

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА

ГУМЕНЮК КРІСТІНА ВОЛОДИМИРІВНА

Допускається до захисту:
в.о. завідувача кафедри
інформаційних технологій
канд. техн. наук, доцент
_____ О. В. Зелінська
« ____ » _____ 20__ р.

**ВЕБСЕРВІС ДЛЯ ОНЛАЙН ЗАМОВЛЕНЬ ВІЙСЬКОВОГО
СПОРЯДЖЕННЯ**

Спеціальність 122 Комп'ютерні науки

Кваліфікаційна (бакалаврська) робота

Керівник:

Т. В. Січко, доцент кафедри
інформаційних технологій,
к. т. н., доцент

Оцінка: _____ / _____ / _____
(бали/за шкалою ЄКТС/за національною шкалою)

Голова ЕК: _____

Вінниця - 2024

АНОТАЦІЯ

Гуменюк К.В. Вебсервіс для онлайн замовлень військового спорядження. Спеціальність 122 «Комп'ютерні науки», освітня програма «Комп'ютерні науки». Донецький національний університет імені Василя Стуса, Вінниця, 2020.

У кваліфікаційній (бакалаврській) роботі досліджено розробку вебсервісу (вебсайту) для онлайн замовлень військового спорядження, інакше кажучи, онлайн-магазин.

Клієнт (користувач) може здійснити реєстрацію свого особистого кабінету. Обирати необхідний йому товар, додавати в кошик та здійснювати оформлення замовлення. Вебсервіс було розроблено за допомогою Html, JavaScript та додаткових технологій.

Ключові слова: вебсервіс, JavaScript, Html, онлайн-магазин, військове спорядження.

Сторінки 67 , 59 рис., 42 джерела.

ABSTRACT

Humeniuk K.V. Web service for online orders of military equipment. Specialty 122 “Computer Science”, educational program “Modern Information Technologies and Programming”. Vasyl' Stus Donetsk National University, Vinnytsia, 2020.

The qualification (bachelor's) thesis investigates the development of a webservice (website) for online orders of military equipment, in other words, an online store.

The client (user) can register his personal account. He can choose the product he needs, add it to the cart and place an order. The webservice was developed using Html, JavaScript and additional technologies.

Keywords: webservice, JavaScript, Html, online store, military equipment.

Pages 67, 59 figure, 42 sources.

ЗМІСТ

ВСТУП	4
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ	7
1.1 Поняття вебсайту.....	7
1.2 Аналіз останніх досліджень	10
1.3 Огляд видів сайту	11
1.4 Постановка задачі.....	21
РОЗДІЛ 2. АНАЛІЗ ВИКОРИСТАНИХ ТЕХНОЛОГІЙ	23
2.1 Технології веброзробки HTML та JavaScript	23
2.2 Фреймворки Tailwind CSS та Next.js.....	27
2.3 JavaScript бібліотека – React	31
2.4 Система управління базами даних MongoDB	33
2.5 Додаткові інструменти VSCode та Figma	37
РОЗДІЛ 3. РЕАЛІЗАЦІЯ ВЕБСЕРВІСУ ВІЙСЬКОВОГО СПОРЯДЖЕННЯ	42
3.2 Розробка дизайну вебсервісу військового спорядження	42
3.3 Програмна реалізація вебсервісу військового спорядження	53
ВИСНОВКИ.....	62
СПИСОК ЛІТЕРАТУРИ.....	64

ВСТУП

У сучасному світі, характеризованому стрімким технологічним прогресом та швидким розвитком електронної комерції, інтернет-платформи стають невід'ємною частиною життя людей, забезпечуючи зручні та ефективні способи отримання товарів та послуг. У зв'язку з цим, зростає попит на вебсервіси, спрямовані на задоволення потреб різних груп користувачів. Однією з таких груп є військові, які потребують спеціального спорядження для виконання своїх обов'язків.

Сьогодні, у часи, коли країна переживає складні випробування та веде боротьбу за свою територіальну цілісність, забезпечення військових підрозділів належним спорядженням є важливим питанням. У світлі цих обставин, розробка ефективної та зручної платформи для онлайн замовлення військового спорядження набуває особливої актуальності. Зважаючи на потреби та вимоги сучасного бойового середовища, створення вебсервісу для замовлення військового обладнання є важливим етапом у покращенні системи забезпечення українських військ.

Військове спорядження відіграє критичну роль у підтримці військових дій. Якість та належний стан спорядження може вирішувати успіх чи невдачу військової місії. Тому, забезпечення військових необхідним спорядженням має важливе стратегічне значення. У контексті цього, використання інтернет-технологій та розробка вебсервісів для онлайн замовлень військового спорядження стає актуальною та перспективною задачею.

Актуальність вебсервісів в сучасному світі важко переоцінити. Вони стали не лише платформою для комунікації та розваг, але й незамінним інструментом для комерції та бізнесу. Онлайн магазини стали важливими компонентами економічної системи, дозволяючи зручно та ефективно здійснювати покупки та продажі.

Головною ціллю у розробці вебсервісу для онлайн замовлень військового спорядження є забезпечення простоти та зручності замовлення для клієнтів, а також, інтуїтивно зрозумілого інтерфейсу.

Мета бакалаврської роботи: створення зручного та інтуїтивно зрозумілого вебсервісу, який дозволить швидко здійснювати замовлення.

Задачі дослідження:

- аналіз потреб військових у спеціальному спорядженні;
- вивчення існуючих вебсервісів для замовлення спорядження та їх аналіз;
- розробка дизайну та функціоналу вебсервісу для замовлення військового спорядження;
- реалізація вебсервісу на практиці;
- тестування та вдосконалення розробленого вебсервісу.

Об'єктом дослідження є процес створення та впровадження вебсервісу для онлайн замовлень військового спорядження.

Предметом дослідження є архітектура та функціональні можливості вебсервісу для онлайн замовлень військового спорядження.

Апробація результатів дослідження. Результати кваліфікаційної (бакалаврської) роботи апробовано на II Міжнародній науково-практичній конференції «Сучасний стан та основні пріоритети розвитку економіки», яка відбулась 14 квітня 2024 року. на базі Східноєвропейського центру наукових досліджень (м. Одеса). Тези на тему: «Адаптивний дизайн веб-сайтів: принципи та методи», були опубліковані в електронному збірнику наукових праць, який можна переглянути за посиланням <https://researcheurope.org/book-49/>.

Результати роботи апробовано на XXIV Всеукраїнській науково-технічній конференції молодих вчених, аспірантів та студентів «Стан, досягнення та перспективи інформаційних систем і технологій», яка відбулась 18-19 квітня 2024 р. - Одеса, Видавництво ОНТУ, 2024 р. Тези на тему «Оптимізація продуктивності веб-сайтів», що були опубліковані в електронному збірнику наукових праць,

який можна переглянути за посиланням https://ontu.edu.ua/download/konfi/2024/Conference_abstract-IT-2024.pdf.

Також, результати роботи апробовано на V Всеукраїнській науково-практичній конференції здобувачів вищої освіти та молодих вчених «Прикладні інформаційні технології», що відбулася 24 травня 2024 року, м. Вінниця: ДонНУ імені Василя Стуса. Тези на тему «Методи анімації вебінтерфейсів: вплив на користувачський досвід», що були опубліковані в електронному збірнику наукових праць.

