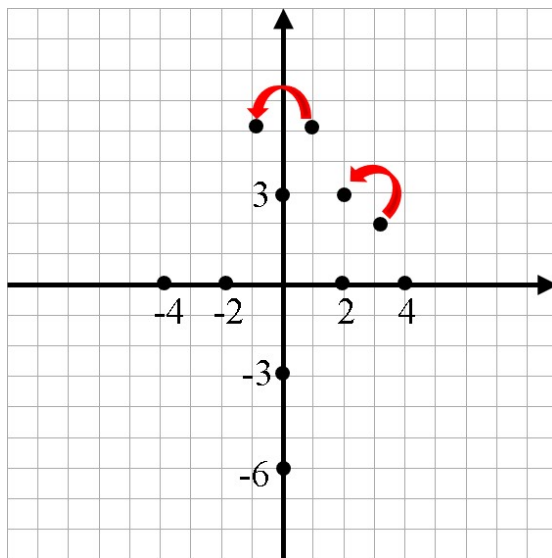


Fig. 1. An example of two pixels rotation with angle 22.6°

Nowadays, the set of possible pixels with rotation and scale invariance is designed using their symmetric positions. This work is a part of the project “Smart City Mesh Network of eHealth Intelligent Agents for Visually Impaired and Blind People”.

SUMMARY AND CONCLUSION

In this paper, a new approach of the fast object detection with rotation and scale invariance via symmetric positions of a pixel with low computational complexity is discussed for image processing. The information in the pixel might be presented using the gradient magnitude only since the angle is known.

This work is a part of the project “Smart City Mesh Network of eHealth Intelligent Agents for Visually Impaired and Blind People”.

REFERENCES

1. Lowe, D. (1999) Object recognition from local scale-invariant features. In Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on, Vol. 2, 1150-1157.

2. Bay, H., Tuytelaars, T., & Van Gool, L. (2006). Surf: Speeded up robust features. *Computer Vision – ECCV 2006*, 404-417.
3. Kapur, S. & Thakkar, N. (2015). *Mastering OpenCV Android Application Programming*. Birmingham, UK: Packt Publishing.
4. Ghosh, H. & Sharma, M. (2015). Histogram of gradient magnitudes: A rotation invariant texture-descriptor. TCS Innovation Labs, Delhi TATA Consultancy Services Ltd., 4614-4618.

СЕКЦІЯ 5. ARTIFICIAL INTELLIGENCE. КОМП'ЮТЕРНІ СИСТЕМИ ТА МЕРЕЖІ ШТУЧНОГО ІНТЕЛЕКТУ

Лебедєв В.О.
Харківський національний університет
радіоелектроніки
Руденко О.Г.,
д.т.н., професор, Харківський національний
університет радіоелектроніки.

ДОСЛІДЖЕННЯ НЕЛІНІЙНИХ АЛГОРИТМІВ НАВЧАННЯ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ ПРЯМОГО ПОШИРЕННЯ

Розглянуто питання імітаційного моделювання процесу навчання ШНМ прямого поширення в різних умовах за допомогою різних алгоритмів, зокрема багатокрокових проєкційних алгоритмів, в різних умовах функціонування нейронних мереж. Аналіз результатів моделювання дозволив розробити рекомендації щодо практичного використання алгоритмів навчання.

Штучні нейронні мережі (ШНМ) завдяки своїй ефективності отримали останнім часом широке розповсюдження при вирішенні самих різноманітних задач. Синтез або використання ШНМ потребує рішення задач структурної та параметричної оптимізації, які відповідають вибору оптимальної топології мережі та її навчання. Якщо задача визначення структури є дискретною оптимізаційною (комбінаторною), то пошук оптимальних параметрів здійснюється в безперервному просторі за допомогою класичних методів оптимізації, що засновані, як правило, на методі найменших квадратів (МНК). На сьогоднішній день існує велика кількість стандартних архітектур нейронних мереж, що істотно полегшує задачу побудови ШНМ з нуля і зводить її до підбору підходящої конкретному завданню структури мережі.

Для навчання мережі застосовуються, як правило, методи, які вимагають обчислення градієнта функціоналу (зазвичай квадратич-

ного), що використовується (алгоритм зворотного поширення помилки (ОР), метод сполучених градієнтів, алгоритм Гауса-Ньютона, Левенберга-Марквардта і т.д.). Вибір функціоналу іншого виду приводить до отримання алгоритму, відмінного від алгоритмів, що мінімізують квадратичний функціонал (типу МНК), і, отже, маючого інші властивості. Найбільш поширеним серед таких функціоналів є модульний, мінімізація якого здійснюється за допомогою методу найменших модулів (МНМ). Слід зазначити, що ефективність МНМ при гаусових розподілах завади нижче, ніж МНК, але при наявності завад з розподілами з важкими хвостами (наприклад, Лапласа або Коші) ефективність МНМ істотно перевищує ефективність МНК. Таким чином, алгоритми МНМ мають кращі в порівнянні з алгоритмами МНК робастні властивості.

Алгоритми МНК і МНМ для налаштування параметрів використовують всі наявні спостереження, що не завжди зручно при дослідженні нестационарних об'єктів. З іншого боку, застосування алгоритмів типу стохастичного градієнта або Уїдрой-Хоффа дозволяє відстежувати зміну параметрів, що оцінюються (налаштовуються), з деяким запізненням, величина якого часто є значною. [1]. Тому виникає задача розробки та дослідження багатокрокових проєкційних алгоритмів навчання ШНМ, які використовують не всю наявну інформацію, а лише деяку її постійну частину. Кількість такої інформації визначає пам'ять алгоритмів. Як показано в [2], це відповідає зменшенню розмірності розв'язуваної задачі на величину, яка визначається цією кількістю інформації.

В роботі розглядаються питання імітаційного моделювання процесів навчання ШНМ прямого поширення в різних умовах за допомогою різних алгоритмів. Вивчалися властивості алгоритмів за наявністю гаусовських і негаусовських завад. Крім того, проводилися дослідження для різних видів нестационарності параметрів, що налаштовуються (лінійний дрейф, гармонійний, випадкове змінення параметрів тощо). Одним важливим дослідженням було визначення оптимального вибору розміру пам'яті (кількості інформації) в багатокрокових проєкційних алгоритмах.

На основі аналізу результатів моделювання були розроблені рекомендації щодо практичного використання алгоритмів навчання.

ЛІТЕРАТУРА

1. Руденко О.Г., Лебедев В.О., Олейник К.О. Алгоритми навчання штучних нейронних мереж прямого поширення, що використовують нелінійні перетворення матриці спостережень // Матеріали XVII міжнародної науково-практичної конференції «Математичне та програмне забезпечення інтелектуальних систем». (MSSIS-2019) 20-22 листопада 2019. Дніпро. – 2018.-С. 222-223.
2. Ищенко Л.А., Либероль Б.Г., Руденко О.Г. О свойствах одного класса многошаговых алгоритмов идентификации // Кибернетика.-1985.-№ 5.-С.1-17.

Олейник К.О.

*Харківський національний університет
радіоелектроніки*

Руденко О.Г.,

*д.т.н., професор, Харківський національний
університет радіоелектроніки.*

АЛГОРИТМИ НАВЧАННЯ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ ПРЯМОГО ПОШИРЕННЯ, ЩО ВИКОРИСТОВУЮТЬ НЕЛІНІЙНІ ПЕРЕТВОРЕННЯ МАТРИЦІ СПОСТЕРЕЖЕНЬ

Розглянуто алгоритми навчання штучних нейронних мереж прямого поширення, які з метою спрощення реалізації використовують нелінійні перетворення матриці спостережень. Однак застосування такого перетворення потребує рішення питання підвищення обчислювальної стійкості даних алгоритмів. Результати моделювання свідчать про ефективність підходу, що розглядається.

При використанні штучних нейронних мереж (ШНМ) для рішення практичних задач вельми важливим є вибір алгоритму навчання мережі, особливо це стосується випадку роботи мережі в реальному часі.

Базовими стратегіями навчання є: навчання з учителем і навчанням без учителя. У разі навчання з учителем, особливість полягає в тому, що для кожного навчального образу, висунутого мережі,

також пред'являється деяка інформація про коректність відгуку мережі.

Навчання ШНМ - ітеративний процес. На кожній ітерації обчислюються мережеві виходи для одного (або більше) зразка в наборі навчання, і коригуються мережеві ваги з метою зменшення помилки між фактичним мережевим виходом і цільовим виходом для даного зразка. При використанні підходящої мережевої архітектури і алгоритму навчання мережеві ваги прагнуть до значень, при яких мережевий вихід стає прийнятно близьким до цільового виходу для кожного зразка в наборі навчання.

Вибір функції помилки має настільки ж істотний вплив на ефективність ШНМ, як і вибір алгоритму навчання. Найбільш широко використовуваними функціями помилки є квадратична функція помилки і нормалізована її версія, яка називається середньоквадратичною функцією помилки.

Перевага останньої полягає в тому, що вона нечутлива як до числа зразків в наборі навчання, так і до числа нейронів у вихідному шарі мережі, що дозволяє використовувати її для порівняння в різних завданнях навчання.

Навчання ШНМ є успішним при досягненні мережевою помилкою припустимого значення, що визначається з умов конкретного завдання. Слід зазначити, що досягнення мінімального значення помилки часто є небажаним, тому що призводить до явища перенавчання ШНМ і погіршення її фільтруючих і узагальнюючих властивостей. Виняток становлять завдання апроксимації функцій, для яких потрібна висока точність одержуваних рішень. Навчання ШНМ прямого поширення полягає в корекції її вагових параметрів на основі мінімізації цієї помилки. Реалізація застосовуваних для цього градієнтних алгоритмів труднощів не викликає, тим більше, що для обчислення градієнтів можна використовувати спеціальні бібліотеки, такі як Theano і TensorFlow.

Алгоритми МНК і МНМ для налаштування параметрів використовують всі наявні спостереження, що не завжди зручно при дослідженні нестационарних об'єктів. Тому перспективним є застосування для навчання ШНМ багатокрокових проєкційних алгоритмів, які використовують не всю наявну інформацію, а лише деяку її постійну частину. Кількість такої інформації визначає пам'ять алгори-

тмів. Поліпшення робастних властивостей багатокрокового алгоритму може бути досягнуто введенням в даний алгоритм не лінійності [1]. Крім того, використання в алгоритмі нелінійності типу знакової функції ($\text{sign}(x)$) має зменшити кількість обчислень, необхідних для реалізації даного алгоритму, тому що при обчисленні скалярних добутків типу $x^T \text{sign}(x)$ зникає операція множення. Прикладом такого широко відомого нелінійного алгоритму є одно кроковий алгоритм Нагумо-Ноди, який являє собою окремий випадок багатокрокового нелінійного проєкційного алгоритму. Однак при використанні такої операції виникає проблема підвищення обчислювальної стійкості нелінійних алгоритмів. Особливо це стосується багатокрокових нелінійних проєкційних алгоритмів, тому що реалізація даних алгоритмів вимагає обертання відповідної матриці спостережень. У доповіді розглядаються питання спрощення реалізації багатокрокових алгоритмів, засновані на рекурентному обчисленні оберненої матриці, і підвищення їх обчислювальної стійкості при використанні ідеї регуляризації. Отримано рекурентні процедури, які відповідають різному способу організації обчислень (скидання застарілої інформації і включення нової). Наведено результати імітаційного моделювання, що свідчать про ефективність підходу, що розвивається.

ЛІТЕРАТУРА

1. Руденко О.Г., Лебедев В.О., Олейник К.О. Алгоритми навчання штучних нейронних мереж прямого поширення, що використовують нелінійні перетворення матриці спостережень // Матеріали XVII міжнародної науково-практичної конференції «Математичне та програмне забезпечення інтелектуальних систем». (MSSIS-2019) 20-22 листопада 2019. Дніпро. – 2018.-С. 222-223.

*Романюк О.С.
Харківський національний університет
радіоелектроніки
Руденко О.Г.,
д.т.н., професор, Харківський національний
університет радіоелектроніки.*

ДО ВИБОРУ МАТЕМАТИЧНОЇ МОДЕЛІ В ЗАДАЧІ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ

Розглянуто та проаналізовано проблеми вибору математичної моделі в задачі прогнозування часових рядів. Розроблено рекомендації щодо найбільш ефективного такого вибору.

Задача прогнозування часових рядів (ЧР) була і залишається актуальною, особливо останнім часом, коли стали доступними потужні засоби збору та обробки інформації. Прогнозування часових рядів є важливою науково-технічною проблемою, так як дозволяє передбачити поведінку різних факторів в екологічних, економічних, соціальних та інших системах.

Метою будь-якого прогнозування є створення моделі, яка дозволяє прогнозувати майбутнє і оцінити тенденції в змінах того чи іншого фактора. Всі методи прогнозування, як і класичні, так і сучасні, потребують побудови математичної моделі, якість якої і обумовлює точність прогнозування.

СТАТИСТИЧНІ МОДЕЛІ

Розглянуто класичні моделі авторегресії та ковзного середнього (ARMA, ARIMA), моделі експоненціального згладжування, моделі прогнозування на основі ланцюгів Маркова, класифікаційно-регресійні дерева, нелінійні регресійні моделі (NARX та NARMAX моделі). При побудові нелінійних моделей використовують два принципово різні підходи, засновані на різних способах формування за допомогою моделі прогнозованого сигналу об'єкта, які відповідають побудові паралельної і послідовно-паралельної моделі.

Наведено результати порівняльного аналізу моделей прогнозування, в яких відображено переваги і недоліки моделей.

Слід, однак, зазначити, що більшість практичних задач прогнозування ЧР характеризуються не тільки високим рівнем нелінійності, а і нестаціонарністю, зашумленістю, наявністю нерегулярних трендів, стрибків, аномальних викидів тощо. У цих умовах жорсткі статистичні припущення про властивості ЧР часто призводять до неефективності традиційних методів, а в ряді випадків — до їх непридатності [1].

НЕЙРОМЕРЕЖЕВІ МОДЕЛІ

Альтернативою статистичним методам можуть служити методи обчислювального інтелекту, до числа яких, в першу чергу, слід віднести штучні нейронні мережі. Здатність нейронної мережі до різнобічної обробки інформації впливає з її здатності до узагальнення і виділення прихованих залежностей між вхідними та вихідними даними.

При нейромережевому підході досліджуваний об'єкт представляється у вигляді нейронної мережі, що містить крім вхідного і вихідного один або кілька прихованих шарів, кожен з яких складається з певної кількості нейронів, які реалізують задану базисну функцію (функцію активації) [2].

При побудові паралельної або послідовно-паралельної нейромережевої моделі використовується апріорна інформація щодо класу, до якого вона належить, а задача побудови такої моделі полягає в обчисленні ваг нейронних мереж, так щоб нейромережеве представлення нелінійностей було еквівалентним класу, що розглядається.

Абсолютну точність прогнозу оцінювати не прийнято, тому що для кожного конкретного завдання вона може кардинально відрізнятись, тому оцінюють саме помилку прогнозу, яка характеризується деякою метрикою (критерієм).

Необхідно зазначити, що при цьому для побудови моделі (вибору її структури та оцінки параметрів) використовуються ті ж критерії, які характеризують точність прогнозування. У доповіді аналізуються існуючі і досить поширені статистичні (RMSE, MAPE, MAE та інші) і деякі інформаційні критерії (AIC, KIC, HQ тощо) (метрики) вибору математичної моделі для її використання в задачі

прогнозування часових рядів. Розглянуто особливості їх використання, описано переваги та недоліки. На основі результатів аналізу розроблено рекомендації щодо вибору критерію при вирішенні практичних задач.

ВИСНОВКИ

Проведений аналіз показав, що неможливо виділити один універсальний критерій оцінки ефективності моделі, бо на різних наборах даних виходять різні результати оцінок помилок прогнозування, тобто кожен критерій має недоліки, які можуть призводити до неточного визначення якості прогнозування U зв'язку з цим рекомендується використовувати певний набір метрик помилки.

ЛІТЕРАТУРА

1. Руденко О.Г., Безсонов О.О., Романюк О.С.. Про вибір математичної моделі для прогнозування часових рядів // Матеріали XVII міжнародної науково-практичної конференції «Математичне та програмне забезпечення інтелектуальних систем». (MSSIS-2019) 20-22 листопада 2019. Дніпро. – 2018.-С. 221-222.
2. Руденко О.Г., Безсонов О.О., Романюк О.С.. Нейромережеве прогнозування часових рядів на основі багатопарового перцептрона // Бионика интеллекта.-2018. -№1(90).-С.79-83)

Ярош І.В.

старший викладач,

Донецький національний технічний університет

Черняк Т.О.

асистент, Донецький національний технічний університет

ДОСЛІДЖЕННЯ МЕТОДІВ РОЗПІЗНАВАННЯ ОБЛИЧЧЯ ЛЮДИНИ ЗА ДОПОМОГОЮ НЕЙРОННИХ МЕРЕЖ

На основі дослідження була спроектована та розроблена нейронна мережа – неокогнітрон, яка навчається за допомогою генетичного алгоритму.

Незважаючи на існуючі проблеми, бажання змоделювати людський мозок не згасає, а отримання вражаючих результатів надихає на подальші дії. Успішні моделі, засновані на припущеннях

про структуру мозку, розроблюються нейроанатомами та нейрофізіологами з метою вивчення й узгодження структури та функцій цих моделей.

З іншого боку, успіхи в біологічній науці ведуть до модифікації та ретельної розробки штучних моделей. Аналогічно інженери застосовують штучні моделі для вирішення світових проблем і отримують позитивні результати, незважаючи на необхідність подальшого розвитку. [1]

Метою дослідження є розробка та навчання нейронної мережі для розпізнавання обличч людей.

В процесі роботи були виконані наступні поставленні задачі: дослідження сучасних систем та методів розпізнавання обличч людей; розробка структури загорткової нейронної мережі; навчання нейронної мережі на розміченій базі зображень генетичним алгоритмом; тестування навченої мережі.

Точність розпізнавання розробленого додатка, на відміну від класичного, може дати набагато точніші результати за рахунок більшого поля варіативності ваг, що налаштовуються. Однак, хоч і в меншій мірі, вона також залежить від структури неокогнітрона, яку необхідно підбирати під зображення, що розпізнаються.

Під час тестування розроблена мережа, завдяки методу навчання, перевершувала в якості розпізнавання обличч неокогнітрон, який використовує класичний метод навчання. Однак отримання такої переваги вимагає тривалого навчання. І за деяких варіантів структури мережа приводить до схожих з класичним методом результатів.

ВИСНОВКИ

В подальшому отриманий алгоритм може бути оптимізовано для зменшення часу навчання шляхом додавання богатопоточності при обчисленні «фітнес» функції, на яку витрачається найбільша частина часу.

ЛІТЕРАТУРА

1. Мак-Каллок У. С., Питтс В. Логическое исчисление идей, относящихся к нервной активности // Автоматы / под ред. К. Э. Шеннона и Дж. Маккарти. – М.: Изд-во иностр. лит., 1956. – С. 363–384.

*Кокозей А. Ю.,
Криворізький Національний університет
Шаповалова Н. Н.,
ст. викладач, Криворізький Національний університет*

ДОСЛІДЖЕННЯ АСПЕКТІВ СТРАТЕГІЙ СЕМПЛІНГУ

Розроблено модуль для генерації вибірок даних з використанням різних стратегій семплінгу.

В задачах машинного навчання якість моделей сильно залежить від вихідних даних, але дані дуже рідко бувають ідеальними. Отже, перед початком побудови моделей, необхідно провести певні міри, щодо перетворення даних у більш інформативні та зручні для роботи. Такі міри називають попередньою обробкою даних [1]. Найважливіші етапи попередньої обробки даних включають: агрегацію (об'єднання двох або більше ознак в одну), зменшення розмірності, нормалізацію ознак (перетворення значень ознак у абсолютно інший набір значень за допомогою простих функцій), семплінг даних та інші.

Семплінг даних – це метод дослідження множини даних, що використовується для вибору, маніпулювання та аналізу репрезентативної підмножини для виявлення закономірностей та тенденцій. Репрезентативність семплу (вибірки) – це здатність вибіркової сукупності відтворювати основні характеристики генеральної сукупності. Репрезентативність досягається за рахунок правильного формування семплу, який, за принциповими для дослідження параметрами, має відтворювати генеральну сукупність об'єктів. Таким чином, семплінг дає можливість працювати з невеликою, керованою кількістю даних, щоб швидше будувати та запускати аналітичні моделі, при цьому отримуючи точні результати.

Семплінг може бути особливо корисним для наборів даних, які занадто великі для ефективного аналізу. Ідентифікація та аналіз репрезентативного зразка є більш ефективними, зокрема економічно, ніж аналіз усієї сукупності даних. Важливим фактором є розмір семплу даних та можливість введення помилки вибірки.

Помилка вибірки – це відхилення результатів, отриманих за допомогою вибіркового спостереження від справжніх даних генеральної сукупності. Помилки вибірки можна розділити на два класи: це помилки реєстрації та помилки репрезентативності.

До помилок реєстрації відносять випадкові та статистичні помилки. Випадкові помилки – це статистичні похибки, властиві вибіркового методу. Такі помилки зменшуються при зростанні обсягу семплу. Систематична помилка залежить від різних факторів, що впливають на дослідження, зміщують результати дослідження в певну сторону та залежать від якості вихідної генеральної сукупності.

Важливу роль у семплінгу грає розмір семплу. У деяких випадках невелика вибірка зможе бути достатньо репрезентативною та відобразити найважливішу інформацію про набір даних. В інших, використання більшої вибірки може збільшити ймовірність точного представлення даних у цілому, навіть якщо збільшений розмір вибірки може перешкоджати простоті маніпулювання та інтерпретації.

Існує багато різних методів для отримання зразків з даних, відповідний обирається залежно від набору даних та ситуації. Методи семплінгу можна поділити за двома типами. Ймовірнісний семплінг – це такий вид пошуку підмножини, при якому кожний елемент генеральної сукупності має ненульову ймовірність потрапити у вибірку. При неімовірнісний семплінгу у вихідній множині існує хоча б один елемент який може не потрапити у вибірку взагалі, що зробить підмножину не репрезентативною, тому об'єкти обираються не випадково [2].

Розглянемо види ймовірнісного семплінгу. Простий ймовірнісний семплінг – це випадковий вибір предметів з усієї сукупності. Систематичний семплінг є різновидом випадкової вибірки, впорядкованої за будь-якою ознакою: перший елемент відбирається випадково, потім, з кроком n відбирається кожен k -ий елемент семплу. Пропорційний семплінг – такий, що у вибірці представлені різні групи об'єктів за певним параметром в тих самих пропорціях, що і серед генеральної сукупності. У кластерному семплінгу одиницями семплу виступають не об'єкти, а кластери, сформовані за певним параметром. Стратифікований семплінг дозволяє на основі загального фактору створювати підмножини набору даних, а семпли збираються випадковим чином з кожної підгрупи.

При семплінгу, що базується на неімовірнісному підході, вибірку даних визначають на основі аналітичних суджень. Такий вид відокремлення вибірки використовується при зборі інформації шляхом соціологічних опитувань і не може бути застосованим до існуючого набору даних.

ВИСНОВКИ

Семплінг – метод дослідження множини шляхом аналізу її підмножин. Ця методика допомагає зменшити дані під час початкового та кінцевого аналізу даних. Це може бути надзвичайно корисно, якщо обробка всієї сукупності даних є дорогою або забирає багато часу. Семплінг працює, коли семпл є репрезентативним для всього набору даних. Отже, якщо вибірка буде обрана невдало, це буде безпосередньо відображено у кінцевому результаті. Тому існує безліч стратегій, які допомагають виконувати семплінг якісно, залежно від потреби та ситуації. Таким чином, розробка програмного забезпечення, яке автоматизує різні стратегії семплінгу, є актуальною задачею.

ЛІТЕРАТУРА

1. Айвазян С. А. Прикладна статистика: основи моделювання та попередня обробка даних / С. А. Айвазян, Л. Д. Мешалкин, И. П. Єнюков. – М: Фінанси і статистика, 1987.
2. Deming W. Some Theory of Sampling / W. Deming., 2010. – 640 с. – (Dover Publications).

*Юшкевич.С.В ,
Державний університет «Житомирська Політехніка»
к.к.п, доцент Вакалюк.Т.І,
Державний університет «Житомирська політехніка»*

ЗАГРОЗИ ЯКІ ВИНИКАЮТЬ ПРИ ВПРОВАДЖЕННІ ШТУЧНОГО ІНТЕЛЕКТУ

Комп'ютерні технології з'явилися відносно недавно і здавалось би за короткий час їх існування та розвитку ,вони не повинні були б сильно змінитись. Але зрозуміло ,що це не так – ХХ і ХХІ століття стали надзвичайно продуктивними саме в розвитку науки і не останнє місце в цій системі займають комп'ютерні технології.

Звичайно сучасні комп'ютери мало нагадують своїх громіздких предків – вони значно потужніші і менші за розміром ,що робить їх зручнішими ,а головне ефективнішими. Тепер комп'ютер може поміститись в долоні і здатен виконувати не тільки розрахунки ,а й програвати відео і аудіо файли ,виконувати різноманітні програмні дії ,обробляти великі масиви даних ,розпізнавати зображення і текст ,а також багато інших різноманітних задач[2].

Але найбільш проблемною на даний час являється область розробки і використання штучного інтелекту. Пов'язано це з тим ,що людство на даний час ще не готове довірити виконання деяких задач машинам. Стосується це в основному сфери медицини (за основу беруться саме оперативні втручання) і оборони ,так як в них необхідна акуратність і обдуманість. Звичайно машини більш ефективні практично у всіх сферах виробництва і послуг ,але на відміну від людини вони мають низку вразливостей і не застраховані від поломок і збоїв.

Здавалось би ,що програмний код одна з найнадійніших і продуманих частин комп'ютера ,але нажаль це в більшості випадків не так. Код котрий пройшов усі тести і проявив надзвичайну ефективність при одних умов, може видати критичну помилку при інших. Особливо це стосується систем штучного інтелекту. Саме тут і виходить на світло основна проблема цього напрямку науки – код нейросистем настільки складний ,що продумати усі ситуації з якими може зіткнутися кінцевий продукт надзвичайно складно ,а допускати машину до складних задач ,котрі потребують ювелірної точності за таких умов ніхто не буде. Звичайно сучасні нейросистеми здатні самонавчатись ,що з часом покращує їх аналітичні дані ,але не варт забувати ,що навіть такий підхід не вирішує проблему вразливості коду.

Наступна проблема на котру хотілось би звернути увагу – вразливість таких систем перед хакерськими і вірусними атаками. І тут штучний інтелект відкриває свою «темну сторону» - за допомогою нього може бути викрадена інформація або нанесена шкода матеріальним цінностям ,чи навіть людям[1]. Як приклад уявимо ,що якась країна (по суті не важливо яка) вирішила використати штучний інтелект в своїй військовій сфері. Звичайно це допоможе зберегти людські життя ,але рівно до того моменту як на цю систему

буде виконана хакерська атака. Це приведе в кращому випадку до втрати контролю над зброєю ,в гіршому – система може атакувати своїх же. Звичайно таку систему можна ізолювати ,але це значно знизить функціональність ,так як подібній системі потрібно аналізувати ситуацію на полі бою ,а значить безпроводні канали зв'язку просто необхідні для її ефективності. Ось ці канали і можуть стати діркою скрізь котру і може просочитись шкідливий код.

Звичайно ,відмовлятись від технологій на основі штучного інтелекту ніхто не буде – за ними майбутнє. Тому вирішення раніше оголошених проблем повинно бути пріоритетним – потрібно якщо не повністю їх анулювати ,то як варіант звести їх чинники до мінімуму. Допомогти в цьому може відповідальне відношення до написання коду для таких систем ,тестування системи за всіма допустимими сценаріями і фільтрація вхідної інформації ,щоб забезпечити захист від зовнішніх загроз. При цьому потрібно враховувати декілька факторів:

- невідомо коли відбудеться атака;
- невідомий тип даних котрий буде передаватись при атаці;
- відсутність інформації про шлях атаки;

Нажаль передбачити коли відбудеться спроба злому системи зазвичай неможливо. Саме тому всі дії спрямовані на її захист повинні бути активними весь час – навіть коли система не використовується. Одним з найефективніших способів захистити штучний інтелект від вилучення файлу з зміненим типом даних (вовка в овечій шкурі) навчити систему розпізнавати такі атаки і приймати дані лише з відповідних джерел.

Остання проблема повністю підпадає під людський фактор відповідальності і вирішення її залежить від компетентності конкретних осіб.

ЛІТЕРАТУРА

1. Тегмарк.М 3.0. Доба штучного інтелекту /Тегмарк.М /432с.
2. С.Остапов «Технології захисту інформації» Остапов.С \2014 – 428с.

*Двигун Д. І.,
Криворізький національний університет
Шаповалова Н. Н.,
ст. викладач, Криворізький національний університет*

ВИКОРИСТАННЯ ПАРАДИГМИ ІМОВІРНІСНОГО ПРОГРАМУВАННЯ В МАШИННОМУ НАВЧАННІ

Досліджено використання імовірнісного програмування у машинному навчанні, як спосіб визначення найбільш інформативними ознак у наборі даних, на основі баєсівського виводу.

У сучасний час, коли використання машинного навчання все більше впроваджується в різні сфери нашого життя і постійно вимагає все кращих результатів, доводиться ретельніше вивчати наявні проблеми в алгоритмах навчання і шляхи їх уникнення. Кількість різних даних, на яких можна навчати алгоритми, зростає, як і складність цих даних. Зазвичай в даних можуть бути присутні шуми, викиди, може бути безліч ознак і може бути складно визначити чи буде обрана ознака інформативною або, навпаки, вона тільки погіршить якість навчання.

Історично одним із способів допомогти у прийнятті рішень в умовах невизначеності було використання імовірнісної системи міркувань. Імовірнісні міркування об'єднують наші знання про ситуацію з законами імовірності для визначення тих неспостережуваних чинників, які мають вирішальне значення для прийняття рішення. Імовірнісне програмування – це спосіб створення систем, які допомагають приймати рішення в умовах невизначеності. Багато повсякденних рішень включають судження у визначенні відповідних факторів, які ми не спостерігаємо безпосередньо. В імовірнісному міркуванні створюється модель, яка збирає всі відповідні загальні знання області в кількісному, імовірнісному вираженні [1].

Застосувати імовірнісне програмування в машинному навчанні можна за допомогою наївного баєсівського алгоритму. Наївний баєсівський алгоритм – це алгоритм класифікації, заснований на теоремі Баєса з припущенням про незалежність ознак. Тобто, він

передбачає, що будь-будь яка ознака в класі не пов'язана з наявністю якоїсь іншої ознаки. Наприклад, фрукт може вважатися яблуком, якщо він червоний, круглий і його діаметр становить близько 8 сантиметрів. Навіть, якщо ці ознаки є залежними одна від одної, або від будь-яких інших ознак, в будь-якому випадку вони вносять незалежний внесок у ймовірність того, що цей фрукт є яблуком. Саме тому алгоритм називається «наївним».

Моделі, засновані на наївному баєсівському алгоритмі, досить прості і корисні у випадку великого набору даних. Не дивлячись на свою простоту, цей метод здатен перевершити деякі складні алгоритми класифікації [2].

Переваги баєсівського підходу:

- наявність апріорного розподілу: можливість кількісної оцінки будь-яких попередніх знань, за відповідними апріорними розподілами;
- кількісна оцінка невизначеності: можливість отримувати повний апостеріорний розподіл про те, наскільки ймовірні різні значення.

Теорема Баєса надає спосіб перегляду існуючих передбачень або теорій (ймовірностей поновлення) з урахуванням нових або додаткових доказів. У фінансах теорема Баєса може бути використана для оцінки ризику кредитування потенційних позичальників.

Теорема Баєса заснована на включенні попередніх ймовірнісних розподілів для генерації апостеріорних ймовірностей. У Баєсовому статистичному виводі апріорна ймовірність – це ймовірність події до збору нових даних. Апостеріорна ймовірність – це переглянута ймовірність події, що відбулася після врахування нової інформації. Апостеріорна ймовірність розраховується шляхом поновлення апріорної ймовірності з використанням теореми Баєса:

$$P(\theta|y) = \frac{P(y|\theta)P(\theta)}{P(y)}, \quad (1)$$

де $P(\theta)$ – апріорна ймовірність гіпотези θ , висловлює наші знання про значення параметрів, що передують побудові моделі; $P(y|\theta)$ – апостеріорна ймовірність, являє собою розподіл ймовірностей для параметра в нашій моделі, розраховане з урахуванням апріорних знань і правдоподібності нових; $P(\theta|y)$ – правдоподібність, ймовірність спостерігати дані у при істинності гіпотези θ ; $P(y)$ –

свідощтво, повна ймовірність гіпотези θ . Іншими словами, ймовірність спостереження даних, усереднених по всіх можливих значеннях, які можуть приймати параметри.