Структура роботи: кваліфікаційна робота складається зі вступу, трьох розділів, висновків та списку використаних джерел.

Робота містить 67 сторінок, 59 рисунків, та список літератури з 42 найменувань.

РОЗДІЛ 1

АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Поняття вебсайту

Вебсайт - це місце в Інтернеті, де користувач може легко розміщувати інформацію про себе, свій бізнес або теми, які його цікавлять. Це все одно, що мати власний куточок в Інтернеті для демонстрації своїх ідей та спілкування з іншими. Незалежно від того, чи хочете користувач поділитися знаннями, продавати онлайн, спілкуватися з іншими, дізнатися щось нове або просто розважитися - вебсайти служать різним цілям. Щоб отримати доступ до вебсайту, користувачам потрібен пристрій з веббраузером - ноутбук, смартфон або планшет - і підключення до Інтернету.

Вебсайт – це група вебсторінок що пов’язані між собою та мають однакову назву доменного ім'я. Вебсайти можуть створювати та підтримувати окремі особи, команди, компанії та установи.

Хоча іноді вебсайт називають «вебсторінкою», це визначення є неправильним, оскільки вебсайт складається з декількох вебсторінок. Усі суспільні вебсайти разом утворюють Всесвітню павутину.

Від особистих блогів до інтернет-магазинів - вебсайти стали невід'ємною частиною нашого життя, даруючи зручність і доступність. Незалежно від того, чи йдеться про розваги, освіту чи бізнес, вебсайти - це цифрові платформи, які дозволяють спілкуватися, обмінюватися інформацією та здійснювати онлайн-транзакції. Крім того, організації та неприбуткові установи використовують вебсайти для підвищення обізнаності про свою діяльність та зв'язку з потенційними спонсорами.

Інтернет та мережа є основою для вебсайтів. Інтернет - це технологія або інфраструктура, що з'єднує комп'ютери та полегшує обмін інформацією. Мережа - це система, яка дозволяє обмінюватися інформацією за допомогою Інтернету. Вона охоплює цифрові документи, вебсайти, вебсторінки, засоби масової

інформації тощо. Для перегляду вебконтенту ми використовуємо веббраузери, такі як Google Chrome, Microsoft Edge, Internet Explorer або Safari.

Коли користувач вводить вебадресу або доменне ім'я в браузері, комп'ютер надсилає запит на сервер хостингу. Цей запит проходить через систему доменних імен (DNS) для пошуку IP-адреси сервера. Вебсервіси передають інформацію та файли через Інтернет за допомогою таких протоколів, як протокол передачі гіпертексту (HTTP), захищений протокол передачі гіпертексту (HTTPS) і протокол передачі файлів (FTP).

Коли сервер отримує запит, він надсилає необхідну вебсторінку разом із зображеннями та іншими файлами на комп'ютер. Потім браузер відображає вебсторінку за допомогою таких технологій, як мова розмітки гіпертексту (HTML) і каскадних таблиць стилів (CSS). Ці технології структурують і представляють інформацію на екрані. Час завантаження вебсайту залежить від ряду факторів, таких як швидкість сервера, якість інтернет-з'єднання, розмір і складність вебсайту. На рис. 1.1 показано як працює вебхостинг [1].



Рисунок 1.1 - Робота веб-хостингу

Багато сайтів мають стандартний шаблон домашньої/головної сторінки, яка містить посилання на інші категорії та вміст сайту [2]. Домашня сторінка (чи просто «головна сторінка») - це головна сторінка самого вебсайту, яка зазвичай, є свого роду «центром», через який можна, відповідно, отримати доступ до всіх інших вебсторінок. Внутрішня вебсторінка, на яку пов'язано декілька інших сторінок із узгодженою структурою (наприклад, певна категорія тем), називається - «батьківською сторінкою».

Кожна сторінка є окремим HTML-документом, усі вони, пов'язані між собою за допомогою гіперпосилань (або просто «лінків»), і можуть бути об'єднані в панелі навігації для зручності використання.

Панелі навігації відображаються на всіх сторінках, а не лише на головній сторінці, і дозволяють користувачам швидко переходити по основній будові вашого вебсайту.

Іншою важливою частиною більшості вебсайтів є нижній колонтитул, ще один повторюваний розділ, розташований унизу кожної сторінки. Зазвичай колонтитул містить зовнішні посилання на схожі вебсайти та інші зовнішні ресурси, а також застереження, умови використання, політику конфіденційності, посилання на сторінку контактів і адресу організації, що володіє вебсайтом. Доступ до вебсайту можна отримати безпосередньо, ввівши його Uniform Resource Locator (URL) адресу або виконавши пошук у пошуковому сервісі. Класифікацію вебсайтів здійснюють за допомогою доменів верхнього рівня. Нижче представлені деякі приклади:

- вебсайти урядових установ — .gov;
- вебсайти навчальних установ — .edu;
- сайти неприбуткових організацій — .org;
- комерційні вебсайти — .com;
- інформаційні сайти — .info.

Незважаючи на те, що ці розширення домену все ще існують, насправді, вони дають, вкрай мало інформації про фактичний вміст вебсайту. В Інтернеті розширення “com” є найбільш популярним як і багато інших розширень, що мають відношення до конкретних країн (.it, .co.uk, .de, .fr і тому подібні).

Адреса вебсайту, також відома як уніфікований показник ресурсу (Uniform Resource Locator, URL) - є посиланням в Інтернеті на ресурс. Коли користувачі хочуть відкрити вебсайт, вони повинні ввести уніфікований показник ресурсу url-адресу вебсайту у своєму вебпереглядачі. URL складається з

декількох компонентів, розглянемо на прикладі сайту університету (<https://www.donnu.edu.ua/en/about/>):

- Протокол: структура, яка вказує, як отримати доступ до ресурсу. Найпоширеніші протоколи включають HTTP (Hypertext Transfer Protocol) і HTTPS (HTTP Secure). Наприклад, «<https://>».
- Ім'я сервера: вказує на доменне ім'я та розширення, де знаходиться вебсервер «[donnu.edu.ua](https://www.donnu.edu.ua/en/about/)».
- Шлях (ім'я сторінки): вказує на конкретну сторінку або ресурс на цьому вебсервері «[/en/about/](https://www.donnu.edu.ua/en/about/)».

1.2 Аналіз останніх досліджень

Останні дослідження вказують на значне зростання електронної комерції в контексті військових товарів. Споживачі все більше використовують онлайн-платформи для придбання спорядження, завдяки зручності, широкому вибору та доступності. Це підкреслює важливість забезпечення простоти навігації, безпеки транзакцій та наявності широкого асортименту товарів.

Споживачі цінують високу якість та надійність товарів, а також інформативність у виборі відповідного спорядження. Це вимагає від розробників сайту акцентування уваги на характеристиках продуктів та їх користувацьких оглядах.

Останні дослідження в галузі користувацького досвіду та інтерфейсів підкреслюють важливість простоти, інтуїтивності та зручності в дизайні сайту. Оптимізована взаємодія з користувачем, швидкість завантаження сторінок є важливими аспектами для сучасного магазину.

З огляду на поточну ситуацію в Україні, важливість доступності військових товарів через електронну комерцію значно зросла. Онлайн-платформи стали життєво важливими для забезпечення військовослужбовців необхідним спорядженням.

Війна також вплинула на характер попиту, вимагаючи більшої уваги до безпеки та надійності товарів, а також до швидкості та якості обслуговування. Ці

аспекти стали критичними для покупців, особливо тих, хто безпосередньо бере участь у бойових діях.

Зростання електронної комерції також обумовлює потребу в ефективному маркетингу та рекламі, особливо у сферах соціальних медіа та контекстної реклами. Це сприяє підвищенню видимості продукції та залученню нових споживачів.

Компанії, які пропонують безпечні платіжні методи та надають гарантії якості своїх товарів, здобувають велику прихильність клієнтів. Ці фактори можуть стати вирішальними при виборі конкретного магазину для здійснення покупки.

1.3 Огляд видів сайту

Різнноманітні вебсайти допомагають нам у повсякденних справах, таких як приготування їжі чи здійснення покупок, підтримують нашу освіту, надаючи тонни даних, швидко інформують нас про новини, допомагають нам керувати, розраховувати, працювати та спілкуватися.

За даними Internet Live Stats, станом на серпень 2020 року кількість вебсайтів становить майже 1,8 мільярда [3]. Отже, розмаїття вебсайтів потребує дедалі більше різнноманітних дизайнерських підходів та рішень. Проте, якими б різнноманітними не були вебсайти, існують певні базові сторінки, характерні для більшості веб-структур. Розглянемо основні типи вебсторінок, які можна зустріти в Інтернеті.

Розглянемо поняття вебсайту «Блог». Блог - це інформаційний вебсайт або особистий журнал, який веде окрема особа чи організація. Здебільшого використовується для обміну думками на певну тему, наприклад, про технології чи моду [4]. Сьогодні існує безліч блогів, серед яких можна знайти особисті та професійні, деякі з них присвячені повсякденному життю, інші - вузьким темам чи сферам. Блогери діляться публікаціями, які стають основним контентом, навколо якого функціонує вебсайт. У цьому випадку блог є різновидом вебсайту.

Щоб залишатися на хвилі популярності та займати високі позиції в пошукових системах, сайти мають не лише відповідати дизайнерським та технічним стандартам, а й регулярно оновлювати свій контент. Саме тоді в гру вступають блоги. Зараз можна знайти блоги, інтегровані в інтернет-магазини, корпоративні сайти, освітні платформи та портфоліо. І це той випадок, коли блог не є типом вебсайту, а є частиною вебсайту, що представляє цікаві новини та статті за темою вебсайту (рис. 1.2).



Рисунок 1.2 - Приклад веб-сайту блогу з рецептами десертів

Розглянемо Ecommerce вебсайти. Вебсайт електронної комерції - це будь-який сайт, який полегшує купівлю та продаж товарів і послуг. Вебсайти електронної комерції забезпечує бізнесу можливість ефективно виконувати ключові операції, такі як обробка замовлень від клієнтів, організація прийому платежів, управління процесами доставки та логістики, а також надання належного рівня обслуговування споживачів.

Цифрові ринки, інтернет-магазини та аукціони вважаються сайтами електронної комерції, оскільки вони дозволяють споживачам купувати товари. Будь-який власник бізнесу, який має товар або послугу на продаж, може отримати вигоду від сайту електронної комерції, зокрема для таких випадків [5]:

- малий бізнес, який намагається знайти місцевих клієнтів;
- фрілансер, який продає свої послуги;
- велика корпорація, яка демонструє свою продукцію.

Що відрізняє вебсайт електронної комерції від інших типів вебсайтів, так це його здатність обробляти транзакції. Транзакції в електронній комерції - це джерело життєдіяльності будь-якого інтернет-магазину, що дозволяє обмінюватися товарами, послугами та коштами в Інтернеті. Ці транзакції підтримуються платформами електронної комерції - спеціалізованими програмними рішеннями, які надають необхідні інструменти електронної комерції для управління продажами, запасами, клієнтами та іншими важливими аспектами бізнес-операцій в Інтернеті. Конструктори сайтів, такі як Wix, включають функції електронної комерції у свої платформи, щоб допомогти створити успішний інтернет-магазин [6].

Електронна комерція працює по-різному: Продавці можуть створити для себе вебсайт електронної комерції, створити інтернет-магазин на платформі електронної комерції, такий як Shopify або Magento, або створити вітрину на ринку, такому як Amazon або eBay.

Незалежно від способу, всі магазини повинні мати безпечні платіжні шлюзи для полегшення платежів, а також, повинні дотримуватися стандартів безпеки даних Індустрії платіжних карток (PCI).

Ось як це відбувається:

- покупці відвідують інтернет-магазин або вебсайт і переглядають товари за допомогою різних пристроїв, таких як смартфони, ноутбуки, настільні комп'ютери, планшети, смарт-телевізори та годинники;

- зазвичай клієнти додають товари, які вони хочуть купити, до віртуального кошика для покупок. У кошику відстежуються вибрані товари, їхня кількість і загальна вартість;
- коли клієнти готові до покупки, вони переходять на сторінку оформлення замовлення. Вони надають інформацію про доставку, обирають спосіб оплати та переглядають своє замовлення, перш ніж завершити покупку;
- продавець отримує повідомлення про замовлення;
- потім продавець готує товари до відправки клієнту;

Сфера електронної комерції дуже широка, але типи сайтів, на яких проводяться електронні транзакції, можна розділити відповідно до залучених сторін [7]:

- бізнес-споживач (Business-to-consumer, B2C): ймовірно, найбільш знайомий пересічній людині сайт B2C дозволяє обмінюватися товарами або послугами між бізнесом і споживачем, наприклад, купувати футболку в улюбленому інтернет-магазині;
- бізнес для бізнесу (Business-to-business, B2B): платформа B2B полегшує електронні транзакції між двома компаніями. Наприклад, якщо ви є власником компанії, яка продає футболки, ви можете купити ці футболки в онлайн-оптовика;
- бізнес-адміністрування (Business-to-administration, B2A): сайт B2A полегшує електронний обмін між організацією та державною установою, наприклад, вебсайт компанії, яка розробила вебпортал вашого міста;
- від споживача до споживача (Consumer-to-consumer, C2C): сайт C2C, часто відомий як маркетплейс, забезпечує обмін товарами між 2 або більше споживачами. Прикладами є такі сайти, як Etsy та eBay;
- від споживача до бізнесу (Consumer-to-business, C2B): на сайті C2B приватні особи пропонують товари або послуги підприємствам. Це може бути позаштатний SEO-експерт, який працює з компаніями в певній галузі, або інфлюенсер, якому платять за просування продуктів компанії;

- від споживача до адміністрації (Consumer-to-administration, C2A): подібно до B2A, цей тип сайтів дозволяє споживачам надавати інформацію, товари або послуги органам державного управління та урядовим організаціям - наприклад, оплатити паркувальний талон на вебпорталі вашого міста.

Приклад сайту наведено на рис. 1.3.