Суть формули в тому, що вона дозволяє переставити причину і наслідок: за відомим фактом події обчислити вірогідність того, що вона відбулася за певної причини. Цю формулу також називають формулою зворотної ймовірності [3].

ВИСНОВКИ

Використання статистичного баєсівського виводу дає можливість точно визначити які параметри будуть більш інформативними для алгоритму навчання, на основі ймовірностей кожної ознаки. Також перевагою цього методу є те, що він надає спосіб поновлення ймовірностей.

ЛІТЕРАТУРА

1. Pfeffer A. Practical Probabilistic Programming / Avi Pfeffer., 2016. – 456 с.
2. Bayes' Theorem Definition [Електронний ресурс] – Режим доступу до ресурсу: <https://www.investopedia.com/terms/b/bayes-theorem.asp>.
3. Байесовская статистика и вероятностное программирование на Python [Електронний ресурс] – Режим доступу до ресурсу: <https://nagornyy.me/courses/data-science/bayes/#vviedieniie-v-baiiesovskii-statistichieskii-vyvod>.

*Богун М. С.,
Криворізький національний університет
Шаповалова Н. Н.,
ст. викладач, Криворізький національний університет*

МОДЕЛЬ ВИЯВЛЕННЯ ШАХРАЙСТВА З КРЕДИТНИМИ КАРТКАМИ

Розроблено програмний модуль банківської системи безпеки, здатний виявляти шахрайство з кредитними картками. Модуль

побудовано на основі методу машинного навчання — виявлення аномалій за допомогою якого відбувається пошук аномальної закономірності поведінки при роботі з банківськими картками.

Проблема шахрайських операцій з банківськими картками стає все більш актуальною для нашої країни, на думку асоціації Української міжбанківської асоціації учасників платіжної системи, через шахраїв українці тільки в 2018 році втратили 250 млн. грн., а роком раніше — близько 670 млн. грн. НБУ повідомляє, що відсоток пограбувань від загального обсягу всіх операцій збільшився приблизно на третину.

Щоб запобігти зловживанню банківськими рахунками клієнтів, необхідно виявляти будь-які незвичайні схеми використання кредитних карт (аномалії), відстежуючи шаблони використанні клієнтами і виявляючи будь-які відхилення від попередніх моделей використання.

Виявлення аномалій — це метод, для знаходження незвичайних патернів, які називаються викидами. Вони проявляють себе нестандартно порівняно з нормальними діями, які відбувалися в минулому.

Відхилення від норми можуть бути класифіковані як точкові аномалії (один екземпляр даних є аномальним якщо він занадто далеко від інших) і контекстуальні аномалії (аномалія залежить від контексту, цей тип часто зустрічається в даних часових рядів і колективних аномаліях).

Досліджувана проблема належить до класу проблем навчання на прикладах (навчання з учителем). У кожному випадку використання є пара "об'єкт — відповідь". Необхідно знайти функціональну залежність відповідей від описів об'єктів і побудувати алгоритм, який приймає опис об'єкта на вході і дає відповідь на виході.

Пропонується вирішити проблему класифікації за допомогою методу нейронних мереж. Нейронні мережі — поширене рішення класифікаційних проблем. Штучна нейронна мережа — математична модель, побудована за принципом організації та функціонування біологічних нейронних мереж, являє собою систему зв'язаних та взаємодіючих простих процесів.

Необхідно визначити архітектуру нейронної мережі. Необхідно визначити такі параметри, як кількість мережевих шарів, кількість нейронів кожного шару та функції активації, які будуть використовуватися.

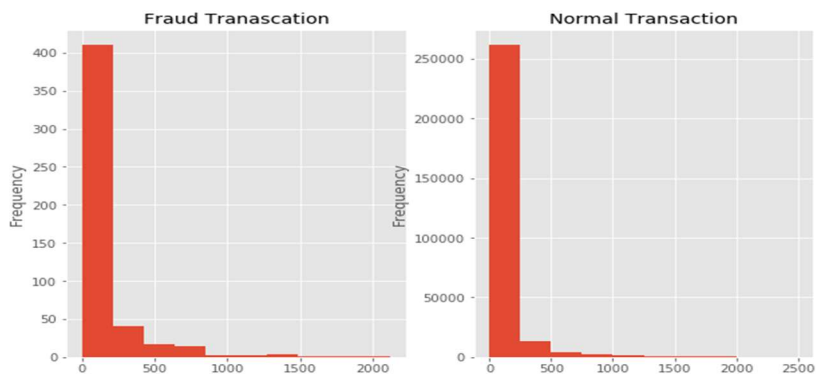


Рис.1 Розподіл звичайних та шахрайських операцій

Не було розроблено методики визначення кількості прихованих шарів та кількості нейронів у них. На практиці ці параметри експериментально визначаються шляхом аналізу якості наближення, наданого мережами різної величини. Таким чином, була обрана мережа з вхідним шаром 38 входів, одним прихованим шаром з 30 нейронами та 2 виходами.

На вході нейрона у нас є вектор параметрів. Це дані транзакції кожного клієнта, представлені в числовій формі(1).

$$X^i = \{x_1^i, x_2^i, \dots, x_n^i\} \quad (1)$$

У цьому випадку кожен клієнт відповідає за клас Y^i . Всього існує два класи набору Y з наступними значеннями: 1 — нормальний, 0 — шахрайство (див. Рис. 1). Нейронна мережа повинна знайти оптимальне розділення гіперповерхні в векторному просторі, розмірність якої буде відповідати кількості ознак. Навчання нейронної мережі, в цьому випадку, полягає в тому, щоб знайти такі значення (коефіцієнти) вагової матриці, в яких нейрон, відповідальний за клас, буде давати значення, близькі до одиниці у випадках нормального поведінки, і значення близькі до нуля в іншому випадку.

Навчання нейронної мережі буде відбуватися шляхом збільшення ваг для кожної ознаки на основі результатів, отриманих з попередніх уявлень даних. Для корекції ваг вибирається метод зворотного поширення помилки. Докладніше про цей метод див. [1].

Експериментально визначено, що для вибору оптимальних ваг для даної нейронної мережі потрібно 74 епох. 284 807 записів навчальної вибірки обробляються для кожної епохи.

Точність розрахунків на випробувальному зразку становить 96,4%.

ВИСНОВКИ

У процесі дослідження проблеми розробки програмного модуля для системи безпеки банку з використанням нейронних мереж проаналізовано сучасний стан використання методів виявлення аномалій і розроблений модуль для виявлення шахрайських транзакцій з кредитними картами.

ЛІТЕРАТУРА

1. Флах Петер.: Машинное обучение. Наука и искусство построения алгоритмов, которые извлекают знания из данных. / пер. с англ. Слинкин А. А. – М.: ДМК Пресс, 2015. – 400 с.

*Горбань В. К.,
Полтавська державна аграрна академія
Науковий керівник: Калініченко О.В.,
к.е.н., доцент, Полтавська державна аграрна академія*

ПЕРСПЕКТИВИ ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ

Висвітлено перспективи розвитку штучного інтелекту у різних галузях, а його також вплив на національну економіку і суспільство цілому.

Зважаючи на те, скільки часу людина проводить у гаджетах – світ уже поневолений технологіями. Але, деякі вчені серйозно обговорюють можливість повстання машин і появи штучного інтелекту. Просунутий інтелектуальний комп'ютер може набути свідомості та здогадатися, що саме людина є головною причиною всіх бід на Землі. Попри те, що такий сценарій не буде позитивним для

долі людства, сучасні дослідники намагаються зробити так, щоб нагородити комп'ютери дедалі більшою кількістю людських рис [1].

В енциклопедіях, штучний інтелект визначають як “здатність комп'ютера виконувати завдання, які можуть виконувати розумні істоти”. Великою мірою, розробники намагаються навчити комп'ютер інтелектуальних здібностей наразок визначення сенсів, узагальнення, навчання на помилках і навіть міркування [2].

Вперше термін “штучний інтелект” використав Джон Маккарті в 1956 році. Його команда є розробником програми для англійського комп'ютера Ferranti Mark 1, яка могла грати в шахи. Варто визначити, що штучний інтелект – надзвичайно узагальнене поняття і, в ідеалі, – це штучно розроблена система, яка має людські або близькі до людських інтелектуальні здібності та може виконати будь-яке завдання з можливих для homo sapiens [3].

Штучний інтелект сьогодні – це здатність машин і програм аналізувати отриману інформацію, робити висновки, приймати на їхній основі рішення. Потенціал застосування штучного інтелекту дуже широкий, вже зараз він використовується у багатьох сферах: медицина, фінанси, промисловість, торгівля і, звичайно, побут людини. Як приклад – голосові помічники Siri і Alexa, яких можна завантажити на iOS, Android або Windows. Існують і автоматичні перекладачі, а також цілісні комплексні системи розумного будинку.

Неабияку роль відіграє штучний інтелект у роботі підприємств. Він допомагає автоматизувати процеси, які потребують зусиль, і тоді участь людини залишається мінімальною. Наприклад, компанія LG планує відкрити завод, де все – від закупівлі матеріалів до відвантаження готового продукту – буде під контролем розумних систем. Запустити виробництво нового формату керівництво компанії збирається вже у 2023 році [3].

Активно застосовується штучний інтелект в робототехніці. Наприклад, в китайській провінції Хенань чергує робот-патрульний, який вміє розрізняти і аналізувати осіб. Один з розумних роботів на ім'я Софія навіть отримав громадянство в Саудівській Аравії. Перспективами штучного інтелекту є: підвищення ефективності, зручність, позбавлення довготривалих процесів і автоматизація звичних. Поки що порівняно новий напрям стикається з низкою труднощів щодо впровадження рішень в життя [1].

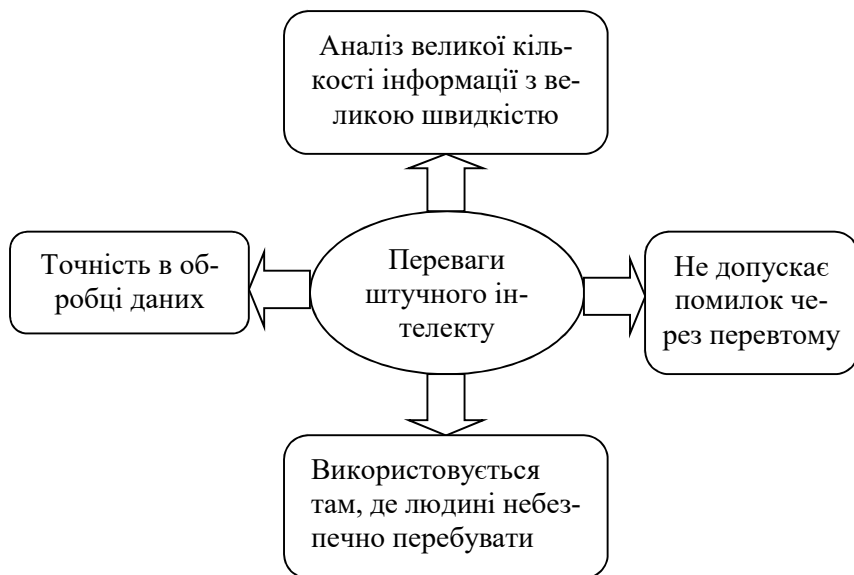


Рис. 1. Переваги штучного інтелекту

Джерело: розроблено автором

ВИСНОВКИ

Отже, сьогоdnішній штучний інтелект – це слабкий або спеціалізований інтелект, який хоч і може обіграти вас у шахи, керувати автомобілем, розпізнавати обличчя, виконувати ваші запити, але все-таки не може замінити людину в будь-якій когнітивній роботі. Тому, на цьому етапі, розмови про те, що штучний інтелект незабаром набуде свідомості, прийматиме рішення проти людської волі – це не більше, ніж наукова фантастика. Сьогодні нам скоріше треба боятися неправильної роботи інтелектуальних систем, які керують літаками, автомобілями, енергомережами або кардіостимуляторами.

ЛІТЕРАТУРА

1. Створити Бога. Навіщо ми розробляємо штучний інтелект і чим нам це загрожує. 2014-2020, ООО “ИЗДАТЕЛЬСКИЙ ДОМ “МЕДИА-ДК” URL: <https://nv.ua/ukr/techno/popscience/gotovimsya-k-matrice-cho->

- takeo-iskusstvennyu-intellekt-i-porabotit-li-on-chelovechestvo-50053922.html (дата звернення 17.02.2020)
2. Як діє штучний інтелект і перспективи його використання. 2020 AI Conference Kyiv. URL: <https://aiconference.com.ua/uk/news/printsipi-raboti-iskusstvennogo-intellekta-i-perspektiva-ego-ispolzovaniya-92238> (дата звернення 17.02.2020)
 3. Як прогресує штучний інтелект: звіт про останні досягнення. 2005-2019, Економічна правда. URL: <https://www.epravda.com.ua/publications/2019/07/15/649648/> (дата звернення 17.02.2020)

*Чорний Д.О., Штуца О.В.
Херсонський національний технічний університет
Шерстюк В.Г.
д.т.н., професор,
Херсонський національний технічний університет*

ШТУЧНІ НЕЙРОННІ МЕРЕЖІ СЬОГОДНІ

Досліджено можливості нейронних мереж на сучасному етапі розвитку, розглянуто деякі сфери людського життя, де застосування нейронних мереж є найбільш успішним.

Сучасність демонструє багато вражаючих можливостей штучних нейронних мереж: мережу навчили перетворювати текст в фонетичне уявлення, яке потім за допомогою вже інших методів перетворювалося в мову; інша мережа може розпізнавати рукописні букви; сконструйована система стиснення зображень, заснована на нейронній мережі і т.д [1]. Нейромережі активно використовують у різних сферах життя суспільства, де вони приносять неабияку користь.

Команда дослідників з Ноттінгемського університету зробила чотири алгоритми машинного навчання для оцінки ступеня ризику серцево-судинних захворювань пацієнтів. Для навчання використовувалися дані 378 тис. англійських пацієнтів. Навчений штучний інтелект визначав ризик кардіологічних захворювань ефективніше реальних лікарів.

Нейромережа, розроблена Марком Уоллером з Шанхайського Університету, спеціалізується на розробці синтетичних молекул. Алгоритм склав шестістадійний синтез похідного бензопіран сульфонаміду (необхідний при лікуванні Альцгеймера) всього за 5,4 секунди.

Японська страхова компанія Fukoku Mutual Life Insurance уклала контракт з IBM. Згідно з ним, 34 співробітників японської компанії замінить система IBM Watson Explorer AI. Нейромережа буде переглядати десятки тисяч медичних сертифікатів і враховувати число відвідувань госпіталів, перенесені операції та інші фактори для визначення умов страхування клієнтів. У Fukoku Mutual Life Insurance впевнені, що використання IBM Watson підвищить продуктивність на 30% і окупиться за два роки.

Машинне навчання допомагає розпізнавати потенційні випадки шахрайства в різних сферах життя. Подібний інструмент використовує, наприклад, PayPal - в рамках боротьби з відмиванням грошей компанія порівнює мільйони транзакцій і виявляє серед них підозрілі. В результаті, шахрайські транзакції в PayPal складають рекордно низькі 0,32%, тоді, як стандарт у фінансовому секторі — 1,32%.

Безпілотні автомобілі - концепт, над яким працює більшість великих концернів, а також технологічні компанії (Google, Uber, Яндекс та ін.) і стартапи, в своїй роботі спираються на нейромережі. Штучний інтелект відповідає за розпізнавання навколишніх об'єктів - будь то інший автомобіль, пішохід або інша перешкода.

Штучний інтелект істотно поліпшив механізми рекомендацій в онлайн-магазинах і сервісах. Алгоритми, засновані на машинному навчанні, аналізують вашу поведінку на сайті і порівнюють його з мільйонами інших користувачів. Все для того, щоб визначити, який продукт ви купите з найбільшою ймовірністю.

Інженери Microsoft спільно з вченими з ICRISAT застосовують штучний інтелект, щоб визначити оптимальний час посіву в Індії. Додаток, що використовує Microsoft Cortana Intelligence Suite, також стежить за станом ґрунту і підбирає необхідні добрива. Спочатку в програмі брало участь всього лише 175 фермерів з 7 сіл. Вони почали посів лише після відповідного SMS повідомлення. В результаті, вони зібрали врожаю на 30-40% більше, ніж зазвичай.

Штучні нейронні мережі запропоновані для різних задач. Потенційними є ті, де людський інтелект малоефективний, а звичайні обчислення трудомісткі або неадекватні [2].

ВИСНОВКИ

Слід підкреслити, що ніяка з сьогоднішніх мереж не є панацеєю, всі вони мають обмеження в своїх можливостях навчатися. Ми маємо справу з областю, що продемонструвала свою спроможність, має унікальні потенційні можливості, але багато обмежень і безліч відкритих питань. Для поліпшення існуючих мереж потрібно багато ґрунтовної роботи. Повинні бути розвинені нові технології, поліпшені існуючі методи і розширені теоретичні основи, перш ніж дана область зможе повністю реалізувати свої потенційні можливості. Також перш ніж нейронні мережі використовувати там, де поставлені на карту людське життя або цінне майно, повинні бути вирішені питання, пов'язані з їх надійністю [1].

ЛІТЕРАТУРА

1. Ф. Уоссермен. Нейрокомп'ютерна техніка: теорія і практика, переклад Ю. А. Зуєв, В. А. Точенов.-1992 .
2. Нейромережа: як штучний інтелект допомагає в бізнесі і в житті [Електронний ресурс] - Режим доступу до ресурсу <https://habr.com/ru/post/337870/>.

Romanenko V.A.
Kryvyi Rih
National University
Krivoulya G.F.
Doctor of technical science,
professor
Kharkiv National University of Radio
Electronics

SOFTWARE AND HARDWARE OF COMPUTER INTELLECTUAL CONTROL FOR MOBILE ROBOT MANIPULATOR

A trend in today's mobile robotics is to perform the control of mobile robots in a distributed fashion. This control scheme requires a wireless communication network connecting the robot to the other existing devices in the environment. Wi-Fi and Bluetooth are examples of such networks. Ad-hoc networks such as wireless sensor networks (WSN) can be deployed instantly and offer a wide coverage by employing a large number of communicating nodes. In addition to provide a communication link to the robots, a WSN can aid the robot's navigation, localization, and also enhance its sensory capabilities.

Recent developments in the artificial intelligence (AI) and robotics has enabled technical objects to be integrated into the business process to help stimulate the workers' imaginations and improve their output. The robotics cannot substitute professionals but it can help perfect whatever process it's initiated into reducing instances of human error thereby saving resources, work hours and energy.

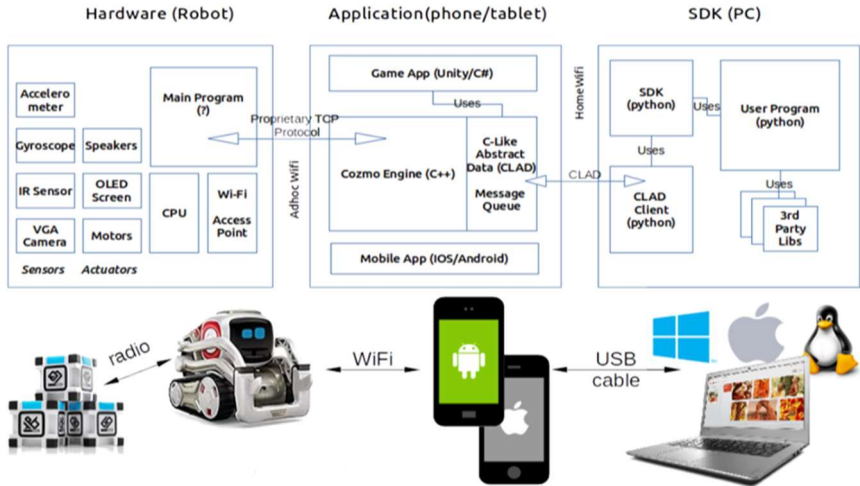
In this paper, we study a mobile robot Cozmo as an example. Robot Cozmo is new, it was first shipped in mid-October 2016, accompanied by an open source python SDK[1]. More features are still being developed by various users and tech-startups, this robot will allow you to enter directly into the world of human interaction with the robot and AI. Cozmo explores the environment, learns, plans its actions, and reacts like a living creature. Cozmo's little mobile robot was designed by former Pixar animators. Using the Code Labs, a simple and intuitive visual programming language based on 'Scratch Blocks', programmers will be

able to develop human behaviors and third-party applications using their Python SDK.

The main purpose of this work is to utilize Cozmo's capabilities to explore a new means of control of robots, the inner workings of machine learning and artificial intelligence, hands-on experience in the aforementioned fields is very paramount to creativity, using the scripting language 'python' Cozmo can be programmed to do pretty much anything. Due to Cozmo's more elementary audience the python code is translated into a simpler programming language 'Scratch blocks' which was developed by MIT enabling users to understand Cozmo's functionalities. These functionalities are wrapped up in graphical blocks of code that students drag and drop onto the application's interface, the reason behind this is to guide students through the most basic coding concepts and lead them to advanced languages like python and Cowing to its Wall-E inspired personality, the earlier students can start making use of the functionalities the easier it will be for them to really understand how these innovative technologies operate.

In terms of locomotion, Cozmo as a robot is not immune to the generic problems that other robots possess: The Path planning, the Search problem, path planning and the search problem. The techniques students employ to make cozmo maneuver these generic problems will be explained and plays a huge part in the educational process as the student is able to have a hands-on experience and well informed approach to locomotion in robots.

There are problems with some affecting interactive operation and others related to locomotion in controlling robots some of which are highlighted below, one of the aims is to solve some of these problems using Cozmo as a test robot, and the concepts could be applied to some other prototype in other fields such as steering systems in autonomous vehicles and drones. Cozmo structure is shown in Figure.



Img. 1. Cozmo Architecture

This robot allow work with real hardware in order to better apply these concepts to help alleviate the challenges of the society, they provide a brief but intensive treatment of a cross-section of the key elements of Robotics, Robot vision, AI and cognitive science[2].

Cozmo as a robot is not immune to the generic problems that other robots possess: The Path planning, the Search problem, path planning and the search problem. The techniques students employ to make Cozmo maneuver these generic problems will be explained and plays a huge part in the educational process as the student is able to have a hands-on experience and well informed approach to locomotion in robots.

LITERATURE

1. Anki Cozmo - Python SDK. [Electronic resource]. – Access mode <https://github.com/anki/cozmo-python-sdk>
2. Adrian Rosebrock - Deep Learning for Computer Vision with Python, 2017.

*Прачик В.В.,
магистр, Херсонский национальный технический университет
Ляшенко Е.Н.
к.т.н., доцент, Херсонский национальный технический универси-
тет*

МОДЕЛИРОВАНИЕ ВЗАИМОДЕЙСТВИЯ ИНТЕЛЛЕКТУАЛЬНЫХ АГЕНТОВ В РАСПРЕДЕЛЕННЫХ МУЛЬТИАГЕНТНЫХ СИСТЕМАХ

В работе рассмотрены семантические характеристики коммуникативных действий интеллектуальных агентов в распределенных мультиагентных системах, а также их формальные модели.

В агентно-ориентированных технологиях построения распределенных мультиагентных систем важное место занимает исследование вопросов взаимодействия (коммуникации) интеллектуальных агентов.

Это связано с тем, что отдельные агенты имеют лишь частичное представление об общей задаче и способны решать лишь некоторые ее подзадачи. Поэтому для решения сложных практических задач, как правило, требуется организация взаимодействия агентов, образующих распределенную мультиагентную систему. Однако, именно процессы взаимодействия интеллектуальных агентов являются наименее изученными и исследованными в контексте построения распределенных мультиагентных систем, что объясняется практическим отсутствием унифицированных моделей участников процесса коммуникации, моделей коммуникативных сред, моделей диалогов, а также сценариев диалогов между участниками взаимодействия [1,2].

Таким образом, предметом настоящего исследования являются:

– модели участников взаимодействия, определяющие структуру распределенной мультиагентной системы в зависимости от таких параметров как совместимость целей интеллектуальных агентов, наличие взаимных обязательств, необходимость привлечения недостающего опыта и ограниченность ресурсов.

- модели диалогов интеллектуальных агентов, включающие протоколы взаимодействия и сценарии диалогов между участниками взаимодействия, которые определяются спецификой и особенностями предметной области.

- языки моделирования взаимодействия интеллектуальных агентов.

Протоколы взаимодействия в распределенных мультиагентных системах определяют последовательность коммуникативных действий (актов) интеллектуальных агентов, объединенных общей задачей и ситуативными условиями. В ходе коммуникативного акта, как минимум двух интеллектуальных агентов, происходит обмен информацией, формализованной в виде сообщений на определенном языке коммуникации [1].

В процессе коммуникации осуществляется воздействие интеллектуальных агентов друг на друга, вырабатывается общность в понимании ситуации и определенная степень согласия между ними. Процесс коммуникации интеллектуальных агентов осуществляется в форме диалога и возможен только при наличии определенной коммуникативной среды.

Таким образом, при разработке протоколов взаимодействия первостепенное значение имеют три аспекта [1,2]:

- наличие коммуникативной среды для осуществления процесса коммуникации агентов.

- наличие общего языка коммуникации, который может быть однозначно интерпретирован всеми интеллектуальными агентами, участвующими в коммуникативном процессе.

- наличие системы онтологий, которая позволит избежать проблемы семантической интероперабельности, под которой понимается способность любых взаимодействующих в процессе коммуникации интеллектуальных агентов одинаковым образом понимать смысл информации, которой они обмениваются.

Семантические характеристики коммуникативных действий интеллектуальных агентов, а также их формальные модели представлены в таблице 1 [3].

Таблица 1- Семантические характеристики коммуникативных действий интеллектуальных агентов и их формальные модели

Семантическая характеристика коммуникативного действия	Формальная модель
Accept Proposal	$\langle i, \text{accept-proposal} (j, \langle j, \text{act} \rangle, \varphi) \rangle \equiv$ $\langle i, \text{inform} (j, I i \text{ Done} (\langle j, \text{act} \rangle, \varphi)) \rangle$ FP: $B i \alpha \wedge \neg B i (B i f j \alpha \vee U i f j \alpha)$ RE: $B j \alpha$ Where: Example $\alpha = I i \text{ Done} (\langle j, \text{act} \rangle, \varphi)$
Agree	$\langle i, \text{agree} (j, \langle i, \text{act} \rangle, \varphi) \rangle \equiv$ $\langle i, \text{inform} (j, I i \text{ Done} (\langle i, \text{act} \rangle, \varphi)) \rangle$ FP: $B i \alpha \wedge \neg B i (B i f j \alpha \vee U i f j \alpha)$ RE: $B j \alpha$ Where: $\alpha = I i \text{ Done} (\langle i, \text{act} \rangle, \varphi)$
Cancel	$\langle i, \text{cancel} (j, a) \rangle \equiv$ $\langle i, \text{disconfirm} (j, I i \text{ Done} (a)) \rangle$ FP: $\neg I i \text{ Done} (a) \wedge B i (B j I i \text{ Done} (a) \vee U j I i \text{ Done} (a))$ RE: $B j \neg I i \text{ Done} (a)$
Call for Proposal	$i, \text{cfp} (j, \langle j, \text{act} \rangle, \text{Ref } x \varphi(x)) \rangle \equiv$ $\langle i, \text{query-ref} (j, \text{Ref } x (I i \text{ Done} (\langle j, \text{act} \rangle, \varphi(x)), (I j \text{ Done} (\langle j, \text{act} \rangle, \varphi(x)))) \rangle$ FP: $\neg B \text{ref } i (\text{Ref } x \alpha(x)) \wedge \neg U \text{ref } i (\text{Ref } x \alpha(x)) \wedge \neg B i I j \text{ Done} (\langle j, \text{inform-ref} (i, \text{Ref } x \alpha(x)) \rangle)$ RE: $\text{Done} (\langle j, \text{inform} (i, \text{Ref } x \alpha(x) = r l) \rangle \mid \dots \mid \langle j, \text{inform} (i, \text{Ref } x \alpha(x) = r k) \rangle)$ Where: $\alpha(x) = I i \text{ Done} (\langle j, \text{act} \rangle, \varphi(x))$ $I j \text{ Done} (\langle j, \text{act} \rangle, \varphi(x))$
Confirm	$\langle i, \text{confirm} (j, \varphi) \rangle$

	FP: $B i \varphi \wedge B i U j \varphi$ RE: $B j \varphi$
Disconfirm	$\langle i, \text{disconfirm}(j, \varphi) \rangle$ FP: $B i \neg \varphi \wedge B i (U j \varphi \vee B j \varphi)$ RE: $B j \neg \varphi$
Failure	$\langle i, \text{failure}(j, a, \varphi) \rangle \equiv$ $\langle i, \text{inform}(j, (\exists e) \text{Single}(e) \wedge \text{Done}(e), \text{Feasible}(a) \wedge$ $I i \text{Done}(a)) \wedge \varphi \wedge \neg \text{Done}(a) \wedge \neg I i$ $\text{Done}(a) \rangle$ FP: $B i \alpha \wedge \neg B i (B i f j \alpha \vee U i f j \alpha)$ RE: $B j \alpha$ Where: $\alpha = (\exists e) \text{Single}(e) \wedge \text{Done}(e), \text{Feasible}(a) \wedge I i \text{Done}(a) \wedge \varphi \wedge$ $\neg \text{Done}(a) \wedge \neg I i \text{Done}(a)$
Inform	$\langle i, \text{inform}(j, \varphi) \rangle$ FP: $B i \varphi \wedge \neg B i (B i f j \varphi \vee U i f j \varphi)$ RE: $B j \varphi$
Inform If	$\langle i, \text{inform-if}(j, \varphi) \rangle \equiv$ $\langle i, \text{inform}(j, \varphi) \rangle \langle i, \text{inform}(j, \neg \varphi) \rangle$ FP: $B i f i \varphi \wedge \neg B i (B i f j \varphi \vee U i f j \varphi)$ RE: $B i f j \varphi$
Inform Ref	$\langle i, \text{inform-ref}(j, \text{Ref } x \delta(x)) \rangle \equiv$ $\langle i, \text{inform}(j, \text{Ref } x \delta(x) = r l) \rangle \dots $ $\langle i, \text{inform}(j, \text{Ref } x \delta(x) = r k) \rangle$ FP: $B r e f i \text{Ref } x \delta(x) \wedge \neg B i (B r e f j \text{Ref } x \delta(x) \vee U r e f j \text{Ref } x \delta(x))$ RE: $B r e f j \text{Ref } x \delta(x)$
Not Understood	$\langle i, \text{not-understood}(j, a, \varphi) \rangle \equiv$ $\langle i, \text{inform}(j, \alpha) \rangle$ FP: $B i \alpha \wedge \neg B i (B i f j \alpha \vee U i f j \alpha)$ RE: $B j \alpha$ Where: $\alpha = \varphi \wedge (\exists x) B i ((\exists e) \text{Done}(e) \wedge \text{Agent}(e, j) \wedge B j (\text{Done}(e) \wedge$ $\text{Agent}(e, j) \wedge (a = e))) = x$
Propose	$\langle i, \text{propose}(j, \langle i, \text{act} \rangle, \varphi) \rangle \equiv$

	$\langle i, \text{inform}(j, I j \text{ Done}(\langle i, \text{act} \rangle, \varphi)) \rangle$ $I i \text{ Done}(\langle i, \text{act} \rangle, \varphi) \rangle$ FP: $B i \alpha \wedge \neg B i (Bif j \alpha \vee Uif j \alpha)$ RE: $B j \alpha$ Where: $\alpha = I j \text{ Done}(\langle i, \text{act} \rangle, \varphi)$ Example $I i \text{ Done}(\langle i, \text{act} \rangle, \varphi)$
Query If	$\langle i, \text{query-if}(j, \varphi) \rangle \equiv$ $\langle i, \text{request}(j, \langle j, \text{inform-if}(i, \varphi) \rangle) \rangle$ FP: $\neg Bif i \varphi \wedge \neg Uif i \varphi \wedge \neg B i I j \text{ Done}(\langle j, \text{inform-if}(i, \varphi) \rangle)$ RE: $\text{Done}(\langle j, \text{inform}(i, \varphi) \rangle \langle j, \text{inform}(i, \neg \varphi) \rangle)$
Reject Proposal	$\langle i, \text{reject-proposal}(j, \langle j, \text{act} \rangle, \varphi, \psi) \rangle \equiv$ $\langle i, \text{inform}(j, \neg I i \text{ Done}(\langle j, \text{act} \rangle, \varphi) \wedge \psi) \rangle$ FP : $B i \alpha \wedge \neg B i (Bif j \alpha \vee Uif j \alpha)$ RE : $B j \alpha$ Where: $\alpha = \neg I i \text{ Done}(\langle j, \text{act} \rangle, \varphi) \wedge \psi$

ВЫВОДЫ

Рассмотрены семантические характеристики коммуникативных действий интеллектуальных агентов в распределенных мультиагентных системах. Приведены их формальные модели.