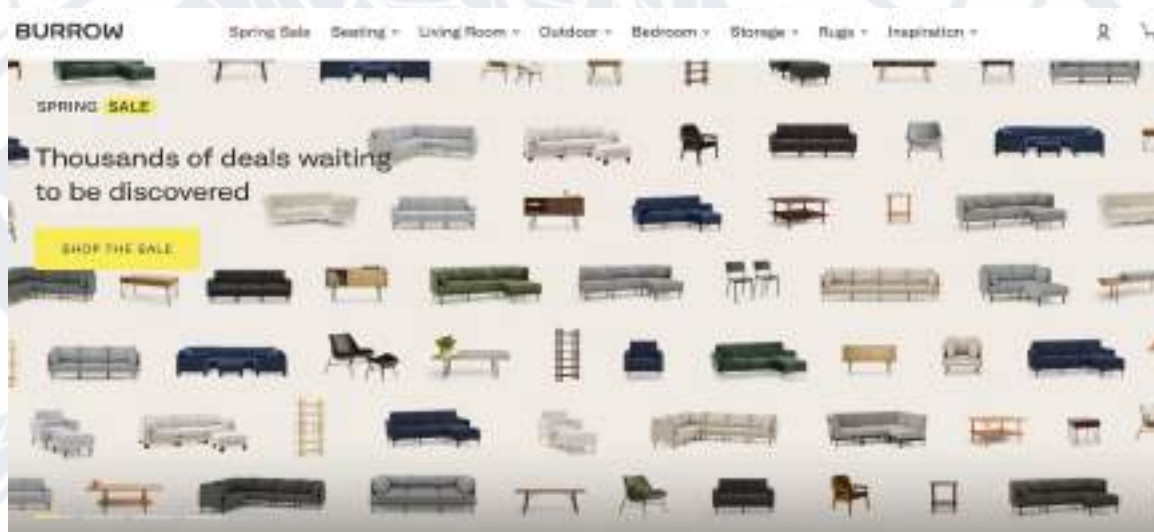


Рисунок 1.3 - Приклад Ecommerce веб-сайту

Розглянемо поняття портфоліо-вебсайту. Якщо користувач фотограф, художник чи письменник, вебсайт-портфоліо може допомогти йому знайти потенційних роботодавців чи клієнтів і справити на них враження, і слугуватиме своєрідним мистецьким резюме. Вебсайт-портфоліо зазвичай містить зображення, відео або кліпи, які демонструють найкращі роботи користувача. Вебсайт-портфоліо також дозволяє створити власний бренд, даючи відвідувачам уявлення про особистість, інтереси та цінності користувача [8].

Цей тип сторінок характерний для вебсайтів, метою яких є професійна презентація, найчастіше для людей або команд, які створюють візуальний контент, такий як фотографії, дизайн, мистецтво, відео, або фізичні товари, наприклад, вироби ручної роботи, одяг, ексклюзивний декор тощо. Можна знайти такі сторінки на професійних платформах, де користувачі можуть створювати власні портфоліо. Або це може бути спеціальна сторінка на персональному сайті. Мета такої сторінки - показати проекти клієнтам з найкращого боку, тому

рекомендується добре попрацювати над якісними зображеннями товарів або дизайнів та оригінальним способом їх виконання з крапелькою вау-ефекту для відвідувачів сайту (рис.1.4).



Рисунок 1.4 - Приклад сайту Портфоліо для фотографа

Розглянемо поняття Landing Page. Landing page - це поодинокі вебсторінки, що створені спеціально для маркетингової або рекламної кампанії. Це місце, куди відвідувач потрапляє після переходу за посиланням в електронному листі або рекламі з Google, Bing, YouTube, Facebook, Instagram, Twitter чи інших подібних місць в Інтернеті [9].

Це особливий вид вебсторінок, який може використовуватися як частина вебсайту, так і окремо. Основна мета цільової вебсторінки - представити чіткий і сфокусований контент на певній темі або задачі, що дозволяє відвідувачам опинитись саме там, де їм потрібно.

Створення спеціальних сторінок означає надання користувачам вказівки, що особливо корисно для великих платформ електронної комерції з сотнями або навіть тисячами товарів. Спрямовування всього трафіку на головну сторінку в

цьому випадку може призвести до поганого користувацького досвіду, особливо коли користувачі приходять з певних маркетингових кампаній на зовнішніх ресурсах. Існує високий ризик того, що вони одразу ж загубляться у величезній кількості контенту і посилань на головній сторінці, або ж їхня увага буде відвернута, і покупка не буде завершена.

Крім того, є багато інших випадків, окрім електронної комерції, коли цільові сторінки можуть бути корисними: вони можуть представляти мобільні додатки або освітні ресурси, рекламувати події та зустрічі, робити анонси, представляти спільноти або просто ділитися інформацією. Так чи інакше, процес розробки будь-якої цільової сторінки починається з постановки чіткої і ясної мети, яку слід досягти за допомогою цієї сторінки.

Вебсайти та landing page (рис.1.5) - це різні речі, але вони можуть мати групові риси. Наприклад, в обох прецедентах використовуються ключові слова SEO для покращення позицій бренду в пошукових системах. І в обох випадках використовується брендовий стиль та елементи дизайну, такі як кольорова гама, макет і типографіка, щоб створити цілісний образ для відвідувачів [10].

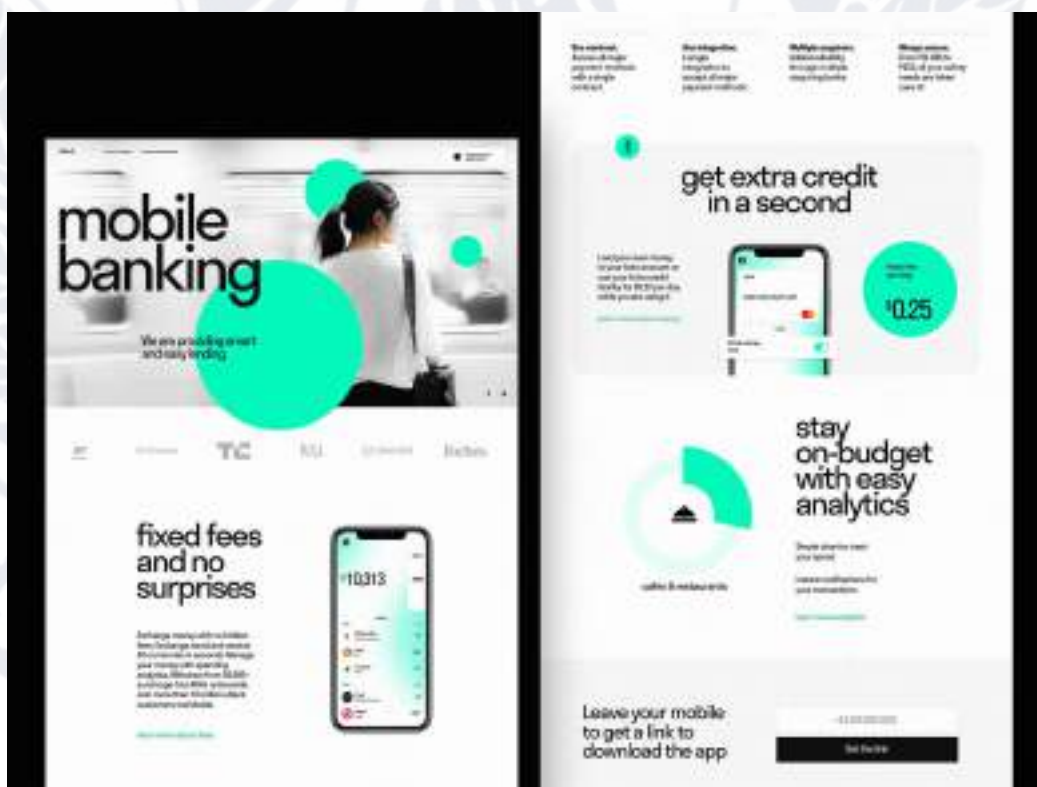


Рисунок 1.5 - Приклад Landing page для сервісу мобільного кредитування

Розглянемо поняття освітнього вебсайту. Освітній вебсайт - це онлайн-платформа, яка надає доступ до навчальних матеріалів і ресурсів для студентів, викладачів, батьків і майбутніх учнів. Він може включати систему управління навчанням (Learning Management System, LMS), таку як Moodle або Blackboard, яка дозволяє викладачеві керувати курсами, створюючи навчальний контент, відстежуючи прогрес студентів і забезпечуючи зворотний зв'язок [11].

Вони надають доступ до широкого спектру освітнього контенту та ресурсів, які, як правило, адаптовані до потреб окремих учнів [12]. Ці вебсайти можна використовувати для надання гнучких і доступних можливостей для навчання, що стало особливо актуальним після пандемії Covid-19.

Розробка та підтримка освітнього вебсайту (рис.1.6) може бути складним завданням, особливо для організацій, які потребують широкої кастомізації. Більше того, існує ризик низького рівня залучення та завершення навчання серед учнів, що може вплинути на ефективність освітнього вебсайту.



Рисунок 1.6 - Приклад освітнього вебсайту

Розглянемо поняття корпоративного вебсайту. Корпоративний сайт, або, як його ще називають, організаційний сайт - це інтернет-сайт, який представляє компанію, установу чи організацію всім її аудиторіям, а не лише її клієнтам

чи користувачам. Таким чином, корпоративний сайт класично відрізняється від комерційного сайту, призначеного тільки для клієнтів або користувачів. З точки зору змісту, корпоративний сайт має на меті просувати компанію (її історію, цінності, зобов'язання, мету, управління, кадрову політику тощо), а не безпосередньо її продукти чи рішення [13].

Корпоративний сайт - це не тільки засіб презентації та збільшення продажів, але й важливий елемент просування практично для будь-якої сучасної організації. До найпоширеніших цілей корпоративного сайту відносяться [14]:

- підвищення іміджу та репутації компанії;
- збільшення кількості клієнтів і партнерів;
- інформування цільової аудиторії про діяльність організації;
- продаж товарів і послуг;
- залучення нових співробітників.

Існує кілька типів корпоративних порталів:

- односторінковий промо-сайт;
- лендінг або сайт-візитка;
- багатосторінковий портал;
- корпоративний інформаційний ресурс;
- сайт-каталог.

Управління корпоративним сайтом вимагає більше зусиль, ніж здається на перший погляд. Робота триває і після запуску сайту. Його необхідно регулярно підтримувати в актуальному стані. Основна увага приділяється змісту та дизайну вебсайту. Команда повинна оновлювати інформацію та поширювати новини про свою діяльність, дотримуючись при цьому правил копірайту, які постійно змінюються. Необхідно розробити систему публікації контенту, щоб полегшити процес і залучити більше трафіку, який потім можна конвертувати в реальних клієнтів. Відстеження останніх тенденцій вебдизайну необхідне для усунення недоліків і підвищення довіри. Приклад сайту наведено на рис. 1.7.

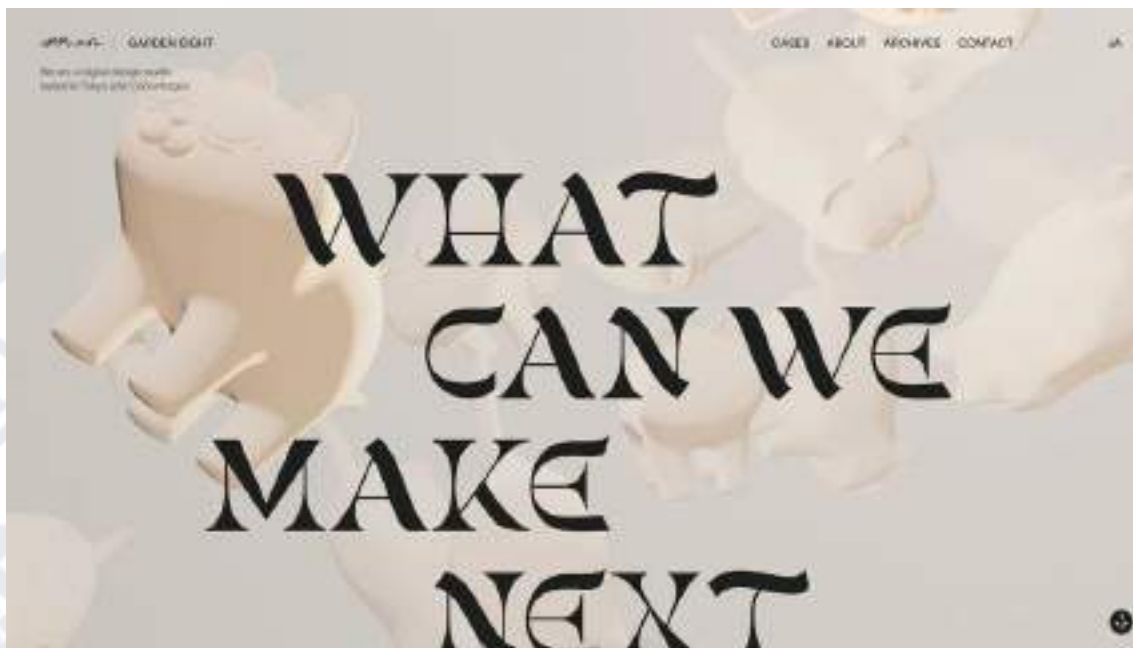


Рисунок 1.7 - Приклад корпоративного вебсайту

Розглянемо поняття новинного вебсайту. Новинні вебсайти - це онлайн-платформи, які надають актуальну інформацію про поточні події, тенденції та розробки з різних сфер, таких як політика, економіка, наука, технології, розваги тощо. У них є основні функції [15].

Висвітлення новин: новинні вебсайти збирають і представляють новини з усього світу. Журналісти та репортери розслідують і висвітлюють події, беруть інтерв'ю та проводять дослідження, щоб надати точну і своєчасну інформацію.

Редакція та думки: на багатьох новинних вебсайтах є редакційні розділи, де автори та експерти висловлюють свої думки та аналізують поточні події. Ці матеріали мають на меті надати контекст, розуміння та різні точки зору на новини.

Мультимедійний контент: окрім письмових статей, вебсайти новин часто включають мультимедійні елементи, такі як зображення, відео та інфографіку, щоб покращити розповідь і забезпечити більш повне розуміння новин.

Оновлення в режимі реального часу: новинні вебсайти постійно оновлюють свій контент, щоб відображати найсвіжішу інформацію. Це дозволяє читачам бути в курсі подій, що швидко розвиваються, та останніх новин.

Категоризація та навігація: новинні вебсайти класифікують новини за різними розділами, що полегшує читачам навігацію та пошук інформації, яка відповідає їхнім інтересам. Найпоширеніші категорії включають політику, бізнес, спорт, розваги, науку та спосіб життя.