ЛИТЕРАТУРА

1. Wooldridge, M. An Introduction to MultiAgent Systems / Michael Wooldridge. – N.Y.: J. Wiley & Sons, 2002. – 366 p.
2. Rao, A.S., and Georgeff, M. P. Modeling rational agents within a BDIarchitecture. KR: Proceedings of the International Conference on Knowledge Representation and Reasoning, 1991. – 473–484 pp.
3. FIPA Communicative Act Library Specification. Foundation for Intelligent Physical Agents, Geneva, Switzerland, 1996-2002. – 18 p.

*Каврин Д. А.,
аспирант, Национальный университет «Запорожская поли-
техника»*

Субботин С. А.

*д.т.н. профессор, Национальный университет «Запорожская по-
литехника»*

ИСПОЛЬЗОВАНИЕ АНСАМБЛЕЙ ДЛЯ РЕДУКЦИИ ОБУЧАЮЩИХ ВЫБОРОК В РАСПОЗНАЮЩИХ МОДЕЛЯХ ПО ПРЕЦЕДЕНТАМ

Решена задача отбора экземпляров для формирования обучающей выборки минимального объема, позволяющей классифицировать данные с точностью не ниже, чем при использовании исходной выборки, при построении распознающих моделей по прецедентам. Предложен новый метод отбора экземпляров с использованием ансамблей базовых классификаторов. Разработано программное обеспечение, реализующее предложенный метод.

В задачах классификации, на сегодняшний день, одним из наиболее успешных направлений увеличения производительности распознающих моделей является использование последовательных и параллельных ансамблей базовых классификаторов [1]. В данной работе предлагается использовать ансамбли классификаторов для сокращения и очистки обучающих выборок при построении распознающих моделей. Предложенный подход открывает новые направления для исследований в области предварительной обработки данных, таких как отбор, прототипирование, редукция экземпляров и признаков, с использованием существующих в задачах классификации методик увеличения производительности.

Существуют несколько разных способов создания ансамблей, которые в основном направлены на достижение максимального разнообразия базовых классификаторов. Разнообразие может достигаться разными методами. В частности, если необходимо построить ансамбль, используя один метод классификации, разнообразие может быть достигнуто варьированием обучающей выборки. Такой

подход называется бэгинг (bagging) [1]. Идея бэгинга состоит в многократном обучении базового классификатора на случайных подвыборках.

Помимо необходимости важным условием обеспечения разнообразия базовых классификаторов является выбор метода классификации, который не должен быть слишком «стабильным», поэтому в работе использовался метод ближайших соседей с одним ближайшим соседом. Случайные подвыборки формировались стохастическим делением исходной выборки на заданное число приблизительно одинаковых по размеру частей (фолдов). Затем все экземпляры исходной выборки классифицировались отдельно с помощью каждого фолда, в качестве обучающей выборки. По результатам классификации рассчитывались весовые характеристики каждого экземпляра w . Новая обучающая подвыборка формировалась из экземпляров, вес которых был выше заданного порогового значения θ . Затем полученная подвыборка классифицировалась с помощью оставшихся экземпляров с целью получения «внутренней» ООВ (out-of-bag) оценки. Для получения наиболее оптимального набора экземпляров, синтезировалось некоторое число новых подвыборок. Подвыборка, получившая максимальную оценку ООВ, использовалась в качестве результирующей обучающей выборки для синтеза распознающей модели по прецедентам.

Оценка эффективности предложенного метода производилась с помощью бинарного набора данных «Banana Data Set» [2], имеющего сложную геометрию расположения классов в пространстве признаков. Данный набор делился случайным образом на обучающую и тестовую выборку в соотношении 75/25 с сохранением баланса классов. Далее с помощью исходной и полученной обучающих выборок синтезировались распознающие модели на основе kNN классификатора.

ВЫВОДЫ

Проведенные эксперименты подтвердили работоспособность, предложенного метода и реализующего его программного обеспечения. При значениях числа фолдов выше 10, метод позволил уменьшить число экземпляров обучающей выборки практически в два раза, при этом точность классификации повысилась. В таблице

1 представлены показатели числа экземпляров и точности классификации для исходной обучающей выборки и выборки полученной с помощью предложенного метода. Данные результаты были получены с использованием следующих параметров: число фолдов – 22, число формируемых подвыборок – 11, пороговое значение $\Theta = 0,9$, классификатор kNN с одним ближайшим соседом. Полученные результаты позволяют рекомендовать предложенный метод предварительной обработки исходных обучающих выборок для синтеза распознающих моделей по прецедентам.

Таблица 1 – Результаты обработки выборки «Banana Data Set»

Набор данных	Число экземпляров	Точность классификации
Исходная выборка	3975	0,8747
Обработанная выборка	2202	0,8928

Дальнейшие исследования в области построения ансамблей для отбора экземпляров могут проводиться в таких направлениях, как разработка методов комбинирования ответов базовых классификаторов, формирование набора нестабильных методов классификации, использование методов параллельных вычислений.

ЛИТЕРАТУРА

1. Kuncheva L. I. Combining pattern classifiers: methods and algorithms / L. I. Kuncheva. – Hoboken: John Wiley & Sons, 2014. – 350.
2. Alcalá-Fdez J. KEEL Data-Mining Software Tool: Data Set Repository, Integration of Algorithms and Experimental Analysis Framework / J. Alcalá-Fdez, A. Fernandez, J. Luengo, J. Derrac, S. García, L. Sánchez, F. Herrera // Journal of Multiple-Valued Logic and Soft Computing. – 2010. – Vol. 17, Issue 4. – P. 255-287.

То Тхі Ха Мі
Студентка групи AI-171
Шпинковський О.А.
к.т.н, доцент каф. інформаційних систем
Одеський національний політехнічний університет

ВИКОРИСТАННЯ НЕЙРОННИХ МЕРЕЖ У КЛАСИФІКАЦІЇ ОБ'ЄКТІВ ЗОБРАЖЕНЬ

У даній роботі розглянута задача класифікації зображень, проведений огляд структури та застосування згорткових нейронних мереж.

Сьогодні штучна нейронна мережа широко використовується для вирішення багатьох проблем, одна з яких – класифікація об'єктів на зображенні [1]. Один з найпоширеніших способів класифікації - на основі описів об'єктів з використанням ознак, в якому кожен об'єкт характеризується набором числових або нечислових ознак. Для класифікації використовують згорткові нейронні мережі (англ. *Convolutional Neural Network – CNN*). Прорив в побудові моделей для класифікації зображень прийшов з відкриттям, того що CNN можна використовувати для поступового вилучення вищого рівня уявлень вмісту зображення [2]. Замість попередньої обробки даних для отримання таких функцій, як текстури і форми, CNN приймає в якості вхідних даних тільки необроблені піксельні дані зображення і «вчиться» витягувати ці функції і, в кінцевому рахунку, робити виведення, з якого об'єкта вони складаються. Для початку, коли система отримує вхідне зображення, їй вже відомий фіксований набір категорій або міток. Це можуть бути будь-які об'єкти. CNN отримує карту вхідних об'єктів - тривимірну матрицю, де розмір перших двох вимірів відповідає довжині і ширині зображень в пікселях [3].

Розмір третього виміру дорівнює трьом (відповідає каналам кольорового зображення: червоному, зеленому і синьому). CNN містить стек модулів, кожен з яких виконує три операції.

1. Згортка
2. Випрямлена лінійна одиниця (*ReLU, Rectified linear unit*)
3. Субдискретизація

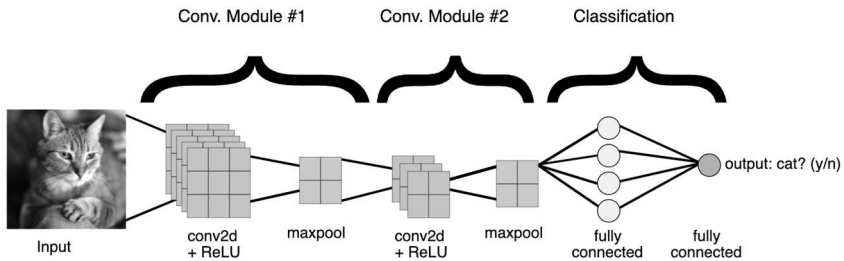


Рис. 1. Наскрізна структура CNN

Згортка витягує листи вхідної карти об'єктів і застосовує до них фільтри для обчислення нових об'єктів, створення вихідної карти об'єктів або згорнутих об'єктів (які можуть мати інший розмір і глибину, ніж карта вхідних об'єктів). Згортки визначаються двома параметрами:

1. Розмір видобутих «плиток» (зазвичай 3x3 або 5x5 пікселів). Плитка – це композиція, яка складена з відносно невеликих фрагментів квадратної форми.
2. Глибина вихідної карти об'єктів, яка відповідає кількості застосованих фільтрів.

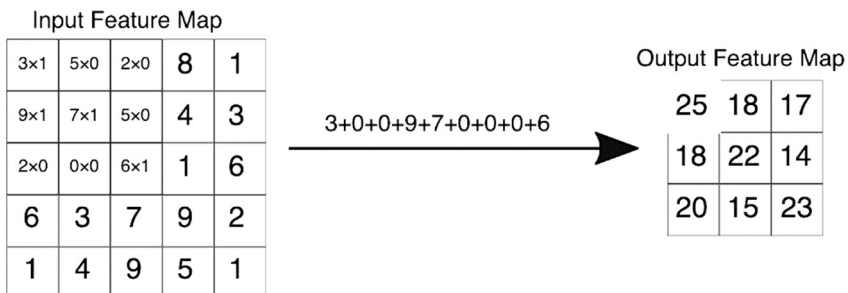


Рис. 2. Розрахунки згорнутої матриці ознак

Переваги у використанні згорткової нейронної мережі наступні:

1. Зменшення кількості початкових параметрів та підвищення швидкості навчання з порівнянням з повно-нейронною мережею.

2. Можливість розпаралелення обчислень і реалізації алгоритмів навчання мережі на графічних процесорах (*GPU*).
3. Стійкість до зсуву позиції об'єкта у вхідних даних. При навчанні *CNN* зсувається по частинах об'єкта. Тому на прикладі навчальних ознак, які залежать від позиції «важливих частин» об'єктів, *CNN* виробляє однакові ознаки для двох зображень, хоча позиції об'єктів в цих зображеннях різні. Таким чином, ця властивість *CNN* допомагає підвищувати якість класифікації.

ВИСНОВКИ

Отже, *CNN* можна застосовувати для вирішення різних типів завдань: класифікації зображень, розпізнавання мови, класифікації текстів тощо. При використанні *CNN* можна істотно зменшувати кількість навчальних параметрів та отримувати високу якість класифікації.

ЛІТЕРАТУРА

1. Шпинковський, О., М. Шпинковська, і В. Голобородько. 2019. Інформаційна система для допомоги фінансовим установам у визначенні кредитоспроможності клієнтів. *Automation of Technological and Business Processes* 11 (3), 14-22. <https://doi.org/10.15673/atbp.v11i3.1496>.
2. Бордан І.С. Сучасні інструменти і технології створення штучного інтелекту / І. С. Бордан, О. А. Шпинковський // Сучасні інформ. технології : матеріали Дев'ятої Міжнарод. наук. конф. студентів та молодих вчених, м. Одеса, 23-24 трав. 2019 р. / Одес. нац. політех. ун-т. – Одеса, 2019. – С. 122–123.
3. ML Practicum: Image Classification. *Introducing Convolutional Neural Networks*. [Електронний ресурс] – Режим доступу: <https://developers.google.com/machine-learning/practica/image-classification/convolutional-neural-networks>

Юкельсон М.В.
Студентка групи AI-171
Шпинковська М.І.

*к.т.н, доцент каф. вищої математики та моделювання систем
Одеський національний політехнічний університет*

ВИРІШЕННЯ ПРОБЛЕМИ ДИСБАЛАНСУ КЛАСІВ В ОБРОБЦІ ДАНИХ

Розглянуто ряд методів, які можна використовувати для обробки незбалансованого набору даних.

Врівноваженість обсягів даних є дуже важливою у застосуванні інтелектуальних інструментів машинного навчання [1]. Незбалансовані дані, як правило, відносять до проблеми в алгоритмах класифікації, коли класи не представлені однаково [2].

Для досягнення бажаного результату в наборах даних машинного навчання з незбалансованими показниками, є сенс прагнути наступного:

- Збирати більшу кількість даних.

Великий набір даних може увявити іншу і, можливо, більш збалансований розподіл на класи.

- Змінювати метрики продуктивності.

1. Перетворювати такі показники продуктивності, які можуть дати більш глибоке розуміння точності моделі, ніж традиційна точність класифікації. Наприклад: матриця плутанини (розбивка прогнозів у таблицю, яка показує вірні прогнози і типи зроблених невірних прогнозів), точність (міра точності класифікаторів), відгук (показник повноти класифікаторів), оцінка F1 (або F-оцінка) та середньозважена точність. Також рекомендовано звернути увагу на наступні показники: Каппа (Каппа Коена) - точність класифікації, нормалізована дисбалансом класів в даних та ROC-AUC [3].

- Створювати синтетичні зразки.

Найбільш простий спосіб генерувати синтетичні вибірки - це випадкова вибірка атрибутів з примірників в класі меншини. Ця зміна називається вибіркою набору даних, і є два основні методи, які можна використовувати для вирівнювання класів:

1. Можливість додати копії примірників з недостатньо представленою класу, який є надмірною вибіркою.

2. Видалити екземпляри з представленого класу, що є недостатньою вибіркою.

Перевага цих підходів у тому, що їх дуже легко реалізувати. Бажано завжди пробувати обидва підходи на всіх незбалансованих наборах даних, для того щоб побачити, чи дає це підвищення бажаних мір точності.

ВИСНОВКИ

Окрім наведених вище методів моделювання незбалансованого набору даних варто спробувати різноманітні алгоритми, општрафовані моделі (штраф-SVM и штраф-LDA) та просто змінити точку зору. Наприклад, зазирнути у такі області досліджень як виявлення аномалії чи змін.

ЛІТЕРАТУРА

1. Шпинковський, О., Шпинковська, М. і Голобородько, В. (2019) ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ ДОПОМОГИ ФІНАНСОВИМ УСТАНОВАМ У ВИЗНАЧЕННІ КРЕДИТОСПРОМОЖНОСТІ КЛІЄНТІВ, Automation of technological and business processes, 11(3), pp 14-22. doi: 10.15673/atbp.v11i3.1496.
2. Прокопович, І. В. Використання інтелектуальних технологій у визначенні діагнозу хвороби / І. В. Прокопович, О. А. Шпинковський // Інформ. системи та технології в медицині (ISM-2018) : І Міжнарод. наук.-практ. конф., 28-30 листоп. 2018 р. : зб. наук. пр. / ХНУРЕ. – Харків, 2018. – С. 127–129.
3. “Классическое машинное обучение: классификация, обобщение, кластеризация” <Електронний ресурс>. Режим доступу: <https://evergreens.com.ua/ru/articles/classical-machine-learning.html>.

Калмикова О.К.
Криворізький національний університет
Кузнєцов Д.І.
к.т.н., доцент, Криворізький національний університет

ПРОГРАМНО-АПАРАТНИЙ КОМПЛЕКС АВТОМАТИЗОВАНОГО ПОЛИВУ НА ОСНОВІ ВИКОРИСТАННЯ КОНЦЕПЦІЇ “ІОТ”

Розглянуто перспективи застосування концепції “ІоТ” та технології “М2М”. Дано опис стільникового модему.

Автополив або розумне зрошення - це сучасне рішення для тих, хто знає як важливо дбати про правильний і своєчасний полив рослин, але не має можливості часто їздити на свою дачну ділянку. Мати гарний сад, квітники або город, який приносить непоганий врожай - гордість для будь-якого дачника або городника-любителя. Влітку погода диктує свої умови і нам доводиться поливати ділянки якомога частіше - від цього залежить і зовнішній вигляд насаджень, і їх майбутній урожай. Зрозуміло, що, поливаючи вручну шлангом, доведеться витратити більшу частину свого вільного часу, але сучасні технології пропонують відміну альтернативу такому поливу - автополив.

Більшість стандартних автоматизованих систем поливу використовують або механічні таймери відкриття кранів або дорогі спеціалізовані пристрої із закритою архітектурою без можливості підключення до інтернету речей (Internet of Things, IoT). [1]

На основі концепції інтернету речей запропоновано нове рішення поставленої проблеми, яке дозволить користувачам керувати процесом зрошення землі віддалено. Проектуєма система дозволить не тільки більш ефективно використовувати воду, але також дуже економити час та отримувати поточну інформацію про стан зрошення.

Логіко-функціональна схема системи зрошення представлена на рис.1. Вона включає в себе наступні елементи:

- Використання 3G/4G бездротового маршрутизатора для організації автоматичної системи іригації

зручно в експлуатації. Система дозволяє підтримувати гарний та квітучий ландшафт з мінімальними втратами води. При цьому, вам немає необхідності викидати ваші вже придбані шланги і насоси. Використання технології межмашинної взаємодії (M2M) [2] робить автополив не застарілий, а, навпаки, більш просунутий і «розумний». Пристрій на базі M2M – стільниковий модем. [3] Стільникові модеми дозволяють організувати безпроводний канал передачі даних. Основна технологія доступу - GSM-стандарт. За допомогою бездротових модемів можна передавати дані по 2G, 3G, 4G / LTE стандартам стільникового зв'язку. Завдяки функціоналу пристроїв можливий віддалений збір інформації з контрольно-вимірювального обладнання.

- IP-камера - gsm відеоспостереження має на увазі дистанційне керування системою. Так користувач системи може управляти відеокамерою шляхом відправки SMS повідомлень з встановленими командами для виконання. Також власник може просто зателефонувати на відеокамеру і прослуховувати що відбувається на об'єкті.
- Датчики вологості ґрунту та температури – призначені для вимірювання показників вологості ґрунту та температури навколишнього середовища
- Контролер arduino - управління поливом є однією з основних частин системи автоматичного поливу. Функція контролера автоматичного поливу полягає у визначенні періодичності, початку часу поливу і тривалості поливу. Контролер управління поливом вимагає індивідуальних налаштувань тривалості і періодичності.



Рис.1 логіко - функціональна схема системи автоматичного поливу

ВИСНОВКИ

Таким чином, можливо встановити оптимальний режим роботи систем поливу на основі вище запропонованих елементів. Також при посушливій погоді зможете встановити, як можна більш частий полив для вашої ділянки. Завойовуючи авторитет на ринку, системи автоматичного поливу дійсно виправдовують усі очікування, які можуть виникнути у споживачів.

ЛІТРАТУРА

1. IT enterprise. Технології та інновації [Електронний ресурс] / IT enterprise – Режим доступу до ресурсу: <https://www.it.ua/ru/knowledge-base/technology-innovation/internet-veschej-internet-of-things-iot>.
2. Перспективи технології межмашинного взаємодія [Електронний ресурс] – Режим доступу до ресурсу: <http://cti.com.ua/products/articles/m2m.html>.
3. GSM відеонаблюдення [Електронний ресурс] – Режим доступу до ресурсу: https://xn--80adageboqrpy5j.com.ua/gsm_video_surveillance

*Мигуль Д. С.,
Криворізький національний університет
Шаповалова Н. Н.,
ст. викладач, Криворізький національний університет*

НЕЙРОМОРФНІ ПРОЦЕСОРИ І СИСТЕМИ

Проаналізовано етапи і напрямки розвитку перспективної технології у галузі штучного інтелекту – створення нейроморфних процесорів. Вивчено їх структуру, а також засоби програмування.

В останні десятиріччя інтенсивно почала розвиватися галузь нейроморфних систем и процесорів. Давайте ж розглянемо, що це таке.

Прототипом нейронних мереж став людський мозок. Фактично нейронна мережа — це безліч нейронів, які з'єднані між собою синапсами. Інформація в нейронній мережі передається за допомогою електричних імпульсів. Нейрони отримують інформацію, здійснюють над нею прості операції і передають по синапсам далі.

Людський мозок на відміну від комп'ютера не може швидко оперувати і обчислювати складні алгоритми, але він має властивість адаптуватися до вирішення завдання. Цією властивістю і повинен бути наділений штучний інтелект.

Перші спроби створити нейронну мережу були зроблені ще в середині минулого століття Тедом Хоффом і Бернардом Уїдроу. Вони створили однорівневу нейромережу. У 1980-их роках розвиток нейромереж було призупинено, але вже з 1990-их років і до сьогодні людство інтенсивно займається розробкою штучного інтелекту[1].

Зараз люди займаються розробкою нейроморфних процесорів. Це процесори, які засновані на принципах роботи людського мозку, вони моделюють нейрони і синапси, а потім передають інформацію за допомогою електричних імпульсів. Створенням нейроморфних процесорів займаються такі компанії як IBM, Intel і менш відомі компанії.

Наприклад, IBM розробила нейроморфний процесор TrueNorth, який використовує 4096 ядер, кожне з яких має 256 ней-

ронів, усього в процесорі приблизно 256 мільйонів синапсів. Чіп містить більш ніж 5 мільярдів транзисторів, а також у кожному ядрі є SRAM-пам'ять і маршрутизатор імпульсів. Основним призначенням даного процесора є розпізнавання інформації. Для програмування процесора була розроблена спеціальна мова Corelet[2].

Intel розробила процесор Loihi, він складається з 130 тисячі нейронів і 130 мільйонів синапсів (наприклад, мозок людини має приблизно 85-86 мільярдів нейронів). Програмувати процесор можна за допомогою мови Python. Декілька років тому Intel створила Спільноту нейроморфних досліджень або Intel Neuromorphic Research Community (INRC), де буде відбуватися майбутній розвиток нейроморфних алгоритмів, програмного забезпечення і різних прикладних програм.

Для застосування Loihi використовують спеціальні плати Intel Nahuku (Рис. 1), які можуть об'єднувати в собі від 8 до 32 таких процесорів[3].

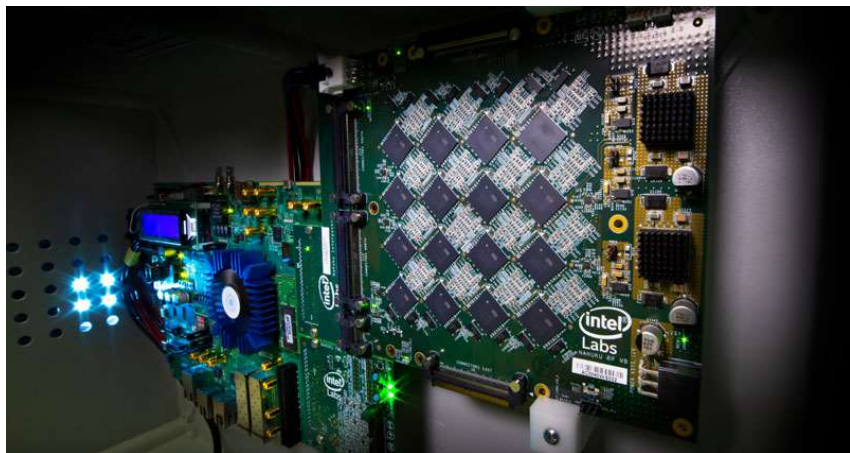


Рисунок 1 – Плата Intel Nahuku

У Китаї був розроблений гібридний нейроморфний процесор Tianji. Чіп складається з 150 ядер. Кожне ядро складається з штучних аналогів синапса, дендрита і аксона, що дозволяє імітувати роботу людського мозку. Процесор адаптований до роботи як класичних нейроморфних мереж, так і до імпульсних. Вчені вважають, що

цей вид нейроморфних процесорів ефективніше, ніж TrueNorth, але поки що він знаходиться на стадії розробки і тестування.

Нейронні мережі використовуються повсякденно, їх основні функції:

1. Класифікація – дані розподіляються за необхідними параметрами.

2. Передбачення – виходячи з даних мережа може передбачити наступний хід об'єкта.

3. Розпізнавання об'єктів на фотографіях, голосів тощо.

Серед плюсів нейронних мереж виділяють:

– здатність здійснювати роботу при великій кількості зайвих даних;

– адаптація і швидкодія;

– за рахунок багаторазового навчання нейронні мережі можуть вирішувати задачі з нестандартними наборами даних.

До мінусів нейронних мереж відносять:

– навчання мереж досить витратна і трудомістка справа;

– результат роботи мережі не завжди можна передбачити.

ВИСНОВКИ

Таким чином використання нейроморфних процесорів це найбільш швидкий і вигідний шлях для створення, вивчення і навчання багатофункціональних нейронних мереж, що в свою чергу призведе до автоматизації ще більшої кількості сфер нашого життя, та відкриє нам новий шлях для розвитку технологій.

ЛІТЕРАТУРА

1. Нейроморфные системы: компьютеры, вдохновлённые устройством человеческого мозга [Электронный ресурс]. – 2019. – Режим доступа до ресурсу: <https://habr.com/ru/company/it-grad/blog/457578/>.
2. Кратко об архитектуре нейроморфных процессоров: взгляд изнутри [Электронный ресурс]. – 2018. – Режим доступа до ресурсу: <https://habr.com/ru/company/it-grad/blog/426737/>.
3. Intel Pohoiki Beach — нейроморфная система с 8 млн нейронов [Электронный ресурс]. – 2019. – Режим доступа до ресурсу: <https://habr.com/ru/company/intel/blog/460211/>.

Скорик М. В.

Криворізький національний університет

Купін А. І.

д-р техн. наук, професор, Криворізький національний університет

ПРОЕКТУВАННЯ КОМП'ЮТЕРНИХ СИСТЕМ НА ОСНОВІ АРХІТЕКТУРИ ANFIS

Розглянуто актуальність застосування, структуру та принципи роботи комп'ютерних систем на основі архітектури ANFIS. Надано характеристику ANFIS, наведено основні переваги обчислювальних систем, створених на даній архітектурі.

У сучасному світі комп'ютери можуть обчислити достатньо велику кількість інформації за короткий час. Цей факт сприяє розвитку моделювання процесів мозкової активності за допомогою математичних алгоритмів та їх подальшого обчислення в комп'ютерних системах. Так нейронні мережі використовують у все більшій кількості галузей для рішення різних задач. Окрім звичайних нейронних мереж, існують гібридні нейрон-нечіткі обчислювальні мережі. Нечіткі нейронні мережі вважаються доволі перспективними у сфері слабо структурованих задач прикладного системного аналізу. Однією з таких мереж є гібридна нечітка нейронна мережа типу ANFIS[1].

ANFIS (Adaptive Neuro-Fuzzy Inference System або українською «Адаптивна мережа на основі системи нечіткого виводу») – це спеціалізована мережа штучного (або обчислювального) інтелекту, що базується на системі нечіткого виводу за алгоритмом Такагі-Сугено. Даний метод обробки інформації є доволі новим, так як він був розроблений лише у 1990-му році. ANFIS одночасно поєднує в собі принципи нечіткої (тобто «fuzzy» англійською) логіки та підходи традиційних нейронних мереж. При цьому базові принципи нейронної мережі дозволяють системі навчатися обчислювати вхідні/вихідні дані, а методологічні підходи нечіткої логіки (а саме набори правил типу «IF THEN ELSE») дають можливість навчання апроксимування функції на виході системи [2]. Приклад простого контролера типу Такаго-Сугено з двома входами, одним виходом та набором правил типу:

$IF P_{11}(x_1) AND P_{12}(x_2) THEN f_1(x_1, x_2)$
 $IF P_{21}(x_1) AND P_{22}(x_2) THEN f_2(x_1, x_2)$

на основі застосування методології з типовою архітектурою типу ANFIS зображено надалі на рисунку 1.

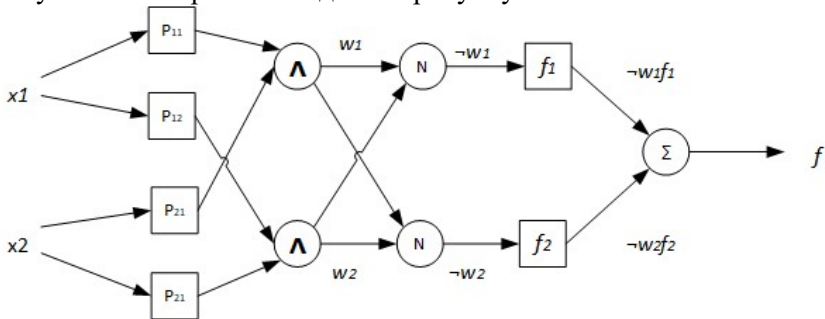


Рисунок 1 – обчислювальна мережа типу ANFIS для простого контролеру на основі принципу Такаґо-Сугено

ВИСНОВКИ

Отже ANFIS – це перспективна інтелектуальна обчислювальна система, яка включає в себе, як базові принципи та переваги нейронних мереж, так і методологічні підходи на основі нечіткої логіки. Це в подальшому спрощує побудову комп’ютерної системи, адже нечіткі правила більш прості для розуміння, а нейронна мережа, в свою чергу, спрощує процес створення і реалізації нечітких правил.

ЛІТЕРАТУРА

1. Богатиков, В.Н. Построение систем управления на основе нейронных сетей: Учебно–методическое пособие / В.Н. Богатиков, Л.В. Дранишников, А.Е. Пророков. - Апатиты: Изд-во КФ ПетрГУ, 2011. – 41 с.
2. “Adaptation of Fuzzy Inference System Using Neural Learning” A. Abraham Computer Science Department, Oklahoma State University, [Електроний ресурс] – Режим доступу до ресурсу: <http://ajith.soft-computing.net>

*Лебідь Д.О.
Криворізький національний університет
Купін А.І.*

Д.т.н., професор, Криворізький національний університет

КОМП'ЮТЕРНА СИСТЕМА НА ОСНОВІ АРХІТЕКТУРИ ЗІ ШТУЧНИМ НЕЙРОПРОЦЕСОРОМ

Показано, що існують декілька видів нейропроцесорів: нейроморфні, тензорні, процесори машинного зору. Вони застосовуються при апаратному прискоренні емуляції роботи нейронних мереж, цифровій обробці сигналів (режими реального часу). Можуть існувати у вигляді окремого процесора, у складі внутрішніх сопроцесорів та GPU-рішень.

З давніх часів людина шукала можливість спростити та прискорити різноманітні робочі процеси. Безперечно, з винаходженням першого комп'ютеру, людина зробила великий крок у майбутнє, можливо навіть не здогадуючись про це. [1]

Арифметичні обчислення, котрі виконували комп'ютери при дешифровці таємних повідомлень «Енігми» в часи Другої Світової Війни дали вченим зрозуміти, що за комп'ютерами «майбутнє». З роками потужності та можливості комп'ютерних систем збільшувалися в геометричній прогресії. [2]

За допомоги комп'ютерів людина отримала змогу здійснити політ у космос, а згодом зробила можливим створення механізмів на базі комп'ютера за для їх залучення у робочі сфери, що сприяло подальшому розвитку автоматизації в промисловій, військовій та науковій галузях. Але згодом людство задалося питанням «а чи можливо створити систему, яка зможе працювати самостійно і в цей же час не буде поступатися людині в можливостях...» Отже, виникла думка про створення «штучного інтелекту».

Внаш час достатньо великого поширення набули нейронні мережі в процесі реалізації інтелектуальних структур [2].