Архіви та функція пошуку: новинні вебсайти зазвичай зберігають архіви минулих статей, що дозволяє користувачам отримати доступ до історичних новин. Вони також надають функцію пошуку, що дозволяє користувачам знаходити конкретну інформацію або статті у своїх величезних базах даних.

Інтеграція з соціальними мережами: багато новинних вебсайтів інтегруються з платформами соціальних мереж, щоб ділитися новинами, взаємодіяти зі своєю аудиторією і заохочувати поширення статей у різних соціальних мережах. Приклад сайту наведено на рис. 1.8.



Рисунок 1.8 - Приклад новинного вебсайту

1.4 Постановка задачі

Метою дипломної роботи є створення вебсервісу для продажу військового спорядження. Сервіс має на меті забезпечити користувачам доступ до вибору якісного військового спорядження, надійної інформації про продукцію, а також можливість купувати товари онлайн.

Головними задачами є:

- розробка дизайну: необхідно створити привабливий, простий та інтуїтивно зрозумілий дизайн сервісу, який відповідатиме військовій тематиці та викликатиме довіру у користувачів;
- інформаційний контент: необхідно забезпечити детальний інформаційний опис про наявні товари, включаючи їх специфікації, фотографії, та рекомендації щодо використання;
- функціональність інтернет-магазину: потрібно впровадити можливість купівлі товарів онлайн, сортування за категоріями, а також безпечні методи оплати;
- створення панелі адміністратора для управління товарами, замовленнями та клієнтами;
- впровадити систему реєстрації та входу для користувачів;
- тестування та забезпечення коректної роботи на різних пристроях.

Важливим є наявність достовірної та актуальної інформації про товари та їх характеристики. Враховуючи цілі та завдання проекту, цей вебсервіс належатиме до категорії комерційних (e-commerce) вебсервісів.

РОЗДІЛ 2

АНАЛІЗ ВИКОРИСТАНИХ ТЕХНОЛОГІЙ

2.1 Технології веброзробки HTML та JavaScript

Розглянемо поняття HTML: HyperText Markup Language. Гіпертекст спочатку означав текст, що зберігався в електронній формі з перехресними посиланнями між сторінками. Зараз це ширше поняття, яке стосується практично будь-якого об'єкта (тексту, зображень, файлів і так далі), який можна пов'язати з іншими об'єктами. Мова розмітки гіпертексту (Hypertext Markup Language, HTML) - це мова для опису того, як текст, графіка і файли, що містять іншу інформацію, організовані і пов'язані між собою [16].

HTML, або мова гіпертекстової розмітки, - це мова розмітки, яка забезпечує роботу Інтернету. Іншими словами, всі вебсторінки в Інтернеті використовують HTML. Коли користувач заходить на вебсторінку, веббраузер інтерпретує текстовий файл, написаний мовою HTML, який містить набір інструкцій для форматування вмісту сторінки так, щоб людина могла її прочитати. Зараз, порівнюючи з минулим, HTML живе в ширшій онлайн-екосистемі з іншими мовами, онлайн-системами управління контентом, пошуковими системами та агрегаторами вебсайтів [17].

Структура HTML-документу визначає, як організовано контент на вебсторінці. Для розуміння та візуалізації правильного синтаксису та компоновання HTML-документів розглянемо рис. 2.1 [18].

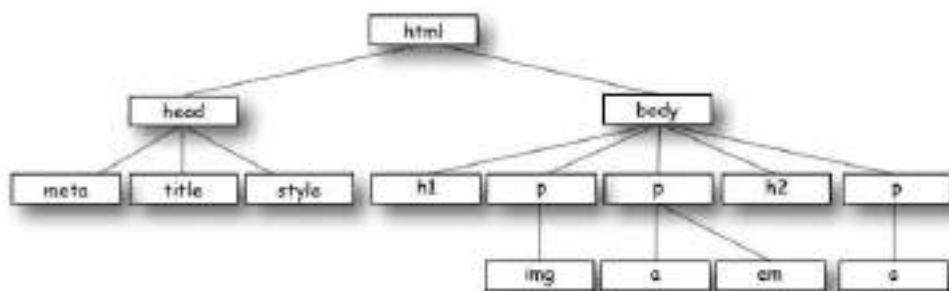


Рисунок 2.1 - Базова структура HTML-документа

Рис. 2.1. візуально демонструє ієрархічну організацію різних HTML-елементів і як вони вкладені один в одного.

На вершині знаходиться кореневий елемент "html". Від нього відгалужуються два основні вузли: "head" і "body".

У секції "head" є додаткові вузли "meta", "title" і "style", які зазвичай містять метадані документа, його заголовок і стилі CSS відповідно.

Секція "body" містить безпосередньо вміст вебсторінки, включаючи заголовки "h1" і "h2", параграфи "p", а також такі елементи як "img" (зображення), "a" (посилання) і "em" (виділений текст).

Ця дерево-подібна структура дозволяє наочно представити семантичну розмітку HTML і взаємозв'язки між різними елементами на вебсторінці.

HTML є основним стандартом веброзробки. Специфікації HTML встановлюються організацією World Wide Web Consortium (W3C), яка відповідає за розвиток стандартів Інтернету.

Розглянемо мову програмування – JavaScript. Напочатку JavaScript не мав своєї теперішньої назви. Спочатку він називався Mocha та LiveScript. Завдяки співпраці між Netscape і Sun (компанія, що стоїть за мовою програмування Java) із маркетингових міркувань (мова програмування Java була вже досить популярною на той час), її врешті-решт перейменували на JavaScript [19].

JavaScript - це не скомпільована мова програмування, а мова програмування з інтерпретатором. Вона дозволяє програмувати як фронтенд, так і бекенд вебдодатку. JavaScript можна використовувати для розробки мобільних додатків, десктопних додатків, вбудованих додатків, ігор та 3D-додатків.

Не менш важливим є те, що JavaScript використовується для додавання динамічної поведінки до вебсторінки (або до компонентів на вебсторінці) або для забезпечення більшої інтерактивності на вебсторінці. Прикладами цього є сортування та фільтрація даних таблиці. У той час як CSS відповідає за дизайн вебсторінки, JavaScript можна використовувати для покращення користувацького досвіду та інтерактивності вебсторінки.

За допомогою JavaScript можна “оживити” сторінки, додати автоматичне заповнення полів форм і всілякі навороти, які покращують користувацький досвід.

JavaScript визначається специфікацією, відомою як ECMA-262. Веббраузери мають різний ступінь підтримки специфікації ECMA-262, тому точна версія JavaScript, доступна в браузері, залежить від версії браузера, який ви використовуєте.

JavaScript сьогодні є невід'ємною частиною вебсторінок. Коли ви бачите щось на кшталт Google Maps, де ви можете прокручувати карту вліво і вправо, просто перетягуючи її, це і є JavaScript за лаштунками. Незліченні віджети та покращення зручності використання, які ми сприймаємо як належне, коли користуємося Інтернетом, насправді є програмами на JavaScript [20].

Однією з найсильніших сторін JavaScript є те, наскільки легко він справляється з асинхронним програмуванням. Як мова, створена для Інтернету, JavaScript з самого початку мала реагувати на дії користувача, як от кліки та натискання клавіш, тому були створені обробники подій, такі як onclick. Обробники подій дозволили розробникам вказати функцію, яка буде виконана в певний момент часу у відповідь на подію [21].

JavaScript може маніпулювати HTML і CSS, реагувати на події, взаємодіяти з користувачем, а також робити запити до серверів для отримання даних або відправлення інформації. JavaScript є слабкотипізованою мовою, і змінні не потребують явно оголошених типів даних. Це може спростити кодування, але слід зазначити, що це також може призвести до проблем з помилками.

JavaScript бібліотеки та фреймворки грають важливу роль у сучасній веброзробці. Вони надають вбудовані функції та методи, які роблять вебсторінки більш динамічними та інтерактивними. Вони позбавляють від повторюваних завдань і дозволяють розробникам зосередити усю свою увагу на основній функціональності. Вони також надають структуру проекту та фреймворк потоку даних для створення швидких і надійних додатків. На рис. 2.2 наведено схему, що ілюструє різноманітні бібліотеки, фреймворки та інструменти JavaScript [22].

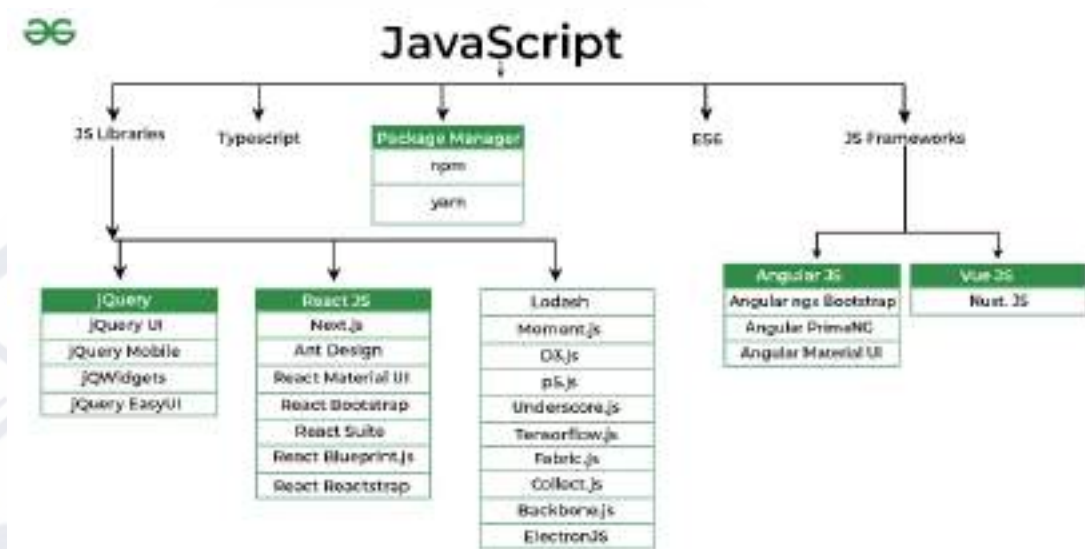


Рисунок 2.2 - Ключові компоненти та інструменти екосистеми JavaScript

У центрі схеми – JavaScript, як основна технологія. З нього випливає кілька основних категорій:

- **JSLibraries:** тут перераховані популярні бібліотеки JavaScript, такі як jQuery та її похідні (інтерфейсні, мобільні, віджети тощо);
- **Typescript:** розширення JavaScript з додатковою функціональністю для більш статичної типізації;
- **PackageManager:** менеджери пакунків npm та yarn для встановлення та керування залежностями JavaScript наведені тут;
- **ES6:** вказує на нову версію специфікації JavaScript (ECMAScript 6);
- **JS-Frameworks:** ця категорія включає основні фреймворки JavaScript, такі як Angular, React і Vue, а також бібліотеки та інструменти, похідні від них;
- також показано низку популярних бібліотек JavaScript, таких як Lodash, Moment.js, D3.js та Underscore.js;

Таким чином, діаграма надає огляд ключових компонентів та інструментів екосистеми JavaScript, які зазвичай використовуються у веброзробці.

2.2 Фреймворки Tailwind CSS та Next.js

Розглянемо поняття фреймворку Tailwind CSS. Каскадну таблицю стилів (Cascading Style Sheet, CSS) можна визначити як файл або фрагмент коду, який визначає набір правил у певній послідовності для створення певного зовнішнього стилю. CSS як окремий об'єкт не має суттєвого значення, він лише створює контекст. Цей набір письмових правил потребує посилання на певне місце, щоб створити візуальний ефект [23].

Візуальний вигляд – це визначення CSS. Загалом, якщо вебсайт виглядає візуально привабливо, це означає, що для досягнення такого вигляду використовується CSS.

Використання CSS у веброзробці означає додавання набору правил стилів до вебсторінки, тобто HTML-документа. Правила CSS приймаються, коли HTML-елементи відображаються у веббраузері. CSS вимагає каналу для рендерингу, і HTML-документ слугує цим каналом.

Тепер перейдемо безпосередньо до огляду Tailwind CSS.

Tailwind CSS - це фреймворк CSS з відкритим вихідним кодом, написаний Адамом Ватаном і Стівом Шогером, компанія-розробник - Tailwind Labs. Уперше він був представлений у травні 2019 року.

У версії Tailwind кожен стиль кнопки отримує свій клас у списку класів. Код Tailwind дуже зрозумілий, і потрібно лише поглянути на HTML-розмітку, щоб побачити, як виглядають кнопки. Якщо ви хочете упакувати набір класів для повторного використання, Tailwind надає директиву `@apply`, яка дозволяє створювати нові CSS класи з утиліт Tailwind.

Однією з переваг кастомізації Tailwind є те, що дуже легко створювати прототипи, ітерації та змінювати зовнішній вигляд елементів. Можна вносити невеликі зміни, поки не отримаєте бажаний вигляд. Крім того, не доведеться постійно придумувати назви для комбінацій властивостей CSS, які ви, можливо, ніколине будете використовувати.

Ще однією перевагою є те, що Tailwind пропонує набір префіксів, які дозволяють визначити поведінку в певних випадках. Tailwind також надає набір

префіксів, які дозволяють задавати різну поведінку для різних розмірів екрану. З використанням Tailwind витрачається менше часу на присвоєння імен CSS і управління глобальним CSS, що дозволяє зменшити час на розробку дизайну. Tailwind дозволяє легко вносити поетапні зміни, бачити результати і розуміти масштаб змін, що робить його особливо корисним при створенні прототипу нового сайту [24].

Tailwind - це набір низькорівневих утиліт багаторазового використання, які можна використовувати як цеглинки для створення практично будь-якого дизайну, який ми можемо собі уявити. Цей фреймворк охоплює найважливіші можливості CSS, але може бути легко розширений різними способами. Його можна використовувати як для швидкого прототипування, так і для створення повноцінного дизайну. Tailwind має чудову документацію, яка детально описує кожну утиліту класу та показує, як її можна налаштувати.

У той час як клас компонента - це набір попередньо визначених налаштувань CSS, які можна застосовувати будь-яким чином, клас утиліти - це, по суті, одна властивість або поведінка CSS, яку можна вільно використовувати у передбачуваний спосіб. Таким чином, їх можна вільно комбінувати, поєднувати з різними налаштуваннями за потреби. Зовнішнім виглядом кожного елемента можна керувати більш вільно. За допомогою утилітарних класів зовнішній вигляд елементів можна легше змінювати та налаштовувати [25].