Зараз вони переважно реалізуються двома шляхами:

1. Апаратні розробки (нейропроцесори, графічні прискорювачі, кластери, тощо)

2. Програмні розробки (нейросимулятори, наприклад NeuroSolutions, **NeuroShell**, тощо)

В порівнянні із вище наведеними програмними розробками, апаратні (нейропроцесори) мають істотні переваги, такі як: вища продуктивність, справжня паралельність обчислень, масштабованість тощо.

На рис.1 приведено Блок-схема нейропроцесора типу NeuroMatrix [3]

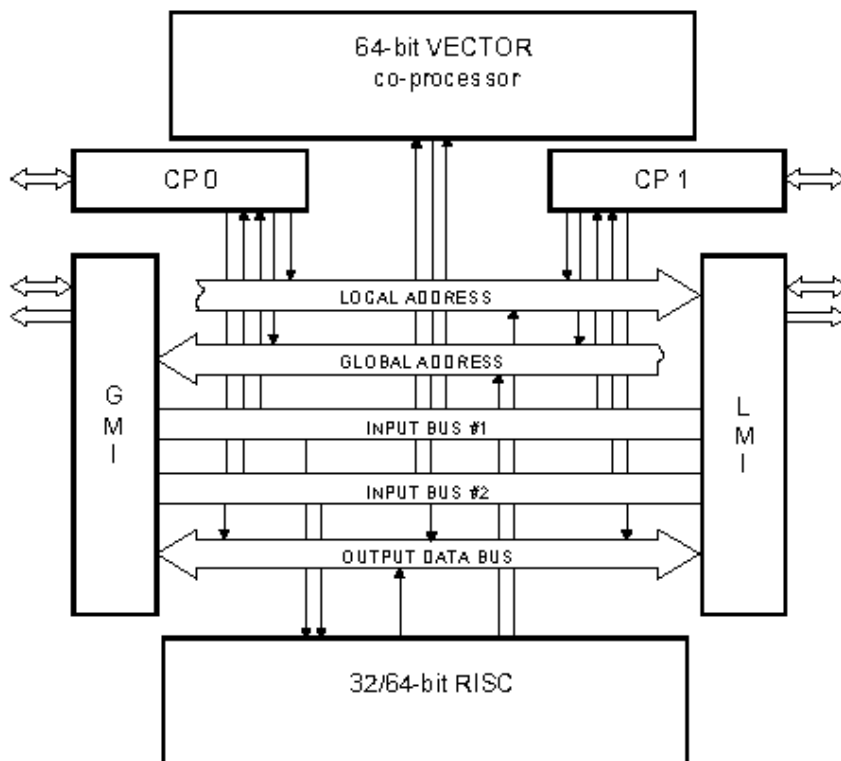


Рис.1 Блок-схема нейро процесора туну NeuroMatrix

ВИСНОВКИ

Отже, існують безпілотні автомобілі та літальні апарати на базі нейропроцесорів. Значна кількість обчислень використовується

в медичній галузі для прогнозування розвитку хвороби та моделювання складних хімічних зв'язків у сучасних медичних препаратах спрямованих на боротьбу з ними. Не можна також залишати без уваги технології розпізнавання голосу, машинний переклад, обробки природної мови, оскільки кожний з цих напрямків спрямований на покращення рівня нашого життя.

ЛІТЕРАТУРА

1. ШІ-прискорювач:
<https://uk.wikipedia.org/wiki/%D0%A8%D0%86-%D0%BF%D1%80%D0%B8%D1%81%D0%BA%D0%BE%D1%80%D1%8E%D0%B2%D0%B0%D1%87>
2. Енігма:
[https://uk.wikipedia.org/wiki/%D0%95%D0%BD%D1%96%D0%B3%D0%BC%D0%B0_\(%D0%B0%D0%B2%D1%82%D0%BE%D0%BC%D0%B0%D1%82\)](https://uk.wikipedia.org/wiki/%D0%95%D0%BD%D1%96%D0%B3%D0%BC%D0%B0_(%D0%B0%D0%B2%D1%82%D0%BE%D0%BC%D0%B0%D1%82))
3. Блок-схема процесора NeuroMatrixNM6403:
<https://www.semanticscholar.org/paper/NeuroMatrix%C2%AE-NM6403-DSP-with-Vector%2FMatrix-engine-Fontine-Tchernikov/60e3c79a807495dbd318b5d7058b6879fa207d2f>

*Сулаков В.О.,
Одеський національний політехнічний університет
Кошутіна Д.В.,
Одеський національний політехнічний університет
Богданова Л.О.,
ст. викладач, Одеський національний політехнічний університет
Ситніков В.С.,
д.т.н., проф., Одеський національний політехнічний університет*

ВИКОРИСТАННЯ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ АНАЛІЗУ ДОРОЖНО-ТРАНСПОРТНИХ ПРИГОД

Центри аналітики великих міст України неспроможні виявляти точні характеристики виникнення аварійності та запобігати цим подіям. Запропоновані моделі дозволяють використовувати

вати класичні методи класифікації об'єктів на зображеннях для виявлення характерних умов виникнення транспортної події на основі навчання системи контролю аварійності.

Дорожньо-транспортні пригоди (ДТП) виникають під дією таких основних елементів: дорожньо-транспортна обстановка, транспортний засіб, водії/пішохід (учасник дорожнього руху).

Вплив дорожньо-транспортної обстановки обумовлюється наступними факторами: незадовільне планування доріг; недосконалість дорожнього покриття, споруд та засобів регулювання; дефекти дорожнього полотна; мала пропускна спроможність вулиць; незадовільна оглядовість, особливо на перехрестях, погана видимість та ін. Транспортний засіб може бути найбільш активним у розвитку механізму дорожньо-транспортних подій.

Стан водія особливо впливає на аварійність на дорогах, зокрема його емоційний, психічний та фізичний стани, крім того стаж та досвід водіння. Такі ж саме характеристики можна застосувати до інших учасників дорожніх подій, таких як велосипедисти, ролери, пішоходи та інші.

Таким чином, однією з особливостей огляду місця ДТП є необхідність дослідження на місці всіх його елементів, механізму дії та взаємодії. Проте усі ці дії проходять постфактум, та мають головною ціллю виявити винних для подальших судових дій і в меншому ступені — збір статистики про аварійність, з ціллю застосувати заходи для її зменшення.

Тому доцільно розробляти методи та інструментальні засоби щодо прогнозування аварійності з ціллю запобігання, або попередження виникнення дорожньо-транспортних подій, шляхом розробки прогнозних систем, які можуть аналізувати та попереджувати про можливість аварійності на майбутній період, або в реальному часі.

На даний момент, в Україні можна проводити аналіз на основі двох факторів: аналіз відеозаписів з камер відеоспостереження сторонніх осіб та організацій та дані з протоколів правопорушення з місць виявлення аварій.

Враховуючи, що в Україні вводиться відеофіксація дорожньої обстановки в реальному часі, а, наприклад в Одесі, інформаційний центр відеонагляду, даний аналіз можна застосувати для оцінки

ймовірності виникнення аварійної обстановки з розширеною зоною контролю.

В рамках даного дослідження запропоновано використання комбінованих методів аналізу даних, серед яких є зображення з камер, фотофіксація, протоколи ДТП, любительські відео, що надасть можливість виявляти причинно-наслідкові зв'язки в подіях, та запобігати виникненню подібних подій, шляхом онлайн аналізу обставин в місті, та виявленню відомих причин, які призводять до аварій. Основними методами дослідження для розробки системи аналіз є нейроні мережі для аналізу та прогнозування.

Вихідними даними для аналітики має бути множина пар факторів, які впливають на аварійність та їх ваги у нейронній мережі. Множина цих пар описує те, які саме параметри є характерними у випадках дорожньо-транспортних подій. Отже, процес аналізу повинен, окрім виявлення пар «нейрон-вага», визначати які саме елементи були характерні для події. Це призводить до необхідності проведення кластеризації виявлених пар, з урахуванням їх приналежності до конкретних елементів.

Таким чином визначення вихідної множини для моделі аналізу визначається в два етапи – визначення пар нейрон/вага та визначення близьких значень для конкретної події.

При даному процесі, вихідними даними є множина векторів, кожний з котрих відповідає конкретному випадку, а елементи цих векторів відповідають параметрам, які ймовірно призвели до цих подій. Для того, щоб сформувані вихідну множину прогнозних значень аварійності, в ході дослідження, прогнозну модель було реалізовано у вигляді алгоритму формування висновку.

Реалізація моделі передбачає надходження множини параметрів на блок декомпозицій характеристик. На даному етапі проводиться сортування даних за ознаками елементів моделі та їх підготовка до подання на вхід прогнозної моделі. Сортування здійснюється згідно моделі, тобто визначаються візуальні дані відео та фото фіксації об'єктів аналізу, визначення протокольної інформації для створення статистичних даних прогнозування та текстових тестових даних для уточнення моделі прогнозування. Запропоновано використання вихідного вектору, який складається з 7 значень. Це корелюється з кількістю вхідних елементів моделі аналізу даних та моделі прогнозування, тобто з вхідними векторами. Фізичний

смысл такого підходу наступний – вихідний вектор оцінює інтегральну оцінку впливу кожного елементу моделі з урахуванням її композиції з іншими елементами.

В ході дослідження було запропоновано нейромережеві моделі аналізу та прогнозуванню аварійності. Визначені моделі були оцінені з точки зору точності моделювання, що показало точність моделі за показником MARE у середньому 2,23%, за сумою квадратів помилок 3,7%, за сумою квадратів відхилень 3,17%. Крім того, використання системи дозволило скоротити час аналізу ДТП на 37%, в порівнянні з аналогічною роботою експертної групи.

Хорошев О.О.,

Одеський національний політехнічний університет

Реклізон Г.О.,

Одеський національний політехнічний університет

Богданова Л.О.,

ст. викладач, Одеський національний політехнічний університет

Ситніков В.С.,

д.т.н., проф., Одеський національний політехнічний університет

РОЗПІЗНАВАННЯ НОМЕРІВ АВТОМОБІЛІВ НА ОСНОВІ ЕЛЕМЕНТІВ НЕЙРОННИХ МЕРЕЖ

Робота присвячена розробці системи розпізнавання автомобільних номерів на основі нейронної мережі, яка дозволяє слідкувати за дорогами в місті та допомагає оперативно реагувати на інциденти. В роботі проаналізовано основні математичні методи розпізнавання, виділено їх переваги та недоліки. Для поліпшення точності розпізнавання тексту використовуються згорткові і рекурентні шари в мережі, що дозволило підвищити ефективність системи.

В даний час існує ряд систем визначення номерних знаків, не всі з яких відповідають необхідним показникам. Для цих цілей паралельно з програмним забезпеченням розробляються і апаратні засоби. Системи, що володіють високою швидкістю і точністю розпізнавання, як правило, дуже дорогі. Висока вартість існуючих продуктів не дозволяє здійснити їх масове впровадження. Крім того, у

сучасному світі поширена тема штучного інтелекту та комп'ютерного зору, але більшість сучасних систем є платними та заявлена точність продукту не збігається з дійсною. Для вирішення проблеми точності системи, як правило, збільшують набір даних, але у цього підходу велика витрата ресурсів серверу та часу.

Тому, потрібно оптимізувати алгоритм розпізнавання за рахунок використання архітектури, яка використовує згорткові та рекурентні шари в нейронній мережі. За допомогою цього підходу можна поліпшити точність алгоритму розпізнавання без використання великого обсягу комп'ютерних ресурсів та часу.

На даний момент для систем розпізнавання номерів автомашин широко використовують хвильовий метод, метод адаптивного розпізнавання та фонтанного перетворення.

Але побудова і тестування структурно-плямових еталонів для класів розпізнаваних об'єктів, процес складний і трудомісткий. База зображень, яка використовується для налагодження описів, повинна містити приклади хороших і поганих (гранично допустимих) зображень для кожної випадку, а зображення в базі поділяються на навчальні і контрольні.

При використанні нейронних мереж слід відзначити, що згорткові нейронні мережі забезпечують часткову стійкість до змін масштабу, зсувів, поворотам, зміні ракурсу і іншим спотворень. Вони об'єднують три архітектурних ідеї: локальні рецепторні поля (забезпечують локальну двовимірну зв'язність нейронів); загальні вагові коефіцієнти синапсів (забезпечують детектування деяких рис в будь-якому місці зображення і зменшують загальне число вагових коефіцієнтів); ієрархічну організацію з просторовими підвибірками. На даний момент згорткова нейронна мережа і її модифікації вважаються кращими по точності і швидкості алгоритмами знаходження об'єктів на сцені. Ускладнюючи систему за рахунок рекурентних шарів дозволяє задіювати внутрішню пам'ять мережі для обробки послідовності довільної довжини.

Так, на першому етапі визначається кут повороту номера в площині зображення. На другому - виконується алгоритм отримання нормалізованого зображення номера з вихідного зображення з урахуванням кута його повороту. Для повороту області зображення, що відповідає, нормалізованому номеру, використовується алгоритм, заснований на відповідному перетворенні координат. Для

зменшення спотворень зображення при повороті, пов'язаних з його дискретним характером, використовується метод, заснований на білінійній інтерполяції по найближчих чотирьох пікселях.

Визначення кута повороту зображення номерного знака виконується з використанням декількох етапів обробки та аналізу зображень. На першому етапі виконується операція підкреслення кордонів на зображенні на основі лінійного оператора Собеля для горизонтальних кордонів, яка має маску згортки

$$A = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

На другому етапі виконується розрахунок карти щільності знайдених точок меж в просторі коефіцієнтів лінійних залежностей просторових координат згідно перетворенню Хафа. Метою другого етапу є визначення рівняння прямих, що відповідають верхній і нижній межі номерного знака.

Таким чином завдяки нормалізації система може правильно виявляти текст навіть з труднодоступних місць на зображенні. Після нормалізації, усі данні записуються в базу даних. Завдяки такому підходу, для повторного навчання не потрібно пере розмічати дані, а використовувати вже готове представлення, яке зберігається в базі даних.

Дана архітектура дозволяє отримати до 87% точності правильного виявлення, використовуючи при цьому невеликий об'єм комп'ютерних ресурсів.

*Міргородченко І.О.
бакалавр, Одеський національний політехнічний університет
Ковальчук Є.Д.
бакалавр, Одеський національний політехнічний університет
Ситніков В.С.
д.т.н., проф., Одеський національний політехнічний університет
Дікусар К.В.
ст. викл, Одеський національний політехнічний університет*

ЗАСТОСУВАННЯ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ РОСПІЗНАВАННЯ КУРИЛЬНИКІВ У ГРОМАДСЬКИХ МІСЦЯХ

Проведено застосування нейронної мережі з використанням функцій комп'ютерного зору, яка несе у собі реалізацію розпізнавальної системи курців. Робота виконана на некомерційній системі YOLODarkne. Розглянуто також методи попередньої фільтрації та обробки зображень у реальному часі.

Останнім часом країну захлиснула велика проблема – куріння у громадських місцях. Багато людей страждають від пасивного куріння, через те, що правоохоронні органи не можуть повністю відслідкувати за кожною людиною. Тому пропонується введення програмного продукту на основі нейронної мережі, який має можливість ідентифікувати курця у громадських місцях.

Для застосування нейронної мережі треба виділити основні її функції: розпізнавання, класифікація, передбачення.

Якщо аналізувати існуючі рішення варто звернути увагу на систему – Darknet YOLO, дана система є однією з кращих з відкритим вихідним кодом.

Нейронна мережа є згортувальною. Тренується дана система масивом збережених фото – ImageNet. Дану систему побудовано на основі архітектури YOLOv3 яка має 106 шарів.

Дана система повністю виконує поставлене завдання – розпізнає об'єкти людини та цигарки. Система була розроблена базуючись на систему YOLODarknet, який треба було повністю модифікувати під поставлене завдання. Щодо розпізнавання так

званих вейпів, та трубок, на жаль, на даний момент Закон України нічого не говорить про дані типи зловмисних предметів, тому намагатися їх розпізнавати – не було доцільно.

Доцільно вважати що вдалося розпізнати курців з зображень з непоганою якістю. Це свідчить про те, що система було достатньо тренувана, та могла приймати рішення правильно. Щодо результатів отриманих з камер відеоспостереження, то вони не такі однозначні. По-перше, практично усі доступні камери відеоспостереження, знаходяться дуже високо. Це є великою проблемою для розпізнавання, крім того якість встановлених камер необхідно покращити. При поганих погодних умовах, дані камери є більш формальним предметом, ніж корисним.

*Ivchenko Rodion Anatoliyovych,
Graduate student, Kryvyi Rih National University
Kupin Andriy Ivanovich,
Doctor of Technical Sciences, Professor, Kryvyi Rih National
University*

RESEARCH AND DEVELOPMENT OF ACTUAL METHODS, TECHNOLOGIES AND METHODS APPLIED TO SOLUTION OF APPLIED MACHINE TRAINING PROBLEMS FOR PREDICTIVE PROTECTION

A study was made of relevant techniques, technologies and techniques used to solve applied problems of machine learning, based on materials from scientific articles in highly rated journals of foreign researchers, analytical and review notes from open sources, as well as technical documentation and press releases of technical and software solutions. The search for new methods of model selection, cross-validation, evolutionary and analytical selection of training algorithms is of both scientific and purely practical interest. The development of machine learning technologies will only accelerate in the near future. Currently, we are witnessing progress in the development of automated search methods for constructing effective learning models for data analysis that are applicable to many practical problems of data mining. During the review of modern trends in machine learning, we identified promising areas of fundamental and applied research in this area [1].

Development of a process model based on the use of neural networks. Neural networks are successfully used for the synthesis of control systems for dynamic objects. Neural networks have a number of properties that determine the prospects of their use as an analytical apparatus of control systems. In the context of the problem under consideration, this is, above all, the ability to learn by example. The presence of large volumes of monitoring data, which presents interconnected measurements of both the inputs and outputs of the studied system, allows the neural network to be provided with representative training samples. Other important properties are the ability of the neural network to adapt to changes in the properties of the control object and the external environment, as well as high resistance to “failures” of individual network elements due to the parallelism originally built into its architecture. The ability of a neural network to predict directly follows from its ability to generalize and highlight hidden relationships between input and output data. After training, the network is able to “predict” future output values based on several previous values and current monitoring data. In the framework of ongoing research, the most promising is the use of counterpropagation networks [2].

Among the considered neural network architectures, the following networks are applicable for solving the problems of approximation and regression analysis: multilayer perceptrons and radial basis networks. Both types of neural networks have their advantages and disadvantages when used in dependency recovery tasks. Each of the networks considered effectively approximates complex functions, learning from noisy data. Multilayer perceptrons show good results in processing experimental data, including multidimensional ones, allowing us to simulate patterns hidden in them. When training a 3-layer perceptron based on the linear function of activation of the output neuron and hidden layers with a hyperbolic activation function, it showed the best result in terms of training accuracy and prediction accuracy.

REFERENCES

1. Koroteev, MV A review of some current trends in the technology of machine learning Creative Commons Attribution 4.0. the world. (<http://creativecommons.org/licenses/by/4.0/>)
2. Chernodub, AN Review of Neuro-management Methods / AN

*Мельник А. О.,
Державний університет «Житомирська політехніка»
Вакалюк Т. А.,
д.п.н., доцент, професор,
Державний університет «Житомирська політехніка»*

АНАЛІЗ ШТУЧНОГО ІГРОВОГО ІНТЕЛЕКТУ

Автором проведено аналіз роботи штучного ігрового інтелекту та зазначено обмеження в його розробіці, що дозволить розглядати підходи до створення ігрового штучного інтелекту.

Ігровий штучний інтелект в основному призначений для вибору дій сутності в залежності від поточних умов. Традиційна література по штучному інтелекту називає дане керування «інтелектуальними агентами». Агентом зазвичай є персонаж гри, але це може бути і машина, робот або навіть щось більш абстрактне – ціла група сутностей, країна або цивілізація. У будь-якому випадку це об'єкт, що стежить за своїм оточенням, що приймає на підставі нього рішення і діє відповідно до цих рішень. Іноді це називають циклом «сприйняття-мислення-дія»:

- сприйняття: агент розпізнає, або йому повідомляють інформацію про оточення, яка може впливати на його поведінку (наприклад, що він знаходиться поблизу небезпеки, які предмети він може зібрати, важливі точки і так далі);
- мислення: агент приймає рішення про те, як вчинити (наприклад, вирішує, чи достатньо безпечно підійняти предмет, чи варто йому боротися або краще спочатку сховатися);
- дія: агент виконує дії для реалізації своїх рішень (наприклад, починає рухатися по маршруту до ворога або до предмету, і так далі);
- ...потім через дії персонажів ситуація змінюється, тому цикл повинен повторитися з новими даними.

Завдання штучного інтелекту, особливо ті, що є актуальними сьогодні, зазвичай зосереджені на «сприйнятті». Наприклад, безпілотні автомобілі повинні отримувати зображення дороги, комбінуючи їх з іншими даними (радара і лідара) і намагатись інтерпретувати те, що вони бачать. Зазвичай, це завдання вирішується машинним навчанням, що особливо добре працює з великими масивами даних реального світу і надає їм якесь значення, витягуючи семантичну інформацію. Такі завдання називаються завданнями класифікації.

Гри незвичайні тим, що для отримання цієї інформації їм не потрібна складна система, оскільки вона є невід'ємною частиною симуляції. Немає необхідності виконувати алгоритми розпізнавання зображень, щоб виявити ворога перед собою; гра знає, що там є ворог і може передати цю інформацію безпосередньо процесу прийняття рішень. Тому «сприйняття» в цьому циклі зазвичай сильно спрощено, а вся складність виникає в реалізації «мислення» і «дії».

Під час розробки ігрового штучного інтелекту є певні обмеження:

- на відміну від алгоритму машинного навчання, він зазвичай не тренується заздалегідь; при розробці гри непрактично писати нейронну мережу для спостереження за десятками тисяч гравців, щоб знайти найкращий спосіб грати проти них, тому що гра ще не випущена і гравців у неї немає;
- зазвичай, передбачається, що гра повинна розважати і кидати гравцеві виклик, а не бути «оптимальною» - тому навіть якщо і можна навчити агентів протистояти гравцям найкращим чином, то швидше за все гравець від цього задоволення не отримає;
- часто до агентів висувають вимогу «реалістичного» поведінки, щоб гравці відчували, що змагаються з людиноподібними противниками. Програма AlphaGo виявилася набагато краще, ніж люди, але обрані нею ходи настільки далекі від традиційного розуміння гри, що досвідчені супротивники говорили про неї як про гру проти інопланетянина. Якщо гра симулює противника-людини, то зазвичай це небажано, тому алгоритм потрібно налаштувати таким чином, щоб він приймав правдоподібні рішення, а не ідеальні;

- штучний інтелект повинен приймати рішення «в реальному часі». У цьому контексті це означає, що алгоритм не може для прийняття рішення монополізувати ресурси процесора на тривалий час. Тому що більшість ігор мають всього декілька мілісекунд на виконання всіх операцій для наступного кадру графіки;
- в ідеалі хоча б частина системи повинна залежати від даних, а не бути жорстко заданою, адже так процес налаштування буде значно простіше.

На основі проведеного аналізу, можна розпочати розглядати підходи до створення штучного інтелекту, що реалізують весь цикл «сприйняття-мислення-дія» способами, що забезпечують ефективність і дозволяють дизайнерам гри вибирати складні поведінки, схожі на дії людини.

ЛІТЕРАТУРА

1. Game Engine Architecture / Jason Gregory, 2018 рік
2. Обзор техник реализации игрового ИИ [Електронний ресурс] – Режим доступу до ресурсу: <https://habr.com/ru/post/420219>
3. GameAIPro / David “Rez” Graham, 2016 рік
4. GDX-AI [Електронний ресурс] – Режим доступу до ресурсу: <https://github.com/libgdx/gdx-ai/wiki/>

**СЕКЦІЯ 6. AUTOMATION, INDUSTRY 4.0.
ПРОМИСЛОВІ МЕРЕЖІ, КІБЕРФІЗИЧНІ СИСТЕМИ,
ВЕЛИКІ ДАНІ, ІНТЕРНЕТ РЕЧЕЙ, МОБІЛЬНІ ТА ХМАРНІ
СЕРВІСИ, ЗАСОБИ ДОПОВНЕНОЇ РЕАЛЬНОСТІ**

Федоренко О.О.

Херсонський національний технічний університет

Сидорук М.В.

к.т.н., доцент, Херсонський національний технічний університет

ВИКОРИСТАННЯ ХМАРНИХ ТЕХНОЛОГІЙ В БІЗНЕСІ

У даній статті розглянуті проблеми впровадження та використання хмарних технологій в малому і середньому бізнесі. Виявлено переваги хмарних технологій в порівнянні з традиційними серверами.

Технічний прогрес не стоїть на місці, і сьогодні однією з найбільш обговорюваних тем в діловому світі є - хмарний сервіс, або хмарне сховище. Ця послуга надається будь-якої організацією, вона виділяє віртуальний простір (в обмеженому обсязі) на сервері для зберігання різної інформації. Користувач має можливість завантажувати дані за допомогою програми на хмарний сервер, і ці дані будуть доступні йому з інших комп'ютерів або смартфонів. Крім того, хмарний сервіс дозволяє користуватися віддалено різними програмами.

На відміну від традиційних серверів, хмарний сервіс є найбільш необхідним в середовищі малого і середнього бізнесу. Як правило, бізнес-процеси невеликих компаній досить прості, і для задоволення їх потреб достатньо впровадження продуктів SaaS. Дослівно це розшифровується як Software as a Service - «програмне забезпечення як послуга». Сервіси цієї групи виглядають як звичайні інтернет-сайти, але працюють як повноцінні програми.

Існує безліч позитивних моментів використання в організації хмарних сервісів. Економія коштів очевидна, в ситуації розширення організації, за умови використання віддаленого комп'ютера істотно знижуються вимоги до технічної складової ПО організації. Також, ресурси виділяються в міру їх необхідності, в будь-який момент

можна збільшити або скоротити потужність. Для подібних змін раніше були потрібні місяці.

Незаперечна перевага даної технології полягає в мобільності співробітників організації. Можливість отримання даних з «хмари» з будь-якого іншого комп'ютера або навіть смартфона, ліквідує проблеми доступу до даних при, припустимо, переїзді головного офісу [1].

Звичайно, є і негативні моменти в застосуванні хмарних сервісів (табл. 1).

На жаль, безпека - це все ще предмет суперечок прихильників і противників використання хмарних технологій. Це питання має дві сторони: реальне забезпечення безпеки даних клієнтів із застосуванням кращих технологій і практик, а також підтвердження відповідності процесів забезпечення безпеки вимогам стандартів і законодавства. З метою підтвердження відповідності провайдера хмарних сервісів вимогам стандартів і законодавства, а від клієнта - обґрунтування можливості і необхідності винесення важливих інформаційних систем на хмарну платформу, що знаходиться на сторонньому майданчику, для чого доводиться дуже багато і наполегливо працювати службам власної безпеки клієнтів (разом з ІТ-директорами) з постачальниками сервісу (табл. 1).

Таблиця 1 – Переваги та недоліки хмарного сервісу

Переваги	Недоліки
Поліпшення продуктивності і мобільності співробітників	Повна залежність від інтернет - провайдерів.
Широкий вибір програмних засобів хмарного сервісу.	Відносно низький рівень безпеки.
Економія коштів при розширенні бізнесу.	Негативне ставлення до впровадження нових технологій.
гнучкість	
Зниження ризику втрати даних при псуванні обладнання.	
Додатки в «хмарі» можуть автоматично відновлюватися при виявленні несправності.	
терфейс продуктів SaaS інтуїтивно зрозумілий, відпадає	

необхідність користування послугами сторонніх фахівців.	
---	--

дана проблема впливає з менталітету громадян. Фахівці критично і, часто, навіть упереджено, ставляться до інновацій, які з'являються на ринку технологій. І навіть явні переваги в економічній ефективності не можуть переважити упереджене ставлення і небажання служб інформаційної безпеки слідувати технологічним віянням [1].

ВИСНОВКИ

Загальний перехід до хмарних сервісах, як в Україні, так і в світі - лише питання часу. Основна причина непопулярності даних рішень є Негативне ставлення до впровадження нових технологій організацій та суттєвих змін інфраструктури. Дієвим рішенням для розвитку хмарних обчислень в Україні може виявитися перехід до надання різних державних послуг на основі цих сервісів.

ЛІТЕРАТУРА

1. Свекровкіна Е.А. Хмарні технології в малому і середньому бізнесі. Проблеми впровадження та використання // Економіка і менеджмент інноваційних технологій. 2013. № 6 [Електронний ресурс]. URL: <http://ekonomika.snauka.ua/2013/06/2361>.

*Мельник О. О.,
Харківський національний університет радіоелектроніки
Боцман І. В.
к.т.н., доцент, Харківський національний університет радіоелектроніки*

РОЗРОБКА ВЕБ-ДОДАТКУ ДЛЯ КЕРУВАННЯ ЕЛЕКТРОННИМИ ПРИСТРОЯМИ НА ПРИЛАДОБУДІВНОМУ ВИРОБНИЦТВІ У РАМКАХ РЕАЛІЗАЦІЇ КОНЦЕПЦІЇ ІНДУСТРІЇ 4.0

У даній роботі наведено результати розробки концепції керування електронними пристроями через веб-додаток, що спрямована на впровадження на приладобудівних підприємствах основних

засад Індустрії 4.0. Проаналізовано та обрано мікроконтролер для керування пристроями.

Концепція впровадження засад Індустрії 4.0 передбачає перехід на повністю автоматизоване цифрове виробництво, кероване інтелектуальними системами у режимі реального часу за постійної взаємодії із зовнішнім середовищем, що виходить за межі одного підприємства, з перспективою об'єднання у глобальну промислову мережу Речей і послуг [1].

Індустрія 4.0 характеризує поточний тренд розвитку автоматизації та обміну даними, який включає в себе кіберфізичні системи, Інтернет речей і хмарні обчислення. Кіберфізичні системи створюють віртуальні копії об'єктів фізичного світу, контролюють фізичні процеси та приймають децентралізовані рішення. Вони здатні об'єднуватися в одну мережу, взаємодіяти у режимі реального часу, самоналагоджуватися та самонавчатися. Важливу роль відіграють Інтернет-технології, які забезпечують комунікації між персоналом і обладнанням [2].

Тож, Індустрія 4.0 являє собою новий виток промислової революції, який характеризується інтеграцією виробництва та мережевих комунікацій. Нове покоління технологічного обладнання дозволяє збирати актуальні дані у реальному часі, виготовляти персоналізовані продукти, створюючи прямі зв'язки ланцюжка виробництва від замовлення продукту до отримання його споживачем у найкоротші терміни з максимальною ефективністю процесу. Зокрема у промисловості планується модернізувати верстати так, щоб вони здатні були оновлювати програми завдяки підключенню до Інтернету, аналізувати власний знос, вдосконалюватися для виконання різноманітних завдань [1].

Для врахування вимог з модернізації виробничих систем у приладобудівній галузі було сформульоване завдання керування електронними пристроями на виробництві за допомогою веб-інтерфейсу [3].

На рис. 1. наведено структуру інтерфейсу: користувач заходить на сторінку керування пристроями, після цього з'являється можливість змінювати стан пристрою (вмикати або вимикати його), змінювати його параметри, наприклад, швидкість роботи пристрою,

та налаштовувати загальне функціонування системи, зокрема задаючи вид чи кількість продукції, що необхідно виробити.

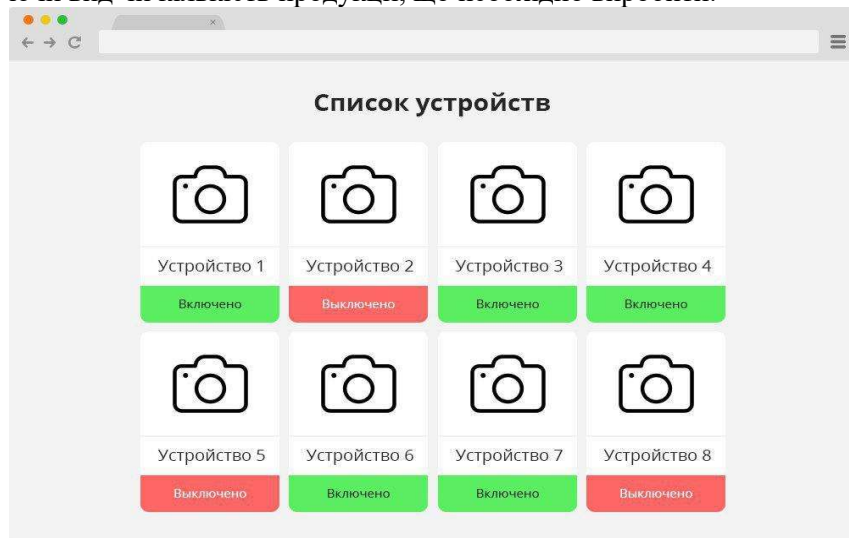


Рис. 1. Інтерфейс керування пристроями

Під час обґрунтування вибору електронних засобів для реалізації поставленого завдання були проаналізовані два типи мікроконтролерів для вибору найбільш відповідного для реалізації проекту – ESP32 та ESP8266 [4]. За результатами проведеного аналізу можна зробити висновки, що мікроконтролер ESP32 є більш потужним, ніж ESP8266, містить більше вбудованої пам'яті, датчик для підключення сенсорного дисплея та більшу кількість GPIO-виходів з декількома функціями, а також підтримує протоколи Wi-Fi, Bluetooth 4.2 та BLE.

ВИСНОВКИ

Таким чином, за допомогою розроблюваного інтерфейсу передбачається реалізація функцій керування промисловими роботами, верстатами з ЧПК та іншими технологічними об'єктами.

У подальшому планується розробити серверну та клієнтську частини веб-додатку, за допомогою яких і буде виконуватися керування електронними пристроями на промисловому виробництві, а також буде розроблено програму для мікроконтролера.

ЛІТЕРАТУРА

1. Индустрия 4.0: Какие перемены грядут на рынке услуг [Електронний ресурс]. – Режим доступу: <https://delo.ua/business/industrija-40-kakie-peremeny-grjadut-na-rynke-uslug-328161/> – Загол. з екрану.
2. Индустрия 4.0 [Електронний ресурс]. – Режим доступу: <http://kaspersky.vedomosti.ru/industrii/industry4>.
3. Мельник О. О. Розробка веб-сервісу для керування компонентами розумного будинку // Збірник студентських наукових статей «Автоматизація та приладобудування» ADED-2019. Вип. 1, 2019. – Харків: ХНУРЕ. – С. 63-67.
4. Офіційний сайт компанії розробника прошивки [Електронний ресурс]. – Режим доступу: <https://www.espruino.com>. – Загол. з екрану.

*Рижук А.В.
Державний університет «Житомирська політехніка»
Петросян Р.В.
ст. викладач,
Державний університет «Житомирська політехніка»*

ІНФОРМАЦІЙНО КЕРУЮЧА СИСТЕМА МАНІПУЛЯТОРОМ ДЛЯ РОБОТИ В НЕБЕЗПЕЧНИХ УМОВАХ

Проаналізована проблема необхідності розробки інформаційно керуючих систем роботами маніпуляторами. Описано основні компоненти для створення таких систем.

На сьогоднішній день науковий прогрес не стоїть на одному місці, а крокує семи мильними кроками і в сьогодні важливу роль відіграє штучний інтелект та автоматизація всіх виробничих процесів. Роботи надалі замінюють людей там, де безпосередня присутність людини є недоцільною з точки зору економіки, або ж неможливою, чи шкідливою для життя, або здоров'я. Якщо враховувати дані статистики, то маніпулятори складають 84-91% всіх промислових роботів світу, також за даними CAGR ринок робототехніки буде рости і до 2023 року досягне позначки 16,79 млрд. доларів США, з

яких майже 45% припадає на медичні, фармацевтичні та дослідницькі автоматизовані системи. І це не дивно, бо автоматизація з економічної точки зору, дозволяє скоротити час на виготовлення тої чи іншої деталі, зменшити витрати на зарплатню працівникам (замінивши їх роботами), а також зменшує кількість небезпек, або ж нещасних випадків спричинені людським фактором. Так і набагато безпечніше працювати на відстані з небезпечними об'єктами чи речовинами. На превеликий жаль в нашій країні така галузь як робототехніка, знаходиться на досить низькому рівні, можна сказати стоїть на місці, а це означає, що ми не здатні конкурувати з іншими країнами у створенні надійних, якісних та швидкодійних маніпуляторів. Найкращим рушієм для робототехніки може стати аграрний сектор, так як наша країна обдарована родючими землями і автоматизація цієї сфери потрібна понад усе, щоб збільшити врожайність або пришвидшити виробничі процеси.

Беручи до уваги дослідні центри, науковці зацікавлені мінімізувати або ж звести на нуль прямий контакт з небезпечно активними речовинами. При дослідженні використовують досить багато шкідливих речовин, такі як пари газів, їдкі речовини, радіація тощо, які негативно впливають на здоров'я працівників та оточуючих. Також маніпулятори можуть використовуватись військовими, в розмінуванні об'єктів, де сама присутність людини є недоцільною і небезпечною, бо невдале розмінування приведе до втрати людини, що є великим горем для сім'ї потерпілого, але якщо невдале розмінування проведе робот, то його швидко замінять на новий і ніхто при цьому не постраждає. Чудовим прикладом використання маніпулятора є дослідження морського дна. Адже знаходження людини без спеціального обладнання та систем орієнтування під товщою води, де тиск вимірюється не в паскалях, а в мегапаскалях, неможливе, тому альтернативним варіантом заміни буде маніпулятор, так як він без особливих зусиль та часових затрат збере потрібні зразки з морського дна для подальших досліджень. [1]

Отож система управління маніпулятором повинна складатись з 3х основних частин. Самого маніпулятором, серверної частини, де будуть оброблюватись запити та сам клієнт, який буде посилати данні на сервер. Також потрібні телекамери на самому роботі-маніпуляторі, щоб оператор міг бачити всі рухи маніпулятора в реаль-

ному часі. В загальному випадку достатньо однієї камери, але їх кількість може збільшувати в залежності від поставленої задачі. Якщо обставини дозволяють здійснювати передачу інформації за допомогою кабелів, то доцільно використати універсальний послідовний інтерфейс RS-232/RS-485.[2] Якщо умови не дозволяють здійснювати обмін інформацією за допомогою шнурів, то потрібно використовувати безпроводні технології передачі інформації, такі як Wi-Fi, WiMAX тощо. Клієнтська частина являє собою комп'ютер, з'єднаний з сервером використовуючи протокол TCP/IP. [3] Програмне забезпечення клієнтської частини реалізовано за допомогою мови програмування Java. Єдиною умовою є достатня потужність комп'ютера, щоб забезпечити комфортну візуалізацію для оператора. Ідеальна (комфортна) частота оновлення кадрів близько 50-60 кадрів/с. Для мінімального потоку даних між сервером та клієнтом обмін організований за допомогою UTF-8 кодування. Також для передачі зображення потрібна швидкість всього лише 10 КБ/с, використовуючи json-повідомлення.[4]

ВИСНОВКИ

Таким чином, створено метод ефективного дистанційного управління, що в свою чергу майже повністю передає всю необхідну інформацію оператору в режимі реального часу. Завдяки такому способу забезпечується безпека оператора уникаючи прямого контакту при роботі з небезпечними речовинами.

ЛІТЕРАТУРА

1. Автоматизована система віддаленого керування маніпулятором [Електронний ресурс] - Режим доступу до ресурсу: <http://www.rusnauka.com/pdf/248621.pdf>.
2. Алгоритми управління роботом-маніпулятором через інтернет [Електронний ресурс] - Режим доступу до ресурсу: <http://mi.mathnet.ru/mm613>.
3. Luo R., Lee W. et al. Tele-Control of Rapid Prototyping Machine Via Internet for Automated Tele Manufacturing, IEEE International Conference on Robotics & Automation ICRA'99, Detroit (USA), May 1999. – P. 2203-2208.
4. Белоусов И.Р. Методы эффективного управления роботами через сеть Интернет, VIII Всероссийский съезд по теоретической и прикладной механике, Пермь, 23-29 авг. 2001. – С. 91-92.

Зезуль І. В.
Криворізький державний педагогічний університет
Мінтій І. С.
к. пед. н., доцент,
Криворізький державний педагогічний університет

РОЗРОБКА ХМАРООРІЄНТОВАНОГО КУРСУ ШКІЛЬНОЇ ІНФОРМАТИКИ

Метою роботи є розробка курсу дистанційної підтримки шкільної інформатики з використанням вільно поширюваних і хмаро орієнтованих засобів навчання.

Стрімкий розвиток мережі Інтернет спричинив популяризацію хмарних технологій, дослідженням у цьому напрямі присвячено не одну низку конференцій та семінарів, зокрема «Хмарні технології в освіті» (Кривий Ріг, 2012-2019 рр.). Основна перевага хмарних технологій – це можливість працювати будь-де та будь-коли. В той же час, М. П. Шишкіна зазначає, що існує «необхідність модернізації освітньо-наукового середовища вищого навчального закладу, щоб привести його у відповідність до сучасного рівня розвитку науки і технологій» [4]. Доповнимо це твердження тим, що й освітнє середовище закладу загальної середньої освіти потребує такої модернізації, адже хмарні технології вже стали буденними у повсякденному житті пересічного школяра. Відповідаючи запитам сучасності, в курсі шкільної інформатики з'явилась тема «Хмарні сервіси», проте протягом всього шкільного курсу інформатики ще здебільшого використовується комерційні і не хмаро орієнтовані програмні засоби (ПЗ).

Ще одним із актуальних освітніх напрямів сьогодення є неформальна освіта, чому сприяє наявність та розробка масових відкритих електронних курсів на освітніх платформах, проте попередній аналіз показав, що поки що у вітчизняних розробках таких курсів також здебільшого використовуються комерційні ПЗ.

Таким чином, актуальність дослідження зумовлена необхідністю розробки курсу дистанційної підтримки шкільної інформатики з використанням вільно поширюваних і хмаро орієнтованих засобів навчання.

Мета дослідження: розробка дистанційної підтримки шкільної інформатики з використанням вільно поширюваних і хмаро орієнтованих засобів навчання.

Для досягнення мети дослідження були поставлені такі завдання:

1. Проаналізувати масові відкриті електронні курси інформатичного напрямку на вітчизняних освітніх платформах.

2. Визначити вільно поширювані і хмаро орієнтовані аналоги ПЗ, що використовуються у шкільному курсі інформатики.

3. Розробити курс дистанційної підтримки шкільної інформатики з використанням вільно поширюваних і хмаро орієнтованих засобів навчання.

У даній роботі розглянемо результати виконання двох перших завдань.

ВИСНОВКИ

1. Аналіз наявних масових відкритих освітніх курсів інформатичного напрямку на вітчизняних платформах, зокрема, на Prometheus [2] та Edera [1], показав, що: на сайті Prometheus [2] більшість курсів інформатичного напрямку розроблені на підтримку змістової лінії «Основи алгоритмізації і програмування». Це такі курси: «Основи програмування CS50»; «Основи програмування на Java»; «Алгоритми і проекти Scratch»; «Основи Web UI розробки»; «Розробка та аналіз алгоритмів»; «Основи програмування»; «Основи інформаційної безпеки»; «Обробка та аналіз великих даних».

І лише один курс, що стосується змістової лінії «Інформаційно-комунікаційні технології»: «Word та Excel: інструменти і лайфхаки» [3].

На сайті EdEra [1] взагалі розміщено лише один курс інформатичного напрямку: «Основи веб-розробки (HTML, CSS, JavaScript)».

2. Переважна більшість ПЗ, що розглядаються у підручниках з інформатики є комерційними, так, серед текстових і табличних процесорів та редакторів презентацій найбільш часто використовуваними є ПЗ MS Office, а серед математичних пакетів – Mathcad та MatLab.

Після аналізу аналогів офісних додатків та математичних пакетів для розробки мобільного курсу обрано наступні ПЗ: для опрацювання текстових документів, електронних таблиць і презентацій

– WPS Office (характеристики: вільне поширення, хмаро орієнтованість та максимальне наближення до оригінального відображення документів, що розроблені у MS Office); математичні пакети – Scilab (характеристики: широка функціональність, безкоштовність та хмаро орієнтованість).

Напрямок подальшої роботи є висвітлення результатів виконання третього завдання – розробки робіт, що містять теоретичний матеріал, завдання для закріплення матеріалу, покращення навичок користування обраними ПЗ, тестових завдань та додаткових матеріалів.

ЛІТЕРАТУРА

1. EdEra – студія онлайн-освіти [Електронний ресурс]. – Режим доступу :<https://www.ed-era.com/courses/>
2. Prometheus – Найкращі онлайн-курси України та світу [Електронний ресурс]. – Режим доступу : <https://prometheus.org.ua/>
3. Word та Excel: інструменти і лайфхаки | Prometheus [Електронний ресурс]. – Режим доступу : https://courses.prometheus.org.ua/courses/course-v1:DNU+PRIN-101+2017_T1/about
4. Шишкіна М. П. Сучасні тенденції формування і розвитку науково-освітнього середовища вищого навчального закладу // М. П. Шишкіна / Хмарні технології в освіті : матеріали Всеукраїнського науково-методичного Інтернет-семінару (Кривий Ріг – Київ – Черкаси – Харків, 21 грудня 2012 р.). – Кривий Ріг : Видавничий відділ КМІ, 2012. – С. 7-8.

Гаєвський Б.С.,

Криворізький національний університет

Кузнєцов Д. І.

к.т.н., доцент, Криворізький національний університет

АВТОМАТИЗОВАНА СИСТЕМА КЕРУВАННЯ РОБОТОМ-ПІЛОСОСОМ

В даній статті проаналізовано використання засобів іот для створення автоматизованої система керування роботом-пилососом.

На сьогоднішній день кожен замислювався над тим щоб облегшити своє життя різноманітними пристроями, починаючи від звичайного електрочайника, закінчуючи пилососом. Вони потребують для управління людського втручання але не завжди є час на кип'ятіння води та прибирання оселі.

З початку 2000-х років багато компаній стали виробляти так звані «роботизовані пилососи», які оснащені штучним інтелектом на рівні простого автомата для його праці. Їх режими роботи були не такими різноманітними як на сьогоднішній час. Зараз же є такі налаштування як розпорядок, час і тривалість прибирання. Також вони оснащені одним з різновидів керування по радіо чи інфрачервоному каналу передачі, але їх дальність керування залишилася на рівні поля зору, щоб змінити налаштування прибирання потрібно задати режим кнопками на самому пристрої, що є не завжди зручним.

У сучасному світі все частіше зав'язується на смартфоні, онлайн покупки, оплата послуг, замовлення їжі та інше практично в будь-якій точці планети. Багатьом людям спадало на думку, щоб було б доцільно використовувати смартфон для керування пристроїв, і на теперішній час вже існує така концепція як «Інтернет речей».

Інтернет речей від англійської (Internet of things, IoT) - це концепція мережі фізичних пристроїв «речей», оснащених вбудованими технологіями для взаємодії один з одним по мережі Інтернет або локальній мережі. Сама концепція була ще сформульована в 1999 році як осмислення в перспективі широкого застосування засобів радіочастотної ідентифікації для взаємодії фізичних предметів між собою і з зовнішнім середовищем, яке здатне перебудувати економічні та суспільні процеси, що виключає з частини дій і операцій необхідність участі людини.

Починаючи з 2010-х років, вважається стійкою тенденцією в інформаційних технологіях наповнювати концепцію різноманітними технологічними і практичними рішеннями для її реалізації. Перш за все, завдяки розповсюдженню бездротових мереж, появи хмарних обчислень, розвитку технологій. Взаємодія між пристроєм

IoT та смартфоном відбувається за рахунок зв'язку з сервером (брокером). Сервер (брокер) виступає в ролі посередника для зберігання та передачі даних.

При використанні концепції IoT в виграші можуть бути наступні напрямки, такі як: розумні пристрої, розумні будинки, транспорт, логістика, обслуговування.

Колись було складно додати функцію підключення до інтернету в домашні прилади, але за останні декілька років складність та витрати на все це значно зменшилась. Сьогодні завдяки мікроконтролерам типу Arduino можна перетворити будь-який предмет у будинку на розумний пристрій.

Інтернетом речей зараз займаються вже сотні компаній та їх список подовжується с кожним роком, вони готові представити для всіх свої сервери та ресурси для створення IoT пристроїв. Ось кілька найбільших платформ IoT, які зараз діють на цьому ринку:

2. — Amazon Web Services;
3. — Microsoft Azure;
4. — IBM's Watson;
5. — Cisco IoT Cloud Connect;
6. — Oracle Integrated Cloud.

Але є також і недоліки даної концепції, наприклад, вбудувавши комп'ютер у звичні нам речі, компанії перетворюють весь всесвіт у суцільну загрозу безпеці комп'ютера, а у такому роботизованому світі всі зломи не тільки вплинуть на ваші дані, але й поставлять під загрозу вашу власність, життя і навіть національну безпеку. Захопивши пристрій, зловмисник має не тільки доступ до самого пристрою, але й камері або мікрофону який підключений в даний момент.

Ці платформи надають захищене підключення до їх мережі IoT, що значно зменшує ризик взлому зловмисником. Компанії с кожним днем поліпшують безпеку доступу до пристрою та сервера.

ВИСНОВОКИ

Отже, для створення автоматизованої системи керування робота-пилососа з будь-якої точки планети достатньо приєднати його до цієї мережі IoT за допомоги програмно-апаратного комплексу Arduino, що буде підключений до мережі інтернет. Який в свою чергу буде приймати команди з смартфона. «Розумні гаджети» вже

зараз роблять наше життя більш зручним. А в майбутньому ви можете всього лише по голосовій команді з іншої кімнати змусити мікрохвильовку розігріти обід. Підключення звичних побутових речей до інтернету може принести суспільству великі переваги, але водночас, потягнуть за собою і недоліки.

ЛІТЕРАТУРА

1. Конференція Інтернет вещей. URL: <https://iotconf.ru/>. (дата звернення: 21.02.2020).
2. До яких небезпечних наслідків може призвести інтернет речей. URL: <https://www.imena.ua/blog/internet-of-things/>. (дата звернення: 21.02.2020).
3. Что такое IoT, или интернет вещей. URL: <https://coinspot.io/beginners/chto-takoe-iot-ili-internet-veshhej/>. (дата звернення: 21.02.2020).

*Бойко Д. В.,
Криворізький національний університет
Кузнєцов Д.І.,
к. т. н., доцент, Криворізький національний університет*

МЕТОДИ ТА ЗАСОБИ ВИЗНАЧЕННЯ ПРОСТОРОВОЇ ПОЗИЦІЇ РУК НА ОСНОВІ ВИКОРИСТАННЯ СУЧАСНИХ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ

Проаналізовано методи для реалізації проекту з визначення просторової позиції рук та приведений опис кожного з методів з необхідною для реалізації апаратно-програмної бази, приведені приклади застосування цих методів.

На даний момент для реалізацію проекту з визначення просторової позиції рук розглядається три методи та засоби:

- 1) Leap Motion;
- 2) Microsoft Kinect Xbox 360;
- 3) Мікроконтролер Arduino Uno з датчиками вигину та гіроскопом.

1. Leap Motion – це технологія, заснована на захопленні руху, для людино-комп'ютерної взаємодії. Даний пристрій має невеликий

розмір та підключається до ПК за допомогою USB, він створює 3D-область взаємодії близько 277 дециметрів кубічних (куб зі сторонами 61 см.) та відслідковує рухи пальців та рук, олівців, паличок, тощо з великою точністю [1]. За допомогою цього пристрою можливо керувати, наприклад, міні-маніпулятором через той самий мікроконтролер Arduino чи Raspberry Pi та серводвигуни. Але для того що за допомогою Leap Motion можна було керувати Arduino треба використовувати мову програмування Processing яка основана на мові програмування Java та має подібний синтаксис до мови програмування C та власну IDE яка має схожий інтерфейс із Arduino IDE. На Arduino має бути завантажений скетч для прийому сигналу від Leap Motion на яку теж завантажений скетч на мові Processing, що відповідає за обробку та передачу відслідкованої позиції рук у просторі [2]. Недоліком даного методу є її складність у реалізації та ціна самого пристрою Leap Motion .

2. Microsoft Kinect Xbox 360 – це безконтактний ігровий контролер який був розроблений компанією Microsoft для ігрових приставок Xbox 360, Xbox One та пізніше для ПК під керуванням ОС Windows. Він складається з двох сенсорів глибини та кольорової камери. За допомогою спеціального ПЗ здійснюється повне 3D розпізнавання рухів тіла, міміки, лица. Датчик глибини складається з інфрачервоного проектора, об'єднаного з монохромного КМОН-матрицею, що дозволяє датчику Kinect отримувати тривимірне зображення при будь-якому природному освітленні. Діапазон глибини і програма проекту дозволяє автоматично калібрувати датчик з урахуванням навколишніх умов, наприклад меблів, що знаходиться в кімнаті. Одним із простих варіантів його використання у якості пристрою для визначення просторової позиції рук є використання мікрокомп'ютеру Raspberry Pi з встановленою на нього ОС Ubuntu 14.04.1 LTS та ПЗ ROS. ROS – це фреймворк для програмування роботів, що надає функціонал для розподіленої роботи [3]. Другим варіантом є мікроконтролер Arduino та ПК під керуванням ОС Windows та встановленим ПЗ Kinect SDK за допомогою якого пишеться код для керування самим Kinect та передається за допомогою технології блютуз на Arduino. Недоліком даного методу, як і у попереднього, є велика ціна, проте перевагою є те що при використанні Kinect можна продовжити розвиток на відслідковування позиції всього тіла у просторі [4].

3. Arduino – це платформа для аматорського конструювання, основним компонентом є плата мікроконтролеру зі елементами аналогового та цифрового вводу/виводу для підключення різних датчиків та модулів. Основні вузли пристрою – сама Arduino, датчики вигину та гіроскоп. Все це буде встановлено на будівельну рукавичку. На рукавичці встановлені датчики вигину: змінні резистори, які змінюють свій опір при вигині. Вони підключені до однієї сторони подільника напруги і постійним резисторам. Arduino зчитує зміну напруги при вигині датчиків і передає сигнал, наприклад, до серводвигунів, які пропорційно повертаються. Та для відслідковування позиції руки у просторі використовується модуль гіроскопу. Недоліком даного методу є невелика точність по відношенню до попередніх але він є найдешевшим та найлегшим по відношенню вище розглянутих методів. Перевагою є те, що має великий потенціал для удосконалення [5].

ВИСНОВКИ

Всі ці методи мають свої переваги та недоліки але всі вони можуть використовуватися у багатьох галузях, наприклад, у промисловості, у медицині, у дослідженнях, у космічній галузі і т.д і т.п. Вони дозволяють керувати багатьма механізмами та роботами знаходячись на відстані (наприклад для розмінування бомби).

Зараз технології від слідування позиції рук частіше можна зустріти у автівках компаній BMW, Mercedes-Benz, Volkswagen та Chevy яке використовується для керування мультимедією але на мою думку, в цієї технології великий потенціал для впровадження її на виробництво.

ЛІТЕРАТУРА

1. Офіційний сайт Leap Motion [Електронний ресурс] – Режим доступу до ресурсу: <https://developer.leapmotion.com/>.
2. Офіційний сайт Processing [Електронний ресурс] – Режим доступу до ресурсу: <https://processing.org/>.
3. Использование камеры Microsoft Kinect 360 в ROS Indigo [Електронний ресурс] – Режим доступу до ресурсу: <https://habr.com/ru/post/396291/>.

4. Монк С. Raspberry Pi. Сборник рецептов: решение программных и аппаратных задач / Саймон Монк., 2018. – 528 с. – (2). – (O'Reilly (Animals)).
5. Петин В. Проекты с использованием контроллера Arduino / Виктор Петин., 2015. – 464 с. – (2).

Т.В. Смірнова
к.т.н., докторант,
Центральноукраїнський національний технічний університет
Л.І. Поліщук
старший викладач,
Центральноукраїнський національний технічний університет
О.А. Смірнов
д.т.н., професор,
Центральноукраїнський національний технічний університет

АНАЛІЗ ХМАРНИХ ТЕХНОЛОГІЙ ЯК СЕРВІСІВ

Проведений розширений аналіз хмарних обчислень як сервісів. Серед наведених сервісів виділено базові сервіси IaaS, SaaS, PaaS, які є основою для існування більш уніфікованих сервісів (SaaS, MaaS, DaaS, FaaS, IPaaS, MBaaS, NaaS, SeSaaS, DBaaS, MaaS, GaaS, STaaS, TaaS, DRaaS), що збільшують область використання. Визначено, що для того, щоб перенести роботу систем інженерних розрахунків та систем автоматизованого проектування (САПР) на хмарну платформу, з'явився досить новий перспективний сервіс CAEaaS (англ. Computer Aided Engineering as a Service) – комп'ютерні системи інженерного аналізу як сервіс. На думку авторів статті, за хмарним сервісом CAEaaS майбутнє української промисловості.

Будь-яку послугу, що надається користувачеві за запитом через Інтернет із серверів постачальників хмарних обчислень будемо називати *хмарним сервісом* (as a Service, aaS). Таким чином хмарні сервіси саме і створені для того, щоб забезпечити простий, масштабований доступ і повністю керуються постачальником хмарних послуг. З метою задоволення потреб своїх користувачів хмарний сер-

віс може динамічно масштабуватись, тим паче, що компанії не турбуються про власні ресурси і ІТ-персонал, оскільки постачальник хмарних послуг забезпечує для цього необхідну апаратуру та програмне забезпечення. Для ефективного вибору платформи для розміщення технологічного забезпечення підприємства, а саме систем інженерних розрахунків та систем автоматизованого проектування (Computer Aided Engineering as a Service (CAEaaS)), постає нелегке завдання мінімізації часу відповіді для критично важливих програм як корпоративних користувачів, так і масових клієнтів.

В рамках даної роботи були проаналізовані наступні хмарні обчислення як сервіси, які зазвичай називають аaS (від "as a Service", тобто "як сервіс", або "в вигляді сервісу"):

- Content as a service (CaaS) – або managed content as a service (MCaaS) – (керований контент як сервіс).

- Data as a service (DaaS) – дані як сервіс.

- Desktop as a service (теж DaaS) – робочий стіл як сервіс.

- Function as a service (FaaS) – функція як сервіс.

- Infrastructure as a service (IaaS) – інфраструктура як сервіс.

- Integration platform as a service (IPaaS) – інтеграційна платформа як сервіс.

- Mobile backend as a service (MBaaS) – мобільний сервіс як послуга або Backend as a Service (BaaS) – бекенд як сервіс.

- Network as a service (NaaS) – мережа як сервіс .

- Platform as a service (PaaS) – платформа як сервіс або application platform as a service (aPaaS) – платформа застосунків як послуга або послуга на основі платформи.

- Security as a service (SECaaS або SaaS) – безпека як сервіс.

- Software as a Service (SaaS) – програмне забезпечення як сервіс.

- Data Base as a Service (DBaaS) – база даних як сервіс.

- Information as a Service (теж IaaS) – інформація як сервіс.

- Integration as a Service (теж IaaS) – інтеграція як сервіс.

- Management або Governance as a Service (MaaS або GaaS) – адміністрування або керування як сервіс.

- Process as a Service (теж PaaS) – процес як сервіс.

- Storage as a Service (STaaS) – зберігання як сервіс.

- Testing as a Service (TaaS) – тестування як сервіс.

- Disaster Recovery as a Service (DRaaS) – аварійне відновлення як сервіс.
- Backup as a Service (BaaS) – резервне копіювання як сервіс.
- Monitoring as a Service (MaaS) – моніторинг як сервіс.
- Hardware as a Service (HaaS) – устаткування як сервіс.
- Communications as a Service (CaaS) – комунікація як сервіс.
- Container as a Service (Caas) – контейнер як сервіс.
- Resource as a Service (RaaS) – ресурс як послуга.
- Customer Relationship Management as a Service (CRMaaS) – керування взаємовідносинами з клієнтами як сервіс.
- Bookkeeping as a Service (BaaS) – бухгалтерський облік як сервіс.

В рамках даних сервісів з'являється досить новий перспективний сервіс CAEaaS (англ. Computer Aided Engineering as a Service) комп'ютерні системи інженерного аналізу як сервіс, який покликаний перенести роботу систем інженерних розрахунків та систем автоматизованого проектування (САПР) на хмарну платформу.

ВИСНОВКИ

У роботі, з метою визначення базових умов, щодо подальших досліджень проведений розширений аналіз хмарних обчислень як сервісів. Серед наведених сервісів виділено базові сервіси IaaS, SaaS, PaaS, які є основою для існування більш уніфікованих сервісів, що збільшують область використання. Визначено, що для того, щоб перенести роботу систем інженерних розрахунків та систем автоматизованого проектування (САПР) на хмарну платформу з'явився новий перспективний сервіс CAEaaS – комп'ютерні системи інженерного аналізу як сервіс.

Ананьєв В.С.

Криворізький національний університет

Кузнецов Д.І.

к. т. н., доцент, Криворізький національний університет

РОЗРОБКА МУЛЬТИМЕДІЙНОГО РОЗШИРЕННЯ ДЛЯ ВЕБ-БРАУЗЕРА НА ДВИЖКУ ХРОМІУМ

Виконано огляд та аналіз сучасних розширень для веб-браузера. Проаналізовано актуальність та можливість використання браузерних розширень для зменшення відкритих додатків та вкладок.

На сьогодні в мережі інтернет розваги та ЗМІ користуються найбільшою популярністю. Завдяки цьому кошти вилучені з релізів інтернет-додатків зростають щорічно. А через постійне збільшення пристроїв з доступом до інтернету прибутки зростуть ще більше.

Особливою популярністю користуються відеоігри, які потребують плати за встановлення та в подальшому пожертвування «донат».

На малюнку 1 зображено найбільші ринки розваг та ЗМІ.

Страна	Рынок развлечений и СМИ млн долл. США		Место в мире по объему рынка		СТГР*
	2012	2017	2012	2017	
США	498 884	632 094	1	1	4,8%
Япония	191 621	203 062	2	2	1,2%
Китай	115 280	202 765	3	3	12,0%
Германия	97 439	110 023	4	4	2,5%
Великобритания	85 548	103 864	5	5	4,0%

Рисунок 1 – Найбільші ринки розваг та ЗМІ

Музика – самий передовий цифровий сегмент з точки зору продажів: у 2016 році глобальні продажі цифрової музики перевищили фізичні [1,2].

Майже усі люди у сучасному світі слухають музику щоденно. На смартфоні, плеєрі, персональному комп'ютері, в інтернеті або на власному гаджеті. Навіть під час поїздки в метро або в торговому комплексі, у ресторані чи кафе. Музика стала для нас невід'ємною частиною життя. Більшість людей, які слухають музику запевняють, що вона покращує їхнє самопочуття, та допомагає креативно мислити.

ВИСНОВКИ

Таким чином, сформульовано основні вимоги до розроблювального мультимедійного розширення в якому користувач зможе

комфортно прослуховувати музику не виходячи з браузера. Це призведе до зменшення відкритих одночасно програмних додатків та до збільшення швидкої взаємодії з кожним із них.

ЛІТЕРАТУРА

1. Всесвітній огляд індустрії розваг і ЗМІ [Електронний ресурс] // Енциклопедія маркетингу. – 2018. – Режим доступу до ресурсу: <https://www.marketing.spb.ru>.
2. Огляд індустрії розваг і медіа: прогноз на 2019-2023 роки [Електронний ресурс] // PwC. – 2019. – Режим доступу до ресурсу: <https://www.pwc.ru/ru.html>.

*Чорний Р.В., Горбовцова І.В.
Харківський національний університет радіоелектроніки
Сердюк Н.М.
к.т.н., доц. каф. інформаційних управляючих систем,
Харківський національний університет радіоелектроніки*

КІБЕРФІЗИЧНІ СИСТЕМИ У СУЧАСНОМУ СВІТІ

Розглянуто нове покоління систем і методів управління, які допомагають розвивати перспективні міждисциплінарні дослідження – кібер-фізичні системи. Вони впливають на майбутні промислові та соціальні системи процесів збору та обробки даних вимірювань у режимі реального часу.

Кіберфізична система (cyber-physical system - CPS) - складна розподілена система, керована або контрольована комп'ютерними алгоритмами, тісно інтегрована з Інтернетом і його користувачами. Її технологічною основою стала технологія інтернет речей (Internet of Things, IoT). «Мозок» системи у вигляді II та інших технологій отримує дані від сенсорів в реальному світі, аналізує ці дані і використовує їх для подальшого управління фізичними елементами. Завдяки такій взаємодії кіберфізична система здатна ефективно працювати в умовах, що змінюються, як аналог людського організму або сучасна компанія, яка аналізує ситуацію на ринку, щоб розробити саме той продукт, який йому зараз потрібен [1]. Причому

цикл «управління - отримання даних - обробка даних - управління» при налагодженій роботі системи кожен раз повинен давати позитивні результати і створювати нову цінність.

У фізичному контексті виконавчі елементи на основі одержуваних даних вносять зміни в середовище проживання користувачів. У віртуальному контексті кіберфізичні системи застосовуються для збору даних про віртуальних діяч користувачів, таких як використання соціальних мереж, блогів і сайтів електронної комерції. Потім кіберфізичні системи певним чином реагують на такі дані, прогножуючи дії або потреби користувачів в цілому. Використовуючи такі програмні продукти, як IBM WebSphere Sensor Events, можна аналізувати дані і події, що надходять від датчиків в реальному часі, і вбудовувати їх в інтелектуальні рішення.

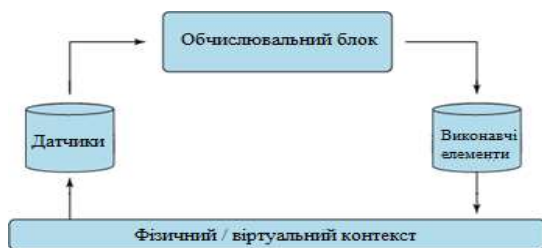


Рис1. Архітектура кіберфізичної системи

З технічної точки зору треба буде ще розв'язати безліч складних проблем - як мінімум ефективним і прийнятним в реальних умовах способом:

- **Різномірність даних.** Різномірність даних - це серйозна проблема, яка може негативно впливати на ефективність взаємодій і розробку комунікаційних протоколів. Системи повинні бути здатні підтримувати велику кількість різних додатків і пристроїв.

- **Управління даними.** Необхідно зберігати і аналізувати великі дані, що надходять від різних мережевих пристроїв, обробляти їх і в реальному часі виводити результати. Даними можна керувати за використанням відкладеної або оперативної потокової обробки, в залежності від призначення системи. При використанні потоків в реальному часі інформація може часто змінюватися і обробка ґрунтується на адаптивних і постійних запитах.[2]

ВИСНОВКИ

Кіберфізичні системи, що є рушійною силою інновацій, охоплюють безліч різних дисциплін. Співпраця різних галузей може зробити їх важливою виробничою силою. Кіберфізичні системи мають величезний потенціал для зміни і вдосконалення кожного аспекту життя людей, допомагаючи вирішувати критично важливі для нашого суспільства проблеми і перевершуючи сучасні розподілені системи в плані безпеки, продуктивності, ефективності, надійності, зручності використання і за багатьма іншими показниками.

ЛІТЕРАТУРА

1. Городецький В.І. Сучасний стан та перспективи індустріальних застосувань багатоагентних систем / Городецький В.І., Бухвалов О.Л., Скобелев П.О. // Управління великими системами: збірник праць. - 2017. - №66. - С. 97-157.
2. Загітова А. І. Система підтримки життєвого циклу складного технічного об'єкта на основі агентних технологій / Загітова А. І., Кондратьєва Н. В., Валєєв С. С. // Вісник УГАТУ. - 2018. - Т. 22. - №. 2 (80). - С. 113-121

*Кобас А. І.
Криворізький національний університет
Рябчина Л. С.
асистент, Криворізький національний університет*

СИСТЕМА БЕЗПЕКИ ПРОТОКОЛІВ ОБМІНУ ДАНИМИ У МЕРЕЖАХ ІОТ

Розглянуто актуальність використання систем безпеки протоколів обміну даних у мережах Інтернету речей. Проведено огляд існуючих технологій безпеки Інтернету речей. Наведено класифікацію систем зв'язку та основні принципи кібербезпеки.

З розвитком інформаційно-комунікаційних технологій, автоматизації та роботизації промислових процесів людина перейшла у час цифрової революції.

Інтернет речей (Internet of Things, IoT) – це мережа фізичних об'єктів та пристроїв, які мають вбудовані технології та програмне забезпечення, що дозволяють здійснювати взаємодію з зовнішнім

середовищем, передавати відомості про свій стан і приймати дані ззовні, за допомогою використання стандартних протоколів зв'язку [1]. Така комплексна мережа взаємопов'язаних пристроїв дозволяє виконувати зчитування та приведення цих пристроїв у дію. За рахунок використання інтелектуальних інтерфейсів, стало можливе виключення необхідності участі людини в програмуванні та ідентифікації.

Швидке поширення IoT сприяло звертанню уваги на більш детальний розгляд проблеми конфіденційності інформації, що послужило появі поняттю безпека інтернету речей.

В даній роботі перед проектуванням власної системи безпеки передачі даних (СБПД), треба спочатку провести аналіз існуючих технологій, протоколів зв'язку та систем тощо.

Велику увагу при розробці IoT та його розвитку звертається на завдання встановлення з'єднання, зв'язку та налаштування роботи мережі. Передача даних і встановлення мережевого з'єднання будуються за принципом систем зв'язку ближньої дії – персональних мереж (PAN), які будуються без встановлення правил IP-протоколу. Дані мережі можуть бути дротовими та бездротовими. До бездротових IoT-мереж/протоколів можна віднести протоколи Zigbee, Z-Wave, mesh-мережі, Bluetooth. Для IIoT (Industrial IoT) відносять Wireless Hart та ISA100. Список дротових мереж більший, ніж бездротових, оскільки до даного переліку відносять всі існуючі промислові мережі/протоколи.

Крім персональних мереж використовуються бездротові локальні мережі та системи зв'язку на основі IP-протоколу, включаючи до даного переліку широкий діапазон Wi-Fi-мереж на базі стандартів IEEE 802.11, 6LoWPAN і технології Thread [2].

Для передачі даних датчиків до Інтернет-простору для даної СБПД необхідні дві технології: маршрутизатор-шлюз та опорні інтернет-протоколи, які забезпечують ефективність обміну даними. Маршрутизатор виконує важливі функції у сферах безпеки, управління та напряму самих даних. Граничні маршрутизатори (Edge routers) керують і стежать за станом відповідних mesh-мереж, а також вирівнюють і підтримують якість даних. Також велике значення належить конфіденційності та безпеки даних. Маршрутизатор відіграє важливу роль в створенні віртуальних приватних ме-

реж, віртуальних локальних мереж і програмно-визначених глобальних мереж. Дані мережі можуть містити тисячі вузлів, що обслуговуються єдиним граничним маршрутизатором, і в якійсь мірі маршрутизатор служить розширенням для хмари (edge device) [2].

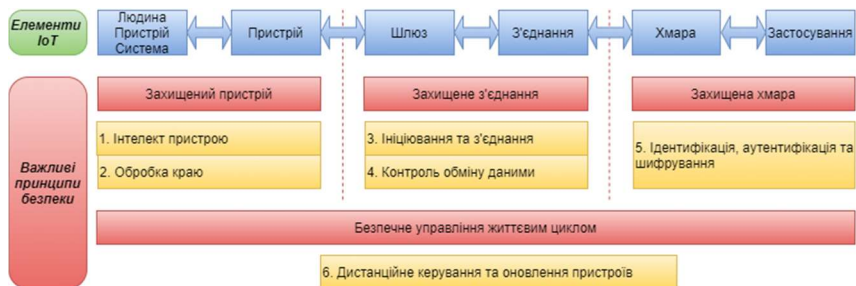


Рис. 1 – Схема основних принципів кібербезпеки в IoT

За типом реалізації даної системи доречно обрати систему загального призначення, так як вона не буде заглиблюватись у специфіку однієї галузі, але дасть загальне представлення основних принципів кібербезпеки обміну даними в IoT (рис. 1), що є більш важливим при бажанні працювати в цій галузі у майбутньому.

ВИСНОВОК

На основі проведеного аналізу можна зробити висновок, що питання безпеки протоколів обміну даними в мережах IoT має велике значення, адже втрата конфіденційності інформації може завдати великих витрат. Захист передачі даних та встановлення безпечного з'єднання мережі повинен здійснюватися комплексно за основними принципами кібербезпеки та в різних напрямках. Чим повніше та досконаліше буде виконано аналіз призначення типу мережі та встановлено її з'єднання, тим стійкіше та безпечніше матиме захист сама система.

ЛІТЕРАТУРА

1. Industrial Internet of Things – ПоТ Промышленный интернет вещей [Електронний ресурс] – Режим доступу до ресурсу: http://www.tadviser.ru/index.php/Статья:IIoT_-Industrial_Internet_of_Things.

2. Understanding IoT Security – Part 2 of 3: IoT Cyber Security [Електронний ресурс] – Режим доступу до ресурсу: <https://iot-analytics.com/understanding-iot-cyber-security-part-2>

Голіков В.В.

Криворізький економічний інститут Київського національного університету ім. Вадима Гетьмана

Вдовиченко І.Н.

К.т.н., доцент, Криворізький національний університет

ВИКОРИСТАННЯ БЕЗСЕРВЕРНИХ ОБЧИСЛЕНЬ У WEB-ДОДАТКАХ

Проаналізовано перспективи безсерверних обчислень та serverless архітектури додатку. Огляд сильних та слабких сторін використання безсерверних технологій при розробці web-додатків.

Останнім часом хмарні технології набирають все більшої популярності. Це відбувається з простої причини - легкої доступності, відносної дешевизни і відсутності початкового капіталу - як знань для підтримки і розгортання інфраструктури, так і фінансового характеру. Serverless - безсерверна архітектура додатків. Насправді, не така вже вона й безсерверна. Основу архітектури складають мікросервіси, або функції (lambda), що виконують певне завдання і запускаються на логічних контейнерах, захованих від сторонніх очей. Тобто кінцевому користувачеві дано тільки інтерфейс завантаження коду функції (сервісу) і можливість підключення до цієї функції джерел подій (events). Іноді безсерверні обчислення також іменують «Функція як послуга», тому що одиницею коду є функція, яка виконується платформою. По суті для виконання одного запиту створюється окремий контейнер, який знищується після виконання.

Основними платформами, що надають послуги безсерверного середовища виконання є такі як AWS Lambda від Amazon, Google Cloud Functions від Google, OpenWhisk від IBM, Azure Functions від Microsoft Azure та інші [1].

Ідея безсерверних обчислень полягає в чотирьох рисах: абстракція, еластичність, ефективна вартість та обмежений життєвий цикл. За допомогою абстракції користувач не керує сервером, на

якому запускається програма. Він взагалі нічого не знає про сервер, усі нюанси операційної системи, оновлення, мережеві налаштування та інше заховане від користувача. Це зроблено для того, щоб розробники могли сконцентруватися на розробці корисного функціоналу, а не адмініструванні серверу. Еластичність виражається в тому, що провайдер безсерверних обчислень автоматично виділяє більше чи менше обчислювальних ресурсів в залежності від того, на скільки велике навантаження приходить на додаток. З цього також впливає і ефективна вартість, якщо додаток простояє — розробник нічого не платить, тому що в цей момент додаток не використовує обчислювальних ресурсів. Додаток запускається у контейнері і через короткий час, від десятка хвилин до декількох годин, сервіс автоматично його зупиняє. Звісно, якщо додаток знову повинен бути викликаний — новий контейнер буде запущено.

При деяких допущеннях, Serverless архітектура може бути використана для будь яких проєктів. Однак є випадки в яких її використання легше та безпечніше — відкладені та фонові задачі. Приклади таких задач: створення резервної копії за розкладом; асинхронна відправка повідомлень користувачам (push, email, sms); різноманітні експорти та імпорти і т.д. Усі ці приклади виконуються по розкладу або не мають на увазі моментальної відповіді користувачу. Це пов'язано з тим, що додатки (функції) в безсерверній моделі не працюють постійно, а запускаються при необхідності, а у разі невикористання автоматично відключаються. Це призводить до того, що для запуску функції потребується час, іноді до декількох секунд [2].

Однак це не означає, що безсерверні обчислення неможливо використовувати в частинах додатку, з якими взаємодіють користувачі і для яких важливий час відповіді. Навпаки, Serverless функції широко використовуються у чат-ботах, бекенд частині для «інтернету речей», маніпуляція запитами до основного бекенду (наприклад, для ідентифікації користувача чи отримання інформації про його геопозицію через IP-адресу).

Перевагами використання Serverless архітектури можна виділити: максимальну еластичність, швидке масштабування з нуля до тисяч паралельно працюючих функцій; повна абстракція від операційної системи чи будь-якого програмного забезпечення, що вико-

ристовується для виконання додатку; при правильному проектуванні функцій легше побудувати слабо-зв'язану архітектуру, при якій помилка в одній функції не вплине на працездатність всього додатку.

Недоліків у такої архітектури багато, основними є необхідність турбуватись про збереження зворотної сумісності; схема взаємодії в класичному монолітному додатку і в розподіленій системі сильно відрізняється, необхідно думати про асинхронну взаємодію, можливих затримках, моніторингу окремих частин додатку; неправильна архітектура може призвести до того, що помилка в одній функції призведе до непрацездатності великої кількості інших; запуск функції може потребувати до декількох секунд, що може бути критично; коли запит від користувача проходить через десяток функцій важко відловити можливу причину помилки, якщо така стається; прив'язка до провайдера хмарних обчислень, функції розроблені для AWS буде важко перенести на, наприклад, Google Cloud Platform, через те що функції рідко використовуються в ізоляції, окрім них використовуються бази даних, черги повідомлень, системи логування та ін., що відрізняються від провайдера до провайдера [3].

Підсумовуючи можна сказати: хоч мінусів вийшло більше ніж плюсів, це не означає, що «Function as a Service» поганий напрям, зовсім навпаки, багато ризиків можуть бути мінімізовані, або сприйняті як факт.

ЛІТЕРАТУРА

1. AWS Serverless Application Model (AWS SAM) [Електронний ресурс] – Режим доступу: <https://docs.aws.amazon.com/serverless-application-model/>
2. AWS Lambda и никаких серверов. [Електронний ресурс] – Режим доступу: <https://habr.com/ru/post/245949/>
3. Learn to build full-stack Serverless apps. [Електронний ресурс] – Режим доступу: <https://serverless-stack.com/>

*Рябчина Л.С.
аспірантка, Криворізький національний університет
Кузнєцов Д.І.
к.т.н., доцент, Криворізький національний університет*

ВЗАЄМОДІЯ ГЕТЕРОГЕННИХ ПРИСТРОЇВ ІНТЕРНЕТУ РЕЧЕЙ

Розглянуто мережі Інтернету речей, їх складові та принципи взаємодії. Проаналізовано проблематику галузі та її актуальність. Наведено напрямки подальших досліджень.

Сьогоденність можна відзначити поширенням автоматизації всіх сфер життя людини, а саме застосування мереж Інтернету речей (Internet of Things, IoT). Сюди можна віднести розумні будинки, переносні пристрої, системи автоматизованого виробництва тощо.

Вищенаведені приклади можуть містити у собі різноманітні складові для повних збору та обробки даних за необхідними алгоритмами. Зазвичай для використання виробниками надається власний програмно-апаратний комплекс, що є однорідним у своїх межах, і не потребує додаткових умов для спільної роботи складових. Складність виникає при додаванні до вже працюючої системи нового пристрою, що є відмінним від складових існуючої мережі.

Традиційна модель взаємодії пристроїв Інтернету речей має наступні властивості:

1. розрізнені мережі;
2. система зв'язку Master-Slave;
3. блокування від постачальника;
4. обмежені можливості підключення [1].

Виходячи з вищенаведеного можна зробити такий висновок, що взаємодія між відмінним обладнанням є або ускладненою, або ж й взагалі неможливою. Таке обладнання може відрізнитись між собою як за апаратним складом, так і за принципами програмної обробки й передачі даних, тобто бути гетерогенним.

При більш детальному розгляді питання впливає глобальна проблематика у вигляді мереж Інтернету всього (Internet of Everything, IoE).

На рис.1 наведено, як розвивались інформаційні технології у напрямку, що розглядається, та показано, як Інтернет речей переходить у Інтернет всього [2].



Рисунок 1 – історичний розвиток інформаційних технологій

Оскільки проблема стандартизації інформаційного обміну між гетерогенними пристроями мережі Інтернету речей й досі є недостатньо опрацьованою, то й при переході до Інтернету всього можна стикнутись із тими самими проблемами.

Для вирішення цього питання необхідним є прийти до глобальної стандартизації пристроїв, з яких мають складатись вищенаведені мережі.

При розробці нових моделей та засобів інформаційного обміну між неоднорідними обчислювальними системами у межах промислових мереж Інтернету речей слід також враховувати не тільки способи поєднання такого обладнання, але й наступні фактори, що є не менш важливими: підвищений рівень безпеки, масштабованість, засоби керування, оптимізація трафіку тощо [3].

ВИСНОВОК

Узагальнюючи, можна сказати, що галузь інтернету речей потребує глобальної стандартизації з метою коректного об'єднання гетерогенних пристроїв у єдині мережі. Таким чином подальшим напрямом досліджень у цьому напрямку може бути розробка нових моделей інформаційної взаємодії, або ж удосконалення вже існуючих інструментів інформаційного обміну.

ЛІТЕРАТУРА

1. Kip Compton, Vikas Butaney. Enabling the Internet of Everything: Cisco's IoT Architecture: slides from the Cisco Live. Milan, 2015.
2. Dave Evans. The Internet of Everything. Cisco IBSG. 2018.
3. The Internet of Things World Forum 2017. URL: <http://www.iotwf.com/>

*Саянін В. Г.,
аспірант, Криворізький національний університет
Кузнєцов Д. І.
к.т.н., доцент, Криворізький національний університет*

СТРУКТУРА СИСТЕМИ КЕРУВАННЯ В MICROGRID

Визначено відмінність microgrid від традиційних електричних мереж. Розглянуто особливості представлення систем керування microgrid та досліджено різні архітектури систем керування microgrid.

Згідно Енергетичної стратегії України, одним з пріоритетних напрямів є сприяння розвитку розподіленої генерації, збільшення частки джерел відновлювальної енергії та впровадження smart-grid до 2035 року. І наріжним каменем цього питання є microgrid. Microgrid – це електрична мережа (ЕМ), що поєднує в собі споживачів, виробників, засоби накопичення електричної енергії та підтримує два стани роботи: ізольований та під'єднаний до основної ЕМ. Вона ефективно включає в себе засоби розподіленої генерації і також є будівним елементом для smart-grid. Microgrid заснована на тісному поєднанні трьох видів технологій: інформаційних, телекомунікаційних та енергетичних. Вони надають їй особливі характеристики у порівнянні з класичною ЕМ: здатність до самовідновлення, повсюдний збір даних на основі великої кількості датчиків, використання цифрових та інтелектуальних засобів керування. Але через дані особливості є і нові недоліки: брак інерційності, що призводить до відхилень частоти та напруги, збурення в ЕМ при зміні режимів роботи, спотворення форми сигналу інверторами, складність прогнозу виробки енергії ДВЕ. Тому microgrid потребує більш тонкої та гнучкої системи автоматизованого керування.

Звичайна ЕМ має централізовану та ієрархічну однонаправлену систему керування. Вона керується інформацією про загальне поточне навантаження та прогнозом на ближній горизонт і не бере до уваги інформацію про стан кінцевих ліній споживача. Зазвичай вона представлена в такій послідовності:

- 1) генерація електроенергії;
- 2) система транспортування електроенергії;
- 3) система розподілення електроенергії;
- 4) споживач.

У microgrid енергія та інформаційні потоки проходять в двох напрямках, і здійснюється керування не тільки розподілом енергії і генерації, а ще й навантаженням, що дозволяє ефективно використовувати ресурси системи. Через це microgrid потребують інших парадигм керування.

Зазвичай система керування microgrid представлена трьома рівнями, див. табл. 1 [1]:

Таблиця 1 - Структура системи керування microgrid

Третинний рівень	Керує обміном енергії з іншою зовнішньою ЕМ. Швидкість реакції може сягати декілька годин.
Вторинний рівень	Відслідковує проблеми в середині ЕМ, синхронізує microgrid з зовнішньою мережею. Швидкість реакції складає декілька хвилин.
Первинний рівень	Підтримує напругу та частоту. Швидкість реакції повинна складати менше секунди.

Саме керування здійснюється за декількома методами:

1. Ведучий – ведений. Доки ведучий контролює напругу та частоту, ведений керує джерелом електричного току.
2. Використання сигналів керування для управління током та потужністю.
3. Метод статички. Може поєднувати останні два методи.

В залежності від розподілення зв'язків між засобами генерації енергії, навантаженням, контролерами, акумуляторами та іншими

елементами microgrid, розрізняють такі архітектури системи керування ЕМ [2]:

1. Централізована архітектура. Вона полягає в тому, що в microgrid існує центральний контролер, що отримує всю інформацію о стані ЕМ і може ефективно нею керувати. Ця архітектура дозволяє отримати високий рівень продуктивності для всієї системи в цілому, але це пов'язано з великою складністю обчислень через велику кількість інформації та появою в ЕМ точки відмови у вигляді центрального контролера. При його виході з ладу перестає працювати весь microgrid
2. Ієрархічна архітектура. Дана архітектура дещо схожа з централізованою, але додається новий елемент, відомий як агрегатор, що збирає дані о навантаженні та виробі енергії і оптимізує їх, далі він надсилає ці дані до центрального контролера, який пов'язаний з центральним координатором. Центральний координатор – це елемент, що дозволяє перерозподіляти ресурси між декількома microgrid. Коли центральному контролеру недостатньо ресурсів для оптимальної роботи microgrid, тоді центральний контролер подає запит до центрального координатора, щоб той надав ресурси з інших microgrid.
3. Розподілена архітектура. В даному випадку всі елементи microgrid або їх об'єднання отримують контролери. Оптимізація роботи елементів проходить локально, що дозволяє власникам розподілених джерел енергії бути більш незалежними, але вони все ж координуються центральним контролером.
4. Архітектура точка-точка. В даній архітектурі всі обчислення повинні проводитися локально, відсутня єдина точка відмови у вигляді центрального контролера. Це підвищує надійність системи, але в той самий час ускладнюється розподільча та комунікаційна система через збільшення кількості зв'язків, і збільшуються вимоги до локальних контролерів.

ВИСНОВКИ

В даній роботі було розглянуто перспективність впровадження microgrid в Україні. Визначено проблеми які виникають у

microgrid. Також досліджено відмінність звичайної ЕМ від microgrid, особливості представлення систем керування microgrid та різні архітектури систем керування microgrid.

ЛІТЕРАТУРА

1. Bordons, C.; Garcia-Torres, F.; Ridao, M.A. Model Predictive Control of Microgrids; Springer: Cham, Switzerland, 2019.
2. Almasalma H. Peer-to-Peer Control of Microgrids [Електронний ресурс] / Н. Almasalma, J. Engels, G. Deconinck. – 2016. – Режим доступу до ресурсу: https://www.researchgate.net/publication/307594134_Peer-to-Peer_Control_of_Microgrids.

*Швец Д.В.,
старший преподаватель,
Криворожский национальный университет
Азарян А.А.,
д.т.н., профессор, Криворожский национальный университет*

АНАЛИЗ ХАРАКТЕРИСТИК ТЕХНОЛОГИЧЕСКОГО ПРОЦЕССА ИЗМЕЛЬЧЕНИЯ РУДОБОГАТИТЕЛЬНОЙ ФАБРИКИ

Проведен анализ характеристик технологического процесса измельчения рудобогагательной фабрики, выделены наиболее значимые факторы, обуславливающие применимость тех или иных методов управления им. Сделаны выводы о необходимости создания математической модели технологического процесса измельчения с целью синтеза управления.

Современный этап развития информационных технологий характеризуется широким внедрением в информационно-измерительные системы (ИИС) технологического процесса измельчения (ТПИ) приборов и устройств, позволяющих с помощью персонального компьютера с широким спектром стандартных программ разного назначения реализовать достаточно разнообразные алгоритмы управления ТПИ. Особое место при реализации алгоритмов управления занимают устройства, использующие ядерно-физические методы взаимодействия гамма-излучения с железорудным сырьем,

поскольку они отвечают основным требованиям к представлению информации. Вместе с тем, анализ ТПИ как объекта управления позволяет отнести его к категории сложных объектов. Действительно, ТПИ присущи черты, характерные сложной системе [1]. Прежде всего, необходимо отметить отсутствие математического описания ТПИ и необходимость в нем. Простым объектом можно управлять и без его математического описания, реализуя принцип обратной связи, опираясь, например, на методы нечеткой логики [2]. Однако, требование точности управления приводит к следующему утверждению: если нужно управлять сложным объектом с использованием формальных методов, то придется создать его математическую модель.

Далее естественно выделить стохастичность ТПИ, что затрудняет анализ процессов в нем и управление. Эта черта обусловлена не столько наличием каких-то специальных источников случайных помех в самом ТПИ, сколько сложностью ТПИ и связанных с этим всякого рода второстепенных процессов. Поэтому поведение ТПИ зачастую оказывается неожиданным для исследователя, причем эту неожиданность удобнее рассматривать как случайный фактор. Таким образом, при управлении ТПИ как сложным объектом целесообразно называть случайным то, что второстепенно и несущественно для реализации целей управления. Важной чертой является «нетерпимость» ТПИ к управлению, что затрудняет управление им. Это объясняется тем, что сложный объект существует и функционирует независимо от субъекта и его потребностей. Так, любой технологический процесс существует для производства продукции, а не для управления им. Управление имеет внешний характер по отношению к объекту. Вследствие этого всякое управление нарушает естественное функционирование объекта, то есть изменяет его самостоятельное поведение и делает зависимым от субъекта.

Нестационарность ТПИ естественно вытекает из его сложности. Эта черта проявляется в дрейфе его характеристик, в изменении его состояния, то есть в эволюции во времени. Следствием нестационарности ТПИ является невозможность воспроизводимости экспериментов. Проявляется это в различной реакции объекта на одну и ту же ситуацию при управлении в различные моменты времени. Сложный объект все время изменяется, что необходимо учитывать при синтезе модели объекта управления путем ее коррекции.

Отмеченные особенности ТПИ как сложного объекта приводят к тому, что цель управления в полной мере никогда не достигается. Действительно, для синтеза управления требуется время, за которое ТПИ изменяется непредвиденным образом, в результате это управление наверняка не приведет к желаемому результату. Эффективным способом борьбы с перечисленными свойствами ТПИ является экстраполяция его поведения, то есть выяснение направления эволюции. В этом случае управляющее воздействие наносится с упреждением с учетом замеченного изменения состояния, например, с помощью математической модели. Другой способ состоит в сокращении цикла управления, чтобы за время синтеза управления объект не успел значительно измениться.

ВЫВОДЫ

Отмеченные особенности ТПИ указывают на необходимость создания его математической модели с целью синтеза управления. С помощью модели ТПИ можно построить управление, переводящее ТПИ в требуемое (целевое) состояние. Без модели процесс управления можно реализовать лишь методом проб и ошибок, что неприемлемо при управлении сложным объектом, так как это требует значительных затрат времени и средств.

ЛИТЕРАТУРА

1. Симанков В. С. Адаптивное управление сложными системами на основе теории распознавания образов / В. С. Симанков, Е. В. Луценко. – Краснодар: техн. ун-т Кубан. гос. технол. ун-та, 1999. 318 с.
2. Рыжов А.П. Элементы теории нечётких множеств и её приложения / А.П. Рыжов. – М., Диалог-МГУ, 2003. 81с.

*Veselovsky Danylo, Ivaschenko Oleksii
Kryvyi Rih National University
Ishchenko Mykola
PhD, Kryvyi Rih National University*

MONITORING AND MANAGING SYSTEM OF MICROCLIMATE INDICATORS IN EDUCATIONAL FACILITIES

The algorithm and structure of an automated system for monitoring microclimate indicators in educational facilities are exposed. A model of for monitoring and managing microclimate system based on Phoenix Contact hardware and software has been developed. A user-defined web-interface for remote monitoring and management of the current layout from mobile devices has been developed.

One of the most serious problems is energy efficiency, namely, inefficient usage of energy resources, their excessive and short-sighted consumption. This problem makes us to think seriously about energy saving and energy efficiency, because the reserves of fuel raw materials are not unlimited [1,2].

Nowadays, the trend of using alternative energy sources and improvement of the efficiency of energy systems and their automation has been spreading for a long time. One of the most energy-consuming structures is the urban infrastructure of electric utilities for heating educational facilities, which modernization cannot be ignored, because it is an inherent part of modern technological progress [1-4].

The algorithm and structure of an automated microclimate control system in educational facilities are suggested by the authors. A unique feature of the suggested system is usage of multi-segment heat-insulated floor [5]. This system consists of separate sections of underfloor heating, each of which consists of a heating panel connected to a corresponding relay. As a result, the smart floor heating system does not heat the all segments of the room, only the part of it, namely where the person is located. Regardless of the type of heating panels used, even least powerful, the total energy consumption is very significant, and this technical solution provides significant energy savings [5,6].

The software of the automated ensuring optimal temperature system of educational facilities implements the following basic functions:

- data collection, processing and analysis;
- getting information from temperature and humidity sensors;
- user communication with the information collection and processing system;
- making decisions based on a given algorithm;
- sending a message to Web server;
- creating databases.

This system was implemented as a model and tested in the laboratory complex "ECOLOGICAL AND ENERGY-EFFICIENT AUTOMATED SYSTEM OF THE CITY ENTERTAINMENT CENTER BASED ON TECHNOLOGICAL SOLUTIONS OF THE PHOENIX CONTACT COMPANY" [7].

For the moment, the following functions are applied in the model:

- convenient change of operating modes;
- ability to control the system from any location;
- a web interface for monitoring and managing microclimate indicators in educational facilities is available;
- convenient scaling of the system, if necessary, its expansion;
- a wireless access module has been installed that allows monitoring and managing the automated system by using mobile devices;

In the future, it is planned to expand the capabilities of this system with the following components:

- usage of the cloud interface adapter CLOUD-COUPLER-PRO for collecting and storing information from all detections used in the complex and further processing;
- installation of an energy management system and emergency notification based on the PSK RTU 50 data logger.

CONCLUSION

The review of existing systems for monitoring the microclimate in educational facilities and the relevance of the selected topic has been described. The algorithm and structure of an automated system of monitoring microclimate indicators in educational facilities are proposed. A model of the existing system of monitoring and managing microclimate indicators based on Phoenix Contact hardware and software has been developed. A user-defined web-interface for remote monitoring and management of the current layout from mobile devices has been developed.

REFERENCES

4. Вирішення проблем енергоефективності у муніципальному секторі міст України [Електронний ресурс] / Копець Г.Р., 2009 р. Режим доступу: http://www.nbu.gov.ua/portal/natural/Vnulp/Ekonomika/2009_640/19.pdf.
5. Постанова Кабінету Міністрів України N 145-р «Енергетична стратегія України на період до 2030» [Електронний ресурс], Режим доступу: <http://mpe.kmu.gov.ua/fuel/control/uk/doccatalog/list?currDir=50358>.
6. Електричне опалення будівель : навч. посіб. / О. О. Савченко, Б.І. Щербатюк; М-во освіти і науки України, Нац. ун-т "Львів. політехніка". – Львів : Вид-во Львів. політехніки, 2014. – 160 с.
7. Конох И. С. Разработка и исследование интеллектуальной системы регулирования параметров микроклимата помещения / И. С. Конох, И. С. Гула, С.В. Сукач // Электромеханические и энергосберегающие системы. – 2010. – №3 (11). – С. 80–85.
8. Веселовський Д.В., Іващенко О.Р., Іщенко М.О. Автоматизована енергоефективна система опалення житлових приміщень// Комп'ютерні інтелектуальні системи та мережі. Матеріали XI Всеукраїнської науково практичної WEB конференції аспірантів, студентів та молодих вчених (21-23 березня 2019 р.).– Кривий Ріг: ДВНЗ «Криворізький національний університет», 2019. – С. 150-151.
9. Іщенко М.О. Коренко О.О. Автоматизована система енергоопалення житлових приміщень // Комп'ютерні інтелектуальні системи та мережі. Матеріали X Всеукраїнської науково практичної WEB конференції аспірантів, студентів та молодих вчених (22-24 березня 2017 р.).– Кривий Ріг: ДВНЗ «Криворізький національний університет», 2017. – С. 57-59.
- 10.Прімачова О.К., Іщенко М.О. Екологічна та енергоефективна система автоматизації міського розважального центру на базі технологічних рішень Phoenix Contact // Комп'ютерні інтелектуальні системи та мережі. Матеріали IX Всеукраїнської науково практичної WEB конференції аспірантів, студентів та молодих вчених (22-26 березня 2015 р.).– Кривий Ріг: ДВНЗ «Криворізький національний університет», 2015. – С. 107-109.

*Шевчук В. В.,
Криворізький національний університет
Ищенко М.О.,
к. т. н., доцент, Криворізький національний університет*

ВИДИ МОНІТОРИНГУ ДОВКІЛЛЯ, НАСЛІДКИ ЗАБРУДНЕННЯ АТМОСФЕРНОГО ПОВІТРЯ ТА РОЗРОБКА СИСТЕМИ МОНІТОРИНГУ ВМІСТУ ШКІДЛИВИХ РЕЧОВИН В ПОВІТРІ

Розглянуто види моніторингу довкілля, проаналізовано наслідки забруднення атмосферного повітря та розроблено власну систему моніторингу вмісту шкідливих речовин в повітрі.

Види моніторингу довкілля поділяються на:

1) Загальний (стандартний) моніторинг – це оптимальні за кількістю параметрів спостереження на пунктах, об'єднаних в інформаційно-технологічну мережу, які дають змогу на підставі оцінки і прогнозу стану довкілля регулярно розробляти управлінські рішення на всіх рівнях [1-3].

2) Оперативний (кризовий) моніторинг – це вивчення спеціальних показників на цільовій мережі пунктів у реальному масштабі часу за окремими об'єктами, джерелами підвищеного екологічного ризику в окремих регіонах, які визначено як зони надзвичайної ситуації, а також у районах аварій зі шкідливими екологічними наслідками для забезпечення оперативного реагування на кризові ситуації та прийняття рішень щодо їхньої ліквідації, створення безпечних умов для населення [1-3].

3) Фоновий (науковий) моніторинг – це спеціальні високоточні спостереження за всіма складовими довкілля, а також за характером, складом, кругообігом і міграцією забруднювальних речовин, за реакцією організмів на забруднення як на рівні окремих популяцій чи екосистем, так і біосфери в цілому. Його проводять на базових станціях у природних і біосферних заповідниках, а також на інших природоохоронних територіях [1-3].

У 2018 році Всесвітня метеорологічна організація (ВМО) опублікувала свою заяву щодо стану глобального клімату. Понад 2 мільйони людей по всьому світу стали вимушеними переселенцями

через екстремальну погоду: засухи, повені, шторми. Наслідки зміни клімату зачіпають не лише Африку, яка найбільше потерпає від посух, чи островні та прибережні країни, які найсильніше страждають від повеней та ураганів. Їх відчуває на собі і Європа. Згідно звіту:

- Травень-липень 2018 року стали найсухішими та найспекотнішими за всю історію спостережень у багатьох регіонах Скандинавії та Фінляндії.
- Квітень-вересень 2018 року були другими найсухішими у Німеччині, що призвело до значних сільськогосподарських втрат.
- Аномальна спека у Центральній Європі наприкінці липня – в серпні була менш потужною, ніж у деякі попередні роки, однак призвела до 1500 смертей.
- Лісові пожежі охопили 25 тисяч гектарів землі у Швеції, зачепили Латвію, Норвегію, Німеччину, Британію та Ірландію, вбили не менше 99 осіб у Греції.
- У 2018 році повені завдали шкоди 35 мільйонам людей [4-6].

Було розроблено власну систему моніторингу атмосферного повітря. Система моніторингу якості повітря складається з датчика вуглеводних та чадного газів, контролера та персонального комп'ютера. Датчик залежно від концентрації вуглеводних та чадного газів в повітрі змінює опір. Чим вища концентрація газів, тим більша напруга проходить через датчик. Даний сигнал надсилається до контролера. Контролер оброблює отриманий сигнал за допомогою аналого-цифрового перетворювача (АЦП) та відправляє його до персонального комп'ютера за допомогою Ethernet інтерфейсу. На персональному комп'ютері доступ до інформації здійснюється через веб-інтерфейс.

ВИСНОВОК

Проблема забруднення атмосферного повітря є надзвичайно важливою та актуальною. Забруднення атмосферного повітря є основним чинником збільшення захворюваності та смертності в світі. Моніторинг стану якості атмосферного повітря допомагає контролювати рівень забруднення повітря.

ЛІТЕРАТУРА

1. Постанова Кабінету Міністрів України «Про охорону навколишнього природного середовища» [Електронний ресурс] // №391 від 30 березня 1998 р. – Режим доступу: <https://zakon.rada.gov.ua/laws/show/>
2. Принципи класифікації систем моніторингу [Електронний ресурс] // Хелпикс.Орг – Режим доступу до ресурсу: <https://helpiks.org/>
3. Моніторинг довкілля : підручник / [Боголюбов В. М., Клименко М. О., Мокін В. Б. та ін.] ; під ред. В. М. Боголюбова. [2-е вид., перероб. і доп.]. — Вінниця : ВНТУ, 2010. — 232 с.
4. Забруднення повітря – одна із головних причин передчасних смертей в Європі [Електронний ресурс] // Освіта в інтересах сталого розвитку в Україні. – 2016. – Режим доступу до ресурсу: <http://ecoosvita.org.ua/storinka-knygy/zabrudnennya-povitrya>.
5. Негативні наслідки зміни клімату стають все помітнішими – ВМО [Електронний ресурс] // Екодія. – 2019. – Режим доступу до ресурсу: <https://ecoaction.org.ua/nehatyvni-naslidky-zminy-klimatu-staiut-pomitnishymy.html>.
6. Зміни клімату прискорилися, – Всесвітня метеорологічна організація [Електронний ресурс] // Західна інформаційна корпорація. – 2019. – Режим доступу до ресурсу: https://zik.ua/news/2019/09/23/zminy_klimatu_pryskorylysy__vsesvitnya_meteorologichna_organizatsiya_1652005.

СЕКЦІЯ 7. SECURITY. ЗАХИСТ ІНФОРМАЦІЇ В КОМП'ЮТЕРНИХ СИСТЕМАХ ТА МЕРЕЖАХ

*Меліхов Є.В.
Інженерний інститут ЗНУ
Міхайлуца О.М
к.т.н., доцент, Інженерний інститут ЗНУ*

ЗАХИЩЕНЕ ПРОГРАМУВАННЯ В ОСНОВІ БУДЬ-ЯКОГО ЗАСТОСУНКУ

Проаналізовано прогалини, які часто зустрічаються в безпеці програм та важливість застосування захищеного програмування для застосунків будь-якого призначення.

Зараз, на нескінченних просторах Інтернету, можна знайти велику кількість інформації про викрадення даних зловмисниками з приладів користувачів. Люди, підприємства і навіть держава зазнають величезних збитків. Все це відбувається через те, що багато розробників приділяють велику увагу інтерфейсу програми, зручності користування і т.і. Але найбільша проблема криється у безпеці. Нікому не потрібна надзвичайно красива та зручна програма, якщо ваші дані можуть з неї викрасти. Це теж саме, що залишити дірку в паркані, сподіваючись на те, що її ніхто не знайде. Для того, щоб дані користувача залишилися у безпеці та жодна людина не мала до них несанкціонованого доступу існує захищене програмування. Його використання дозволяє уникнути важких, а іноді і фатальних, наслідків роботи програми за рахунок застосування спеціальних прийомів раннього виявлення і нейтралізації помилок. Програмуванням з захистом від помилок дозволяє розробникам програмних застосунків не чекати зловмисника, який знайде прогалини в безпеці та скористається ними, а ліквідувати їх на етапі розробки програмного забезпечення самостійно.

Вже багато років спеціалісти зі всього світу намагаються з'ясувати як саме можна ліквідувати найнебезпечніші прогалини у

безпеці програм[1-4]. Але для їх усунення потрібно спочатку їх визначити. Найбільш поширеними прогалинами в безпеці програм, на поточний час, визнані:

- переповнення буферу. Відбувається або коли в програму намагаються ввести більше інформації, ніж вона може обробити, або у разі введення некоректного символу. Запобігти можна, якщо зробити перевірку вхідного рядка на кількість символів та на їх коректність;

- некоректна аутентифікація. Часто користувачі обирають дуже прості паролі навіть не здогадуючись, що у їх аккаунт можна увійти простим перебором паролів. Логіка проста: «Це ж скільки часу потрібно, щоб підібрати мій пароль». Зловмисники ніколи не роблять цього вручну. Вони використовують спеціальні програми, які просто перебирають можливі варіанти з великою швидкістю. Запобігти можна зробивши програмні обмеження на формат паролю: довжина, спеціальні символи, великі літери;

- перехоплення інформації. Якщо використовується слабкий механізм шифрування зловмисник може отримати доступ до інформації перехопивши її. Запобігти цьому можливо завдяки використуванню складних алгоритмів шифрування;

- обробка помилок та виключень. Під час виконання програми може виникнути помилка, дії для якої не передбачені. У такому разі зловмисник може перехопити керування програмою і далі вона буде працювати за його власним сценарієм. Запобігти цьому можна зупиненням виконання програми у разі виникнення помилки, якщо не прописано інших дій.

Саме через вищенаведені причини частіш за все і виникають витоки інформації до сторонніх осіб. Якщо ж розробляти програму згідно з потребами захищеного програмування, то ризики зламу будуть значно мінімізовані. Зробити продукт повністю захищеним на сто відсотків неможливо, адже який захист не розробила би одна людина інша завжди зможе її зламати. А відповідно – і отримати повний доступ до персонального пристрою користувача. Саме тому захищене програмування потрібно використовувати для будь якого застосунку. Не має значення, що саме знаходиться у розробці: онлайн-банкінг, поштова скринька або програма для відображення громадського транспорту на карті міста. Адже зловмисник може використати будь-яку вразливість. Навіть якщо не використовуються

жодні користувачькі дані, при наявності прогалині в безпеці програми можна отримати доступ до всього пристрою. А далі непроханий гість може робити все, що йому заманеться. Саме тому у застосунку потрібно шукати всі можливі недоліки та способи проникнення до неї. Бо якщо програма отримує славу «відкритих воріт для зловмисника» - це буде не найкраща реклама для розробника.

ВИСНОВКИ

Обрана нами стратегія, програмування із захистом від помилок, або явне додавання обробки помилок, безпосередньо залежить від області застосування розроблюваного програмного продукту. Її своєчасне використання по перше, істотно зменшить ймовірність отримання невірних результатів, по друге, дозволить при написанні будь-яких програмних застосунків приділяти посилену увагу саме безпеці. Адже втрата контролю над однією (хай і не дуже важливою) конкретною програмою може означати втрату контролю над всім пристроєм разом з особистою інформацією, яка на ньому знаходиться.

ЛІТЕРАТУРА

1. Essential guide to business continuity and disaster recovery plans. Authentication. SearchSecurity : веб -сайт. URL: <https://searchsecurity.techtarget.com/definition/authentication>.
2. How to handle errors and exceptions in large scale software projects. Raygun: веб-сайт. URL: <https://raygun.com/blog/errors-and-exceptions>.
3. 5 big security mistakes coders make. Javaworld: веб-сайт. URL: <https://www.javaworld.com/article/2456298/5-big-security-mistakes-coders-make.html>.
4. Мистецтво оборонного програмування. Hubr: веб-сайт. URL: <https://habr.com/ru/post/320088>.

Yarosh I.V.
senior lecturer in Donetsk National Technical University
Cherniak T.O.
assistant in Donetsk National Technical University

DEVELOPMENT OF INTRUSION DETECTION SYSTEM FOR WEB APPLICATIONS

The analysis of existing intrusion detection systems allowed to conclude that these systems have a number of disadvantages. The fact forms the need in development of a solution that will eliminate the pitfalls identified.

Web applications are an integral part of the Internet world network. Its network interaction is based on the client-server architecture through the HTTP Protocol. PHP is one of the most popular languages for web application development.

Based on the analysis of the most widely-spread web vulnerabilities, it has been found that the main cause of security breaches lays in insufficiency or absence of filters in data transmitted to the web application. Higher and better level of security for web applications is ensured by intrusion detection systems. [1]

The main aim of this work lies in analysis of typical vulnerabilities in web page scripts written in the PHP language, and development of software for intrusion detection systems.

This intrusion detection system for web applications is based on PHP language. The developed system is designed to protect web applications from real attacks or intrusions. The intrusion detection system is built-in program that works simultaneously with the web application. The input points are the GET and POST methods of the HTTP Protocol, as well as the HTTP Cookie.

It is worth noting that the security provided by the developed intrusion detection system is directed only to the web application with the intrusion detection system integrated into it. In other words, filtration and monitoring of incoming data, as well as registering events occurred, is performed only at the web application level. The system does not monitor events that occur directly in the web server, the PHP interpreter, or server operating system.

CONCLUSIONS

The developed intrusion detection system for web applications is a modern solution that is able to provide information security against attacks for web-based applications.

REFERENCES

1. Низамутдинов М.Ф. Тактика защиты и нападения на web-приложения. Учебное пособие [Текст] / М.Ф. Низамутдинов – ВHV-СПб, 2008. – 423 с.

*Юшкевич.С.В ,
Державний університет «Житомирська Політехніка»
к.п.н, доцент Вакалюк.Т.А,
Державний університет «Житомирська політехніка»*

ЗАХИСТ ІНФОРМАЦІЇ НА ФІЗИЧНИХ НОСІЯХ ТА В МЕРЕЖІ ІНТЕРНЕТ

Проблема захисту інформації стала актуальною ще тоді коли виникли соціальні зв'язки між людьми і виникла потреба в збереженні даних ,серед яких були конфіденційні – такі ,доступ до яких повинні були мати тільки конкретні особи чи групи осіб. В ті часи інформація зберігалась переважно в письмовому вигляді ,так що її захист залежав фізичних факторів ,які не завжди були ефективні. Часто лист паперу з важливою інформацією (наприклад планом наступальної операції) потрапляв в руки сторонніх осіб ,що приводило до негативних наслідків.

Прогрес привів до різких змін – семимильного стрибка в сфері обробки ,зберігання та передачі інформації. Відходять в минуле паперові листи і стоси документації. Все ,що раніше зберігалось в товстостінних сейфах під сімома замками ,тепер може вміститись на звичайній флешці котра менша за фалангу пальця чи на сервері до якого мають доступ тільки певні групи людей. Світ змінився ,але старі проблеми залишилися актуальними.

Перше на що хочеться звернути увагу – це те що інформацію як і раніше потрібно транспортувати. Файл не може просто з'яви-

тись на іншому пристрої не покинувши простір сховища. Для передачі інформації використовуються різні способи і кожен із них не є 100% безпечним[1]. Звичайно на даний час інформація майже миттєво переміщується між пристроями, що дозволяє уникнути низки вразливостей котрі виникали раніше. Але все ж змінюються не тільки способи передачі даних – змінилися і способи їх незаконного отримання.

Перехопити файл підчас відправлення не дуже складно, особливо якщо відстань між відправником і отримувачем значна, і зловмисник має доступ до каналу передачі даних. Тому перше на що потрібно звернути увагу – фізичний захист такого каналу і його пристроїв, якщо він звичайно можливий. Найбільшу увагу варт приділити саме кінцевим пристроям, доступ до яких повинні мати тільки довірені особи. Але тут можуть виникнути проблеми – якщо кількість таких пристроїв і осіб невелика, то реалізувати такий захист просто, однак коли ми маємо справу з великою кількістю осіб, наприклад – банківською системою, то за безпеку свого пристрою кожен клієнт повинен турбуватись особисто. В такому випадку є великий шанс, що кінцевий пристрій таки потрапить в руки зловмисників.

Саме тоді в дію вступають програмні системи захисту інформації – фаєрволи і антивіруси. Перші надають доступ до інформації тільки авторизованим користувачам по системі секретного набору знаків паролю. Другі же відповідають за сканування файлів котрі відправляються на сервер або поміщаються на пристрій і відсіювання тих котрі несуть загрозу для безпечного функціонування мережі. Але навіть цей захист не є надійним.

Основна проблема програмних систем захисту полягає в тому, що методи атак на інформаційні мережі змінюються настільки швидко, що методи протидії зазвичай не встигають реалізувати. Справа в тому, що чим складніша система тим більше в ній вразливостей, про існування котрих зазвичай не знають і її розробники. Як результат якщо раніше система захисту успішно справлялась з певними типами загроз, то з появою нових вразливостей її ефективність може значно упасти. Саме пошуком таких вразливостей «дірок в системі» і зайняті зловмисники.

Тоді і виникає питання «Як захистити інформацію?». Відповідь на це питання неоднозначна, так як єдиного механізму

котрий забезпечить усесторонній захист не існує. Як і не існує ідеальної багаторівневої системи захисту. Єдиний спосіб підняти рівень інформаційного захисту – моніторинг і швидка реакція на нові загрози ,а разом із тим вивчення і прогнозування нових атак. Важливу роль також відіграє тестування інформаційних систем «фальш-атаками» з метою виявлення не помічених на етапі розробки вразливостей для подальшого їх усунення. Також потрібно стимулювання поширення інформаційної грамотності серед населення ,так як кожен учасник мережі може стати жертвою чи інструментом в руках зловмисника.

Як відомо внутрішні загрози являються одними з найнебезпечніших і поширенню їх можна подякувати системі «Принеси свій пристрій на роботу». Звичайно ця система являється надзвичайно мобільною ,економною та прогресивною але безпека особистої інформації підприємства стає більш вразливою так як виникають додаткові канали передачі і кінцеві пристрої до яких не так вже і складно дістатись. І це вже не кажучи ,що норми захисту інформації можуть банально не дотримуватись – важливі документи можуть зберігатись в незапароленій папці чи будуть незашифровані. Звичайно такі пристрої це золото для зловмисника ,так як інформація може бути викрадена майже без зусиль.

ЛІТЕРАТУРА

1. С.Остапов «Технології захисту інформації» Остапов.С \2014 – 428с.

Джакелі Н. С.

Полтавська державна аграрна академія

Калініченко О. В.

к. е. н., доцент, Полтавська державна аграрна академія

ІНФОРМАЦІЙНА БЕЗПЕКА ЯК ОСНОВА НАЦІОНАЛЬНОЇ БЕЗПЕКИ

Розглянуто вплив інформаційної безпеки на національну безпеку України. Та визначено основні наслідки впливу незахищених даних на національну економіку.

Державна інформаційна політика є важливою складовою зовнішньої і внутрішньої політики країни та охоплює всі сфери життєдіяльності суспільства. Бурхливий розвиток інформаційної сфери супроводжується появою принципово нових загроз інтересам особистості, суспільства, держави та її національній безпеці.

Захищаючи свої інформаційні інтереси, кожна держава має дбати про свою інформаційну безпеку. Цього ж вимагає і зміцнення української державності. Збалансована державна інформаційна політика України формується як складова частина її соціально економічної політики, виходячи з пріоритетності національних інтересів та загроз національній безпеці країни [1, с. 320].

Порушення безпеки у державному і військовому управлінні, атомній енергетиці, медицині, ракетно-космічній галузі та у фінансовій сфері, може призвести до тяжких наслідків для навколишнього середовища, особливо для економіки і безпеки держави, здоров'я і навіть для життя людей. Економічні та юридичні питання, приватна та комерційна таємниця, національна безпека – усе це зумовлює необхідність захисту інформації та інформаційної системи.

Головна інформаційна загроза національній безпеці – це загроза впливу іншої сторони на інформаційну інфраструктуру країни, інформаційні ресурси, на суспільство, свідомість, підсвідомість особистості, з метою нав'язати державі бажану систему цінностей, поглядів, інтересів і рішень у життєво важливих сферах суспільної й державної діяльності, керувати їхньою поведінкою і розвитком у бажаному для іншої сторони напрямку. Власне, це є загрозою суверенітету України в життєво важливих сферах суспільної й державної діяльності, що реалізовується на інформаційному рівні. Стратегічне інформаційне протистояння є самостійним і принципово новим видом протистояння, здатним вирішувати конфлікт без застосування збройних сил у традиційному розумінні [2].

Для вивчення закономірностей інформаційного протистояння та аналізу його кількісних характеристик необхідно формалізувати як поняття рівня інформаційної озброєності держави, так і механізм еволюції ресурсного потенціалу конкретної держави та вплив зовнішнього оточення. В даному випадку за основу аналізу вибраний інформаційний стан України.

Забезпечення безпеки інформаційних технологій я – це комплексна проблема, яка охоплює правове регулювання використання

ІТ, удосконалення технологій їх розробки, розвиток системи сертифікації, забезпечення відповідних організаційно-технічних умов експлуатації. Розв'язання цієї проблеми потребує значних витрат, тому першочерговим завданням є співвіднесення рівня необхідної безпеки і витрат на її підтримку. Для цього необхідно визначити потенційні загрози, імовірність їх настання та можливі наслідки, вибрати адекватні засоби і побудувати надійну систему захисту.

В сфері обороноздатності країни рівень інформаційного потенціалу все в більшій мірі визначає оперативність прийняття рішень, структуру і якість озброєнь, оцінку рівня їх необхідної достатності, дієвість пропаганди, ефективність дій союзників і власних збройних сил і, в кінцевому підсумку, результат збройного протистояння [3, с 473]. Рівень інформаційного потенціалу в сфері соціальних стосунків разом з розвиненими інформаційними відносинами дає можливість досягнення соціального компромісу і тим самим – підтримання стабільності суспільства і держави.

ВИСНОВКИ

Отже, безпека та рівень розвитку інформаційного середовища, які є системоутворюючим фактором в усіх сферах національної безпеки, активно впливають на стан політичної, економічної, оборонної та інших складових національної безпеки України [4, с 184]. Забезпечено інформаційна безпека є важливою самостійною сферою національної безпеки, та захищеність життєво важливих інтересів суспільства. Тому стійкий, прогресивний розвиток України як суверенної, демократичної, правової і економічно стабільної держави можливий тільки за умови забезпечення інформаційної безпеки всіх суб'єктів інформаційних стосунків країни.

ЛІТЕРАТУРА

1. Боднар І. Р. Сучасні реалії інформаційного суспільства: проблеми становлення та перспективи розвитку: монографія [Текст] / І. Р. Боднар. – Львів: Видавництво Львівської комерційної академії, 2013. – 320 с.
2. Бондаренко В., Литвиненко О. Інформаційна безпека сучасної держави: концептуальні роздуми [Електронний ресурс] / В. Бондаренко, О. Литвиненко - Режим доступу: <http://www.crime-research.iatp.org.ua/library/strateg.htm>.
3. Зубок М. І., Яременко С. М. Безпека банківської діяльності: підруч. – К.: КНЕУ, 2012. – 473с.

4. Голев Д.В. Інформаційна безпека інформаційно – комунікаційних систем. Навч. Посібник / Г. Кононовича. / Д.В. Голев, О.Ю. Русляченко, Ю.В. Белова, Д.С. Гончарук – Одеса: ОНАЗ ім. О.С. Попова, 2014. – 184 с.

*Невмержицький В. В.,
Криворізький національний університет
Кузнєцов Д.І.,
к. т. н., доцент, Криворізький національний університет*

ОХОРОННА СИГНАЛІЗАЦІЯ ТА КРИПТОСТІЙКІ АЛГОРИТМИ ЇЇ ШИФРУВАННЯ

Досліджено системи охоронної сигналізації, діапазони частот їх роботи. Проаналізовано різні алгоритми та способи захисту які наявні на ринку сигналізацій.

Охоронна сигналізація призначена для захисту житла чи автомобіля від несанкціонованого проникнення, і в даний час набула широкого поширення. Це кращий спосіб забезпечити охорону вашого майна під час вашої відсутності.

Бездротова система сигналізації має такі якості, як портативність, легкість у встановленні й відсутності зайвих дротів. Всі сигнали для зв'язку з пристроями передаються по жорстко заданим частотам. Як правило, в кожній країні є свій безкоштовний спектр радіочастот, а є урядовий, розподіл якого регулюється постановою «Про затвердження Національної таблиці розподілу смуг радіочастот України», від 15 грудня 2005 р. № 1208[1].

У документі перераховано весь перелік існуючих радіочастот і вказані в яких діапазонах дозволено використовувати комерційним організаціям. Згідно з додатком частота 433МГц лежить в діапазоні: 433,05 - 434,79 МГц і відноситься до неспеціалізованих пристроїв радіочастотної ідентифікації, пристроїв охоронної радіосигналізації автомашин, а частота 868МГц (868 - 870 МГц) до неспеціалізованих пристроїв охоронної сигналізації. На базі цих частот створюють бездротові охоронні сигналізації.

До безкоштовних бездротових каналів зв'язку відноситься міжнародний діапазон ISM. Для його застосування не потрібно ліцензування. Bluetooth, Wi-Fi, IEEE 802.15.4, Zigbee працюють в цьому діапазоні.

Завдяки високій швидкості роботи радіоканалу, високої стійкості до помилок зв'язку та малому енергоспоживанню цей діапазон застосовується в більшості сучасних пристроїв. Частоти 433MHz і 868MHz — це дві головні групи, які зазвичай використовуються в системі бездротової сигналізації.

Дальність дії пристроїв, що працюють на частоті 433MHz і 868MHz невелика, і скорочується в залежності від наявності сторонніх об'єктів на шляху передачі сигналу. Частота 433MHz добре себе зарекомендувала для рухливих об'єктів. 868MHz має перевагу у швидкості передачі, обміну даних і дальності передачі сигналу.

Алгоритм шифрування в охоронних системах — набір правил обміну даними між пультом і блоком управління сигналізації, за якими здійснюється захист передачі даних, щоб зловмисник не зміг дістати несанкціонований доступ в автомобіль чи будівлю, навіть якщо зможе перехоплювати пакети обміну даними.

Алгоритми найперших сигналізацій ґрунтувалися на статичному кодуванні. При цьому була всього одна команда для однієї дії, ця команда була статичною та не змінювалась під час експлуатації. Наприклад, команді «закрити двері» завжди був код для надсилання «Z999X» (в такому форматі він передавався від пульта на блок управління). Тоді, коли варіантів коду сигналів було мало, то іноді своїм пультом можна було відчинити іншу машину з такою ж сигналізацією — формати кодів для управління збігалися. Таке кодування не могло забезпечувати належного захисту досить було один раз переловити код відповідної однієї із команд, а потім використовувати її для несанкціонованого доступу.

Зараз на ринку є стійкі до електронного злому системи, цим користуються виробники, які масово виробляють сигналізації. Є таке поняття — мануфактурний код. Фактично це статична база даних електронних посилок, якими обмінюються між собою пульт і сигналізація. Мануфактурні коди, якимсь чином, виявляються у розробників код-граберів і заносяться в пам'ять пристроїв. Після цього граберу досить перехопити одну з команд, щоб пристрій вирахував наступну команду і почав працювати, як штатний пульт. І, до речі,

саме сканування в цьому випадку відбувається непомітно для справжнього власника.

Не піддаються скануванню сигналізації з так званим діалоговим кодом, але, на жаль, часом його вписують в характеристики безпідставно, так би мовити, в маркетингових цілях. Тому, вибираючи сигналізацію, можна побачити в характеристиці Dialog Code а насправді такий код там не використовується. Сигналізаціями які зараз є стійкими до код-граберів являються: Pandora, Magnum, Magic Systems, Starline, Prizrak[2].

Діалоговий код — це не послідовність заздалегідь прописаних сигналів, а індивідуальний алгоритм обробки посилок, який робить перехоплення окремих кодів безглуздим.

Коли власник натискає на кнопку, з пульта на центральний блок сигналізації приходять запит на виконання команди. Далі блоку управління потрібно впевнитися в тому, що команда відправлена саме з пульта власника. Для цієї мети він генерує випадкове число і відправляє його на пульт. Це число за певним алгоритмом обробляється і передається назад на блок управління. В цей же час блок управління обробляє то саме число і порівнює його з числом надійшовшим від пульта. В тому і тільки тому випадку, якщо числа збігаються, центральний блок сигналізації виконує команду. Варто зазначити, що алгоритм, за яким виконуються розрахунки над випадковим числом, суто індивідуальний для кожної сигналізації, закладається в неї ще на етапі виробництва і в більшості випадків є комерційною таємницею.

ВИСНОВКИ

На сьогодні є досить багато охоронних сигналізацій як для автомобіля так і для будівель. Сучасні методи криптистійкого захисту можуть забезпечити максимальний захист, але зловмисники теж знаходять методи їх злому, тому всі системи охоронної сигналізації потребують постійного вдосконалення та розробки нових алгоритмів шифрування сигналу.

ЛІТЕРАТУРА

1. Про затвердження Національної таблиці розподілу смуг радіочастот України [Електронний ресурс].—Режим доступу <https://zakon.rada.gov.ua/laws/show/1208-2005-p>

2. Алгоритмический кодграббер и диалоговый код. [Електронний ресурс].–Режим доступу <https://www.drive2.ru/b/1619320/>

*Дзензура А. А.,
Криворізький національний університет
Кумченко Ю. О.
к.т.н., доцент, Криворізький національний університет*

ЗАХИЩЕНЕ ЛОКАЛЬНЕ МЕРЕЖЕВЕ СЕРЕДОВИЩЕ ДЛЯ ЗБЕРІГАННЯ ФАЙЛІВ НА ОСНОВІ NAS

Розглянуто основні види загроз та засоби захисту локального мережевого середовища на основі NAS. Запропоновано схему побудови захисту сервера та зображено типові методи атак.

Тема захисту інформації, загалом конфіденційних даних, є дуже важливою та актуальною у наш час. Власники інформаційних систем, які не надають належної уваги до захисту, ризикують витоком, знищенням та спотворенням важливої інформації.

Мережеве сховище знаходиться в стані захищеності, якщо забезпечена доступність, конфіденційність та цілісність інформації. Дані, які знаходяться в локальних (LAN), або в глобальній мережах (WAN), постійно піддаються різним видам загроз. Найбільш поширеними видами атак, на локальне мережеве середовище для зберігання файлів на основі Network Attached Storage (NAS), є наступні:

- 1) прослуховування мережі (Sniffing) – крадіжка або перехоплення даних шляхом захоплення мережевого трафіку за допомогою спеціалізованого ПЗ – Sniffer;
- 2) атака-вторгнення (Invasion Attack) – перехоплення управління ресурсами мережі;
- 3) відмова в обслуговуванні (Denial of Service) – атака на обчислювальну систему з метою довести її до відмови;
- 4) ping-флуд (Ping Flooding) – приведення сервера до відмови шляхом великої кількості запитів;

- 5) IP-спуфінг (IP Spoofing) – використання чужої IP-адреси з метою обману системи безпеки;
- 6) шкідливі програми (viruses, worms, trojan horses) та ін.

Є три фундаментальні групи методів захисту мережевого середовища.

Група програмно-технічних засобів до яких входять:

- 1) аналізатори протоколів;
- 2) антивірусні програми;
- 3) міжмережні екрани (брандмауери);
- 4) криптографічні засоби;
- 5) системи резервного копіювання.

Група захисту технічних каналів зв'язку до яких входять:

- 1) використання екранового кабелю;
- 2) встановлення високочастотних фільтрів;
- 3) використання екранового обладнання.

Група захисту периметра інформаційного середовища до якого входять:

- 1) системи охоронної та пожежної сигналізації;
- 2) системи відео спостереження;
- 3) системи контролю доступу.

Запропонована схема, яка представлена на рисунку 1, демонструє захист локального мережевого середовища для зберігання файлів на основі NAS від типових методів атак.

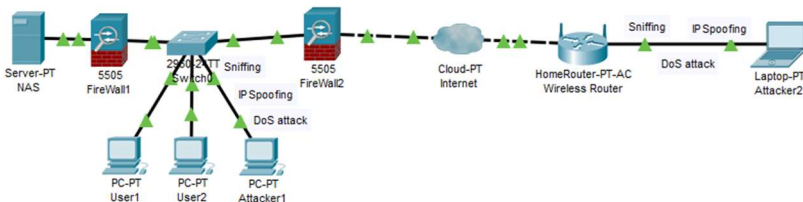


Рис.1 – Схема захисту та нападу на сервер

ВИСНОВКИ

Таким чином захист локального середовища є дуже важливим через велику кількість загроз. Не існує ідеального захисту, тому для

захисту локального мережевого сховища потрібно використовувати комплексні методи захисту.

*Мисливець Д. О.,
Криворізький національний університет
Кумченко Ю. О.
к.т.н., доцент, Криворізький національний університет*

СИСТЕМА ЗАХИСТУ КОМП'ЮТЕРІВ МЕТОДОМ ФІЛЬТРАЦІЇ МЕРЕЖЕВОГО ТРАФІКУ

Розглянуто актуальність використання брандмауеру для користувачів комп'ютерів. Представлено розмежування контент-фільтрів та класифікацію фільтрації.

У зв'язку з розвитком різносторонніх оновлених мережевих технологій збільшується обсяг інформації, яка передається мережею, а також появою нових протоколів передачі даних прикладного рівня, тому все більшої актуальності в наш час набуває метод фільтрації цього трафіку.

Файрвол, брандмауер чи мережевий екран – це пристрій забезпечення мережевої безпеки, що здійснює моніторинг вхідного та вихідного трафіку, на базі певних, встановлених правил безпеки, приймаючи рішення про дозвіл чи заборону трафіку в мережі [1]. Реалізація відбувається шляхом використання певного програмного, апаратного, або програмно-апаратного забезпечення. Застосовується як спосіб, щоб забезпечити захист комп'ютера від різноманітних мережевих атак, таких як: шпигунські програми, DDoS-атаки тощо; блокування відвідування заражених вірусами або небажаних інтернет-сайтів; виявлення різноманітних шпигунських засобів стеження за активністю користувача.

ПАРАМЕТРИ КОНТЕНТ-ФІЛЬТРІВ

Системи фільтрації мережевого трафіку можна розділити на такі показники:

1. Підзвітність. Оцінює кількість участі населення в політиці фільтрації контенту.

2. Відкритість. Дає можливість користувачу отримати перевірену інформацію про відвідування ресурсу, який віднесений до заборонених.
3. Точність. Відповідає за успішність цензури (тобто надмірне або ж недостатнє блокування)
4. Прозорість. Надання параметрів, котрі дозволяють віднести трафік до забороненого.

КЛАСИФІКАЦІЯ ФІЛЬТРАЦІЇ ТРАФІКУ

1. Міжнародний рівень. Фільтрація DNS-запитів на державному рівні.
2. Рівень інтернет-шлюзу. Потребує інсталяції програмного забезпечення (ПЗ), що забезпечить фільтрацію. Метод зберігає швидкість інтернет-доступу. Використовується приватними підприємствами та державними організаціями.
3. Рівень інтернет-провайдерів. Для даної фільтрації використовуються переліки заборонених сайтів, які сформувались судами та державними службами. Метод визнаний надійним.
4. Рівень комп'ютера користувача. ПЗ інсталується на комп'ютер. Метод вважається ефективним для домашнього використання, а також для застосування на невеликих підприємствах. Є доступним рішенням.

ВИСНОВКИ

Таким чином, фільтрація мережевого трафіку є необхідною для кожного користувача комп'ютера. На сьогодні один із методів захисту від комп'ютерних зловмисників є саме міжмережевий екран. І хоча він не гарантує повноцінний захист від професійних хакерів, але все ж таки ускладнює їм отримати доступ до конфіденційної інформації.

ЛІТЕРАТУРА

1. Что такое межсетевой экран? – URL: https://www.cisco.com/c/ru_ru/products/security/firewalls/what-is-a-firewall.html (дата звернення: 20.02.2019).

*Ковбасюк Ю.Б.
Криворізький національний університет
Сенько А.О.
асистент, Криворізький національний університет*

НАВЧАЛЬНІ КУРСИ НА ПЛАТФОРМІ GOOGLE CLASSROOM

Розглянуто актуальність веб-сервісів і хмарних платформ у організації та модернізації навчальних процесів. Проведено короткий огляд платформи Google Classroom.

На сьогоднішній день все більше освітніх закладів для покращення організації навчальних процесів використовують сучасні інструменти. Сучасні технології дозволяють проводити дистанційне навчання, поширювати та обмінюватися знаннями в електронній формі в будь-який час і в будь-якому куточку світу. Тому розвиток платформ, що зможуть допомогти зробити освітні процеси ще більш модернізованими та ефективними є доволі актуальним. Однією з найвідоміших і найбільших платформ для оптимізації навчання є Google Classroom.

Google Classroom - веб-сервіс від компанії Google, що розроблений для покращення процесу керування освітніми та навчальними процесами, полегшення поширення навчальних матеріалів в електронному вигляді. В першу чергу спрямований на використання у закладах вищої освіти та школах.

Google Classroom було випущено у серпні 2014 року. Спочатку сервіс був доступним для користування лише через браузер, але у січні 2015 було випущено мобільні додатки для пристроїв на базі iOS та Android з підтримкою оффлайн доступу. Веб-сервіс підтримує різні мови, в тому числі українську. На сьогоднішній день Google Classroom знаходиться в активному стані розробки, останнє оновлення було випущено у серпні 2018. Веб-сервіс є безкоштовним.[3]

Google Classroom складається з наступних сервісів та інструментів від Google: Google Calendar – сервіс для планування занять і розкладу; Google Drive – сховище для зберігання даних, що дозволяє користувачам ділитися даними з іншими; Google Docs, Sheets

and Slides – веб-пакет для роботи з документами; Gmail – сервіс електронної пошти.

Веб-сервіс Google Classroom дозволяє створити окремий клас для учнів, виділяючи місце на Google Drive для розміщення навчальних матеріалів, значно полегшує взаємодію між викладачами і учнями, дозволяючи відмовитись від паперових носіїв передачі інформації. Сервіс має просте налаштування та інтеграцію з G Suite for Education, що дозволяє більш ефективно організувати роботу, впорядковувати файли, завдання, оцінки, тощо. Також сервіс підтримує функцію архівації курсів після закінчення навчального року або семестру. Окрім безпосереднього спілкування з окремим учнем, викладачі можуть розміщувати об'яви з інформацією для потоку класів. Це дозволяє полегшити і пришвидшити розповсюдження інформації.[1]

Сервіс підтримує функцію перевірки робіт учнів на оригінальність. Звіти про оригінальність можуть допомогти покращити якість виконання завдань, забезпечуючи швидку перевірку на академічну доброчесність. Використовуючи пошук, сервіс порівнює роботи учнів з веб-сторінками та книгами, висвітлюючи у разі виявлення плагіат або велику схожість із зовнішнім джерелом.[2]

ВИСНОВКИ

Виконавши огляд хмарної платформи для навчання Google Classroom, можна зробити висновок, що дана платформа є однією з найбільших і найвідоміших платформ подібного типу, що забезпечує весь необхідний функціонал для модернізації та полегшення процесу організації навчального процесу.

ЛІТЕРАТУРА

1. Manage teaching and learning with Classroom [Електронний ресурс] – Режим доступу до ресурсу: https://edu.google.com/products/classroom/?modal_active=none.
2. Help students keep their ideas authentic [Електронний ресурс] – Режим доступу до ресурсу: https://edu.google.com/products/originality/?modal_active=none.
3. Google Classroom [Електронний ресурс] – Режим доступу до ресурсу: https://uk.wikipedia.org/wiki/Google_Classroom.

Маркова О. М.
к. пед.н., ст. викладач Криворізький національний університет

ДЕМОНСТРАЦІЯ ОСНОВНИХ МЕТОДІВ КОДУВАННЯ ЗАСОБАМИ WOLFRAM DEMONSTRATION PROJECT

Автором наведені можливості одного із хмаро зорієнтованих дидактичних демонстраційних матеріалів – Wolfram Demonstration Project для демонстрації роботи коду Хемінга у процесі вивчення змістового модуля «Теорія кодування» спецкурсу з основ математичної інформатики.

У роботі [1] показано, що Wolfram Demonstration Project (проект компанії Wolfram Research) можливо ефективно застосовувати як хмаро зорієнтований дидактичний демонстраційний матеріал з основ математичної інформатики, оскільки бібліотека проекту налічує значну кількість демонстрацій, зокрема, зі змістового модуля «Теорії кодування» (рис. 1).

Для використання можливостей даного проекту достатньо встановити вільно поширюваний програвач Wolfram CDF Player, за допомогою якого можна переглядати CDF-файли (обчислювальний формат документа) у поширених веб-браузерах.

Продемонструємо можливості проекту при вивченні теми «Лінійні коди» змістового модуля «Теорія кодування», що входить до спецкурсу з основ математичної інформатики. Демонстрація роботи коду Хемінга доступна за посиланням <http://demonstrations.wolfram.com/TheHamming74Code/> (рис. 1).

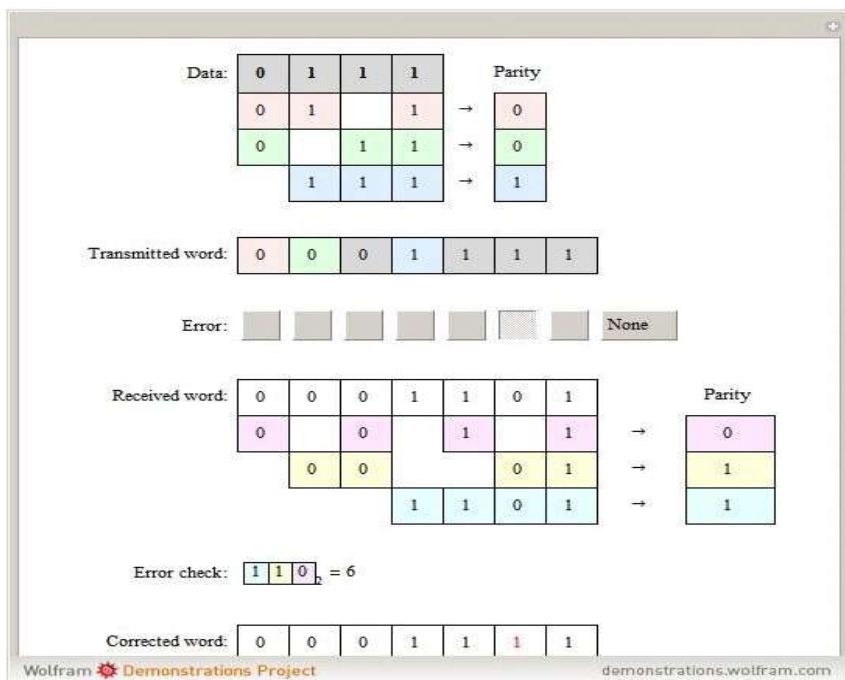


Рис. 1. Демонстрація роботи коду Хемінга

Код Хемінга (7, 4) є кодом, що може виправити одну помилку, використовуючи 7-бітне кодове слово для передавання чотирьох бітів даних. Відправник обчислює три біти парності для кожного 4-бітного слова даних, збирає дані та біти парності у 7-бітне кодове слово та передає це кодове слово. Отримувач обчислює три біти перевірки парності з прийнятого 7-бітного слова. Якщо помилки при передаванні не було, всі три контрольні біти парності дорівнюватимуть нулеві. Якщо один біт при передаванні був змінений, значення трьох бітів парності (інтерпретується як 3-бітне двійкове число) буде вказувати на позицію помилки, яка потім може бути усунута.

ЛІТЕРАТУРА

1. Маркова О. М. Хмарні технології як засіб навчання основ математичної інформатики студентів технічних університетів : дис. ... канд. пед. наук : 13.00.10 – інформаційно-комунікаційні тех-

нології в освіті / Маркова Оксана Миколаївна ; Державний заклад „Луганський національний університет імені Тараса Шевченка”. – Старобільськ, 2019. – с. 141 – 142.

ЗМІСТ

СЕКЦІЯ 1. DIAGNOSTICS. ДІАГНОСТИКА КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ	3
ОПТИМІЗАЦІЯ КОРИСТУВАЦЬКОГО ДОСВІДУ З ВИКОРИСТАННЯМ СУЧАСНИХ ФРЕЙМВОРКІВ.	3
ДОСЛІДЖЕННЯ СИСТЕМИ ВІДДАЛЕНОГО КЕРУВАННЯ МОДУЛЯМИ З ЗВОРОТНИМ ЗВ'ЯЗКОМ ЧЕРЕЗ ІНТЕРНЕТ.....	6
СИСТЕМА МОНІТОРИНГУ ТА КОНТРОЛЮ СТАНУ ПАРАМЕТРІВ ВІДДАЛЕНОГО ВУЗЛА ЗВ'ЯЗКУ	8
СЕКЦІЯ 2. PARALLEL COMPUTING. ВИСОКОПРОДУКТИВНІ КОМП'ЮТЕРНІ СИСТЕМИ, ПАРАЛЕЛЬНІ ТА РОЗПОДІЛЕНІ ОБЧИСЛЕННЯ	12
ПОРІВНЯННЯ ПРОДУКТИВНОСТІ ПАРАЛЕЛЬНИХ ОБЧИСЛЕНЬ ПРИ ВИКОРИСТАННІ РІЗНИХ ПЛАНУВАЛЬНИКІВ OPENMP	12
ПОРІВНЯЛЬНИЙ АНАЛІЗ ЗАСОБІВ ПРОФІЛЮВАННЯ ПРОДУКТИВНОСТІ ПРИ РОБОТІ З OPENMP ДОДАТКАМИ... ..	15
ВИКОРИСТАННЯ ДНК ЯК ЗАСІБ ЗБЕРЕЖЕННЯ ІНФОРМАЦІЇ.....	18
ПЕРСПЕКТИВИ РАЗВИТИЯ МИКРОПРОЦЕССОРНЫХ УСТРОЙСТВ	20
ВИБІР КРАЩОГО ТА ПРОСТОГО В ОСВОЄННІ ДВИЖКА ДЛЯ СТВОРЕННЯ ІНДІ-ІГОР	23
СЕКЦІЯ 3. DESIGN. ПРОЕКТУВАННЯ КОМП'ЮТЕРНИХ СИСТЕМ ТА МЕРЕЖ	28
ВИКОРИСТАННЯ ХМАРНИХ ТЕХНОЛОГІЙ В ОРГАНІЗАЦІЇ НАВЧАЛЬНОГО ПРОЦЕСУ	28
ВИКОРИСТАННЯ СУЧАСНИХ ТЕХНОЛОГІЙ БЕЗДРОТОВИХ МЕРЕЖ У ЗАКЛАДАХ ВИЩОЇ ОСВІТИ	30
РАЗРАБОТКА СИСТЕМЫ АВТОМАТИЗАЦИИ ТЕСТИРОВАНИЯ БОЛЬШОГО ОБЪЕМА ПРОЕКТОВ	32

РОЗГОРТАННЯ ШПД КОМП'ЮТЕРНОЇ МЕРЕЖІ ЗА ТЕХНОЛОГІЄЮ GPRN.....	35
АВТЕНТИФІКАЦІЯ КОРИСТУВАЧІВ НА WEB-ДОДАТКАХ... 37	37
ПРОГНОЗУВАННЯ НАВАНТАЖЕННЯ КОМП'ЮТЕРНИХ МЕРЕЖ НА ОСНОВІ МОДЕЛІ ARMA	40
ФАКТОРЫ ВЛИЯЮЩИЕ НА РАНЖИРОВАНИЕ WEB-САЙТА. 41	41
АНАЛИЗ ПСИХОЛОГИИ И ЮЗАБИЛИТИ WEB-ДИЗАЙНА	44
ОБґРУНТУВАННЯ ВИБОРУ ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ ТА ВИМОГ ДО АПАРАТНОГО ЗАБЕЗПЕЧЕННЯ	47
ІНФОРМАЦІЙНА СИСТЕМА Е-СТУДЕНТ	50
СЕКЦІЯ 4. PROGRAMMING. СИСТЕМНЕ ТА ПРИКЛАДНЕ ПРОГРАМУВАННЯ В КОМП'ЮТЕРНИХ СИСТЕМАХ ТА МЕРЕЖАХ	52
АНАЛІЗ ПОПУЛЯРНИХ НА ІТ-РИНКУ ПРАЦІ МОВ ПРОГРАМУВАННЯ	52
БЛОКЧЕЙН-ТЕХНОЛОГІЯ: ХАРАКТЕРИСТИКА ТА СФЕРИ ЗАСТОСУВАННЯ.....	55
ЗАГРОЗИ І РИЗИКИ ВПРОВАДЖЕННЯ ТЕХНОЛОГІЇ БЛОКЧЕЙН.....	60
МОВА ПРОГРАМУВАННЯ PYTHON - ПОТУЖНИЙ ІНСТРУМЕНТ ДЛЯ РОЗРОБКИ ДОДАТКІВ ЗІ ШТУЧНИМ ІНТЕЛЕКТОМ	64
ОСОБЛИВОСТІ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ WEB-СЕРВІСУ АВТОМАТИЗАЦІЇ СТО	66
АКТУАЛЬНІСТЬ РОЗРОБКИ СИСТЕМИ РОЗПІЗНАВАННЯ ВІЛЬНИХ МІСЦЬ ДЛЯ ПАРКУВАННЯ АВТО	70
СИСТЕМИ ЕЛЕКТРОННИХ ТИРІВ: ПЕРЕВАГИ ТА НЕДОЛІКИ.....	71
РАЗРАБОТКА PHP-ФРЕЙМВОРКА ДЛЯ СОЗДАНИЯ WEB- ПРИЛОЖЕНИЙ	74

ПОРІВНЯЛЬНИЙ АНАЛІЗ СУЧАСНИХ JAVASCRIPT ФРЕЙМВОРКІВ	77
РОЗРОБКА ТА ВПРОВАДЖЕННЯ ПРОГРАМНОГО ЗАСОБУ ДЛЯ КОНТРОЛЮ ДОСЯГНЕНЬ УЧНІВ ЗАГАЛЬНООСВІТНЬОЇ ШКОЛИ.....	79
СПОСОБИ ПОВЫШЕНИЯ СКОРОСТИ ОБРАБОТКИ ЗАПРОСОВ БАЗ ДАННЫХ	82
ИНТЕРАКТИВНАЯ ПОЗНАВАТЕЛЬНАЯ ИГРА НА ПЛАТФОРМЕ UNITY 2D.....	84
ВИКОРИСТАННЯ МОВ ПРОГРАМУВАННЯ GOLANG ТА JAVA В ЯКОСТІ СЕРВЕРНИХ МОВ ДЛЯ РОЗРОБКИ WEB- ЗАСТОСУНКІВ	86
МЕТОДИ ТА ЗАСОБИ ПОШУКОВОЇ ОПТИМІЗАЦІЇ САЙТІВ НАВЧАЛЬНИХ УСТАНОВ	90
ОРГАНАЙЗЕР РОБОТИ СТУДЕНТА НА БАЗІ DISCORD API....	92
ІНФОРМАЦІЙНА СИСТЕМА ДЛЯ ОРГАНІЗАЦІЇ КОРПОРАТИВНОЇ РОБОТИ НА ОСНОВІ TELEGRAM API.....	94
ВИКОРИСТАННЯ ХМАРНИХ ОБЧИСЛЮВАЛЬНИХ СИСТЕМ ДЛЯ РОЗРОБКИ WEB-ЗАСТОСУНКІВ	96
РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ СОЦІАЛЬНОЇ МЕРЕЖІ З ВИКОРИСТАННЯМ NODE.JS І REACT NATIVE	98
РОЗРОБКА МЕСЕНДЖЕРУ З ВИКОРИСТАННЯМ СУЧАСНИХ ВЕБ-ТЕХНОЛОГІЙ.....	101
РОЗВАЖАЛЬНА ОНЛАЙН ПЛАТФОРМА З ЗАСТОСУВАННЯМ ТЕХНОЛОГІЇ NODE.JS	104
АВТОМАТИЗАЦІЯ РОЗГОРТАННЯ WEB-ДОДАТКІВ З ВИКОРИСТАННЯМ КОНТЕЙНЕРІВ DOCKER.....	106
ВИКОРИСТАННЯ NIBERNATE ДЛЯ РОЗРОБКИ ПРОГРАМНИХ ДОДАТКІВ.....	109
АРХІТЕКТУРА ТА ОПИС ТЕХНОЛОГІЇ SPRING FRAMEWORK	112

ОСНОВНІ КОНЦЕПЦІЇ РЕАКТИВНОГО ПРОГРАМУВАННЯ	115
СИСТЕМА ВЕБ-МОНІТОРИНГУ ВИПУСКНИКІВ УНІВЕРСИТЕТУ У СОЦІАЛЬНИХ МЕРЕЖАХ	118
ОПТИМІЗАЦІЯ ДИНАМІЧНОГО КОНТЕНТУ З ВИКОРИСТАННЯМ СУЧАСНИХ ВЕБ-ТЕХНОЛОГІЙ	119
ОБҐРУНТУВАННЯ ВИБОРУ ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ РОЗРОБКИ ФІТНЕС-ДОДАТКУ	122
ОБҐРУНТУВАННЯ ВИБОРУ ІНСТРУМЕНТАЛЬНИХ ЗАСОБІВ ДЛЯ РОЗРОБКИ СИСТЕМИ ОБЛІКУ ВІДВІДУВАННЯ ЗАНЯТЬ СТУДЕНТАМИ	125
FAST OBJECT DETECTION WITH ROTATION AND SCALE INVARIANCE VIA SYMMETRIC POSITIONS OF A PIXEL: A CONCEPT FOR IMAGE PROCESSING	127
СЕКЦІЯ 5. ARTIFICIAL INTELLIGENCE. КОМП'ЮТЕРНІ СИСТЕМИ ТА МЕРЕЖІ ШТУЧНОГО ІНТЕЛЕКТУ	131
ДОСЛІДЖЕННЯ НЕЛІНІЙНИХ АЛГОРИТМІВ НАВЧАННЯ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ ПРЯМОГО ПОШИРЕННЯ	131
АЛГОРИТМИ НАВЧАННЯ ШТУЧНИХ НЕЙРОННИХ МЕРЕЖ ПРЯМОГО ПОШИРЕННЯ, ЩО ВИКОРИСТОВУЮТЬ НЕЛІНІЙНІ ПЕРЕТВОРЕННЯ МАТРИЦІ СПОСТЕРЕЖЕНЬ ...	133
ДО ВИБОРУ МАТЕМАТИЧНОЇ МОДЕЛІ В ЗАДАЧІ ПРОГНОЗУВАННЯ ЧАСОВИХ РЯДІВ	136
ДОСЛІДЖЕННЯ МЕТОДІВ РОЗПІЗНАВАННЯ ОБЛИЧЧЯ ЛЮДИНИ ЗА ДОПОМОГОЮ НЕЙРОННИХ МЕРЕЖ	138
ДОСЛІДЖЕННЯ АСПЕКТІВ СТРАТЕГІЙ СЕМПЛІНГУ	140
ЗАГРОЗИ ЯКІ ВИНИКАЮТЬ ПРИ ВПРОВАДЖЕННІ ШТУЧНОГО ІНТЕЛЕКТУ	142
ВИКОРИСТАННЯ ПАРАДИГМИ ЙМОВІРНІСНОГО ПРОГРАМУВАННЯ В МАШИННОМУ НАВЧАННІ	145
МОДЕЛЬ ВИЯВЛЕННЯ ШАХРАЙСТВА З КРЕДИТНИМИ КАРТКАМИ	147

ПЕРСПЕКТИВИ ВИКОРИСТАННЯ ШТУЧНОГО ІНТЕЛЕКТУ	150
ШТУЧНІ НЕЙРОННІ МЕРЕЖІ СЬОГОДНІ	153
SOFTWARE AND HARDWARE OF COMPUTER INTELLECTUAL CONTROL FOR MOBILE ROBOT MANIPULATOR.....	156
МОДЕЛИРОВАНИЕ ВЗАИМОДЕЙСТВИЯ ИНТЕЛЛЕКТУАЛЬНЫХ АГЕНТОВ В РАСПРЕДЕЛЕННЫХ МУЛЬТИАГЕНТНЫХ СИСТЕМАХ.....	159
ИСПОЛЬЗОВАНИЕ АНСАМБЛЕЙ ДЛЯ РЕДУКЦИИ ОБУЧАЮЩИХ ВЫБОРОК В РАСПОЗНАЮЩИХ МОДЕЛЯХ ПО ПРЕЦЕДЕНТАМ	164
ВИКОРИСТАННЯ НЕЙРОННИХ МЕРЕЖ У КЛАСИФІКАЦІЇ ОБ'ЄКТІВ ЗОБРАЖЕНЬ	167
ВИРШЕННЯ ПРОБЛЕМИ ДИСБАЛАНСУ КЛАСІВ В ОБРОБЦІ ДАНИХ.....	170
ПРОГРАМНО-АПАРАТНИЙ КОМПЛЕКС АВТОМАТИЗОВАНОГО ПОЛИВУ НА ОСНОВІ ВИКОРИСТАННЯ КОНЦЕПЦІЇ “ІОТ”	172
НЕЙРОМОРФНІ ПРОЦЕСОРИ І СИСТЕМИ	175
ПРОЕКТУВАННЯ КОМП'ЮТЕРНИХ СИСТЕМ НА ОСНОВІ АРХІТЕКТУРИ ANFIS	178
КОМП'ЮТЕРНА СИСТЕМА НА ОСНОВІ АРХІТЕКТУРИ ЗІ ШТУЧНИМ НЕЙРОПРОЦЕСОРОМ	180
ВИКОРИСТАННЯ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ АНАЛІЗУ ДОРОЖНО-ТРАНСПОРТНИХ ПРИГОД.....	182
РОЗПІЗНАВАННЯ НОМЕРІВ АВТОМОБІЛІВ НА ОСНОВІ ЕЛЕМЕНТІВ НЕЙРОННИХ МЕРЕЖ	185
ЗАСТОСУВАННЯ НЕЙРОННОЇ МЕРЕЖІ ДЛЯ РОСПІЗНАВАННЯ КУРИЛЬНИКІВ У ГРОМАДСЬКИХ МІСЦЯХ.....	188
RESEARCH AND DEVELOPMENT OF ACTUAL METHODS, TECHNOLOGIES AND METHODS APPLIED TO SOLUTION OF	

APPLIED MACHINE TRAINING PROBLEMS FOR PREDICTIVE PROTECTION	189
АНАЛІЗ ШТУЧНОГО ІГРОВОГО ІНТЕЛЕКТУ	191
СЕКЦІЯ 6. AUTOMATION, INDUSTRY 4.0. ПРОМИСЛОВІ МЕРЕЖІ, КІБЕРФІЗИЧНІ СИСТЕМИ, ВЕЛИКІ ДАНІ, ІНТЕРНЕТ РЕЧЕЙ, МОБІЛЬНІ ТА ХМАРНІ СЕРВІСИ, ЗАСОБИ ДОПОВНЕНОЇ РЕАЛЬНОСТІ.....	194
ВИКОРИСТАННЯ ХМАРНИХ ТЕХНОЛОГІЙ В БІЗНЕСІ	194
РОЗРОБКА ВЕБ-ДОДАТКУ ДЛЯ КЕРУВАННЯ ЕЛЕКТРОННИМИ ПРИСТРОЯМИ НА ПРИЛАДОБУДІВНОМУ ВИРОБНИЦТВІ У РАМКАХ РЕАЛІЗАЦІЇ КОНЦЕПЦІЇ ІНДУСТРІЇ 4.0	196
ІНФОРМАЦІЙНО КЕРУЮЧА СИСТЕМА МАНІПУЛЯТОРОМ ДЛЯ РОБОТИ В НЕБЕЗПЕЧНИХ УМОВАХ.....	199
РОЗРОБКА ХМАРООРІЄНТОВАНОГО КУРСУ ШКІЛЬНОЇ ІНФОРМАТИКИ	202
АВТОМАТИЗОВАНА СИСТЕМА КЕРУВАННЯ РОБОТОМ-ПІЛОСОСОМ	204
МЕТОДИ ТА ЗАСОБИ ВИЗНАЧЕННЯ ПРОСТОРОВОЇ ПОЗИЦІЇ РУК НА ОСНОВІ ВИКОРИСТАННЯ СУЧАСНИХ ІНТЕЛЕКТУАЛЬНИХ СИСТЕМ.....	207
АНАЛІЗ ХМАРНИХ ТЕХНОЛОГІЙ ЯК СЕРВІСІВ	210
РОЗРОБКА МУЛЬТИМЕДІЙНОГО РОЗШИРЕННЯ ДЛЯ ВЕБ-БРАУЗЕРА НА ДВИЖКУ ХРОМІУМ	212
КІБЕРФІЗИЧНІ СИСТЕМИ У СУЧАСНОМУ СВІТІ.....	214
СИСТЕМА БЕЗПЕКИ ПРОТОКОЛІВ ОБМІНУ ДАНИМИ У МЕРЕЖАХ ІОТ	216
ВИКОРИСТАННЯ БЕЗСЕРВЕРНИХ ОБЧИСЛЕНЬ У WEB-ДОДАТКАХ.....	219
ВЗАЄМОДІЯ ГЕТЕРОГЕННИХ ПРИСТРОЇВ ІНТЕРНЕТУ РЕЧЕЙ	222

СТРУКТУРА СИСТЕМИ КЕРУВАННЯ В MICROGRID	224
АНАЛІЗ ХАРАКТЕРИСТИК ТЕХНОЛОГІЧЕСКОГО ПРОЦЕСА ИЗМЕЛЬЧЕНИЯ РУДОБОГАТИТЕЛЬНОЙ ФАБРИКИ	227
MONITORING AND MANAGING SYSTEM OF MICROCLIMATE INDICATORS IN EDUCATIONAL FACILITIES	230
ВИДИ МОНИТОРИНГУ ДОВКІЛЛЯ, НАСЛІДКИ ЗАБРУДНЕННЯ АТМОСФЕРНОГО ПОВІТРЯ ТА РОЗРОБКА СИСТЕМИ МОНИТОРИНГУ ВМІСТУ ШКІДЛИВИХ РЕЧОВИН В ПОВІТРІ	233
СЕКЦІЯ 7. SECURITY. ЗАХИСТ ІНФОРМАЦІЇ В КОМП'ЮТЕРНИХ СИСТЕМАХ ТА МЕРЕЖАХ	236
ЗАХИЩЕНЕ ПРОГРАМУВАННЯ В ОСНОВІ БУДЬ-ЯКОГО ЗАСТОСУНКУ	236
DEVELOPMENT OF INTRUSION DETECTION SYSTEM FOR WEB APPLICATIONS	239
ЗАХИСТ ІНФОРМАЦІЇ НА ФІЗИЧНИХ НОСІЯХ ТА В МЕРЕЖІ ІНТЕРНЕТ	240
ІНФОРМАЦІЙНА БЕЗПЕКА ЯК ОСНОВА НАЦІОНАЛЬНОЇ БЕЗПЕКИ	242
ОХОРОННА СИГНАЛІЗАЦІЯ ТА КРИПТОСТІЙКІ АЛГОРИТМИ ЇЇ ШИФРУВАННЯ	245
ЗАХИЩЕНЕ ЛОКАЛЬНЕ МЕРЕЖЕВЕ СЕРЕДОВИЩЕ ДЛЯ ЗБЕРІГАННЯ ФАЙЛІВ НА ОСНОВІ NAS	248
СИСТЕМА ЗАХИСТУ КОМП'ЮТЕРІВ МЕТОДОМ ФІЛЬТРАЦІЇ МЕРЕЖЕВОГО ТРАФІКУ	250
НАВЧАЛЬНІ КУРСИ НА ПЛАТФОРМІ GOOGLE CLASSROOM	252
ДЕМОНСТРАЦІЯ ОСНОВНИХ МЕТОДІВ КОДУВАННЯ ЗАСОБАМИ WOLFRAM DEMONSTRATION PROJECT	254

Наукове видання

КОМП'ЮТЕРНІ ІНТЕЛЕКТУАЛЬНІ СИСТЕМИ ТА МЕРЕЖІ

Матеріали конференції
24-26 березня 2020 р.

Матеріали

XIII Всеукраїнської науково-практичної WEB конференції аспірантів,
студентів та молодих вчених «КІСМ-2020»

Вчений секретар
Комп'ютерна верстка

Івченко Р. А.
Саяпін В. Г.

Здано в набір 18.03.20. Підписано до друку 20.03.20
Формат 60×84 1/8. Папір офсетний. 9 ум. друк. аркушів. Тираж 100 прим.

Оригінал-макет виготовлено на кафедрі
комп'ютерних систем та мереж
Криворізький національний університет

Адреса видавництва:
50027, Кривий Ріг, вул. Віталія Матусевича, 11
Криворізький національний університет