Ключові особливості Tailwind CSS [26]:

- налаштування: Tailwind CSS розроблено так, щоб його можна було налаштовувати з нуля. Просто відредагувавши простий конфігураційний файл, ви можете легко змінювати кольори, розміри шрифтів, інтервали та інше відповідно до вимог дизайну вашого проекту;
- адаптивний дизайн: з Tailwind CSS легко створювати адаптивні вебсайти. Адаптивні класи утиліт дозволяють розробникам адаптувати макети та контент до різних розмірів екрану без написання медіа-запитів;

- підтримка Flexbox і Grid: у Tailwind CSS є широкий набір класів утиліт для роботи з Flexbox і Grid, які спрощують створення адаптивних макетів. Класи Flexbox дозволяють легко встановлювати гнучкі контейнери і вирівнювати елементи. Для роботи з Grid є класи для створення сіток, налаштування кількості колонок та проміжків між елементами;
- велика бібліотека компонентів: хоча Tailwind CSS не містить готових до використання компонентів, активна спільнота створила численні бібліотеки компонентів і шаблонів, які можна легко інтегрувати у проекти;
- інтеграція з PurgeCSS: щоб спростити ваш проект, Tailwind CSS легко інтегрується з PurgeCSS, видаляючи весь невикористаний CSS з фінальної збірки.

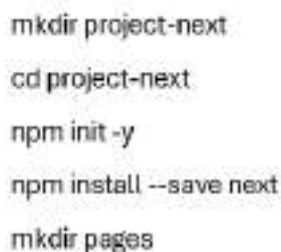
Розглянемо фреймворк Next.js. Компанія Vercel розробила Next.js, який змінив правила гри. З моменту свого першого випуску фреймворк забезпечив багато інноваційних функцій, таких як автоматичне розділення коду, рендеринг зі сторони сервера, маршрутизація файлової системи, попередня вибірка маршрутів іт.д.. Next.js показав, наскільки простим має бути написання універсальних вебдодатків, дозволив розробникам писати код, який повторюється як на стороні клієнта, так і на стороні сервера, роблячи дуже складні завдання (такі як розділення коду та рендеринг зі сторони сервера) простими для реалізації.

Next.js надає безліч нових можливостей [27]:

- статична генерація сайту;
- покрокова статична генерація;
- нативна підтримка TypeScript;
- автоматичне заповнення полів;
- оптимізація зображень;
- підтримка інтернаціоналізації;
- аналітика продуктивності.

І це ще не все, він має багато інших чудових можливостей. Сьогодні Next.js використовується такими компаніями, як Netflix, Twitch, TikTok, Hulu, Nike, Uber та Elastic. Next.js показує, наскільки універсальним є React для створення різноманітних додатків будь-якого розміру, тому не дивно, що його використовують як великі компанії, так і невеликі стартапи.

Next - це технологія з відкритим вихідним кодом, яку випустила компанія Zeit, щоб допомогти інженерам легше писати додатки для серверного рендерингу. Для того, щоб створити Next.js проект, потрібно виконати наступні команди (рис. 2.3) [28]:



```
mkdir project-next  
cd project-next  
npm init -y  
npm install --save next  
mkdir pages
```

Рисунок 2.3 - Створення проекту Next.js за допомогою командного рядка

1. Створюємо новий каталог, використовуючи команду `mkdir project-next`, де `project-next` є ім'ям папки нового проекту.
2. Переходимо до новоствореної папки за допомогою команди `cd project-next`.
3. Ініціалізуємо новий проект Node.js у папці проекту за допомогою команди `npm init -y`, що створює файл `package.json` із базовими налаштуваннями.
4. Встановлюємо пакет `next` у проект, використовуючи команду `npm install --save next`. Ці пакети є необхідними для розробки додатків на базі Next.js.
5. Створюємо папку `pages`, використовуючи команду `mkdir pages`. Ця папка є обов'язковою для проектів Next.js, оскільки в ній зберігаються сторінки вебдодатку.

Next.js – використовують багато інженерів, оскільки він пропонує неймовірні переваги продуктивності, такі як рендеринг на стороні сервера та

оптимізовані зображення. Next.js забезпечує чудовий досвід розробки, включаючи швидкі оновлення та якісні виправлення помилок. Крім того, Next.js легко інтегрується з широким спектром інструментів та сервісів, що спрощує розробку та аналіз і робить його ідеальним вибором для створення високопродуктивних вебсервісів [29].

2.3 JavaScript бібліотека – React

React - це унікальний інструмент, про необхідність якого веброзробники можуть навіть не здогадуватися, але після спроби його використання, вони вже не зможуть від нього відмовитися.

React пропонує новий підхід до спрощення фронтенд веброзробки при правильному використанні React пропонує альтернативу, яку багато великих компаній, включаючи Facebook, Netflix та Airbnb, прийняли і розглядають як шлях вперед: потужну бібліотеку користувацького інтерфейсу. Замість того, щоб визначати шаблон інтерфейсу користувача один раз, React дозволяє створювати багаторазові компоненти інтерфейсу користувача на JavaScript, які можна використовувати на вашому сайті знову і знову [30].

React дотримується декларативного підходу до програмування. Ви визначаєте бажаний стан додатку, а React створює відповідне представлення на основі цього стану. React аналізує дані браузера і створює власні представлення.

За останні кілька років зростає попит на прості способи створення користувацьких інтерфейсів за допомогою JavaScript. Фронтенд-розробники шукають готові рішення, які надаються низкою різних бібліотек та фреймворків. Популярність React у цій сфері не зменшується з моменту його першого релізу у 2013 році [31].

Коли розробник створює React-додатки, навіть якщо він створює простий додаток, наприклад, блог-додаток, навіть тоді доведеться працювати з маршрутизацією. Буде кілька екранів у додатку, таких як домашня сторінка, сторінка блогу, сторінка створення/редагування блогу, сторінка автора і так далі, і кожна сторінка матиме власний URL, який потрібно буде зіставити з потрібними

компонентами. Маршрутизація - це процес, під час якого користувача перенаправляють на різні сторінки залежно від його дії або запиту. Наприклад, на головній сторінці YouTube можна побачити сітку відео, і коли користувач натискає на будь-яку з них, вона перенаправляє на сторінку перегляду відео. Таке перенаправлення при натисканні називається маршрутизацією. React не має вбудованого API для роботи з маршрутизацією. Тому використовується пакет під назвою react-router для роботи з маршрутизацією. Він містить вбудовані компоненти для створення маршрутів для різних компонентів. Надає hooks для доступу до властивостей і методів, пов'язаних з місцезнаходженням, історією, параметрами пошуку, параметрами URL і так далі [32].

Простота використання та популярність React пояснюється його модульністю, підходом Virtual DOM, великою спільнотою розробників та універсальністю для кросплатформної розробки. Можливість створювати багаторазові компоненти, інтеграція з React Native та широка екосистема інструментів і бібліотек роблять його ідеальним вибором для розробників, які прагнуть створювати сучасні, масштабовані вебдодатки. JSX-код у React - це синтаксис, який дозволяє писати елементи інтерфейсу користувача шляхом поєднання JavaScript та HTML у звичний та виразний спосіб. Це робить створення інтерфейсів користувача в React простішим і більш читабельним [33].

На рис. 2.4 зображено ключові концепції та фундаментальні складові React [34].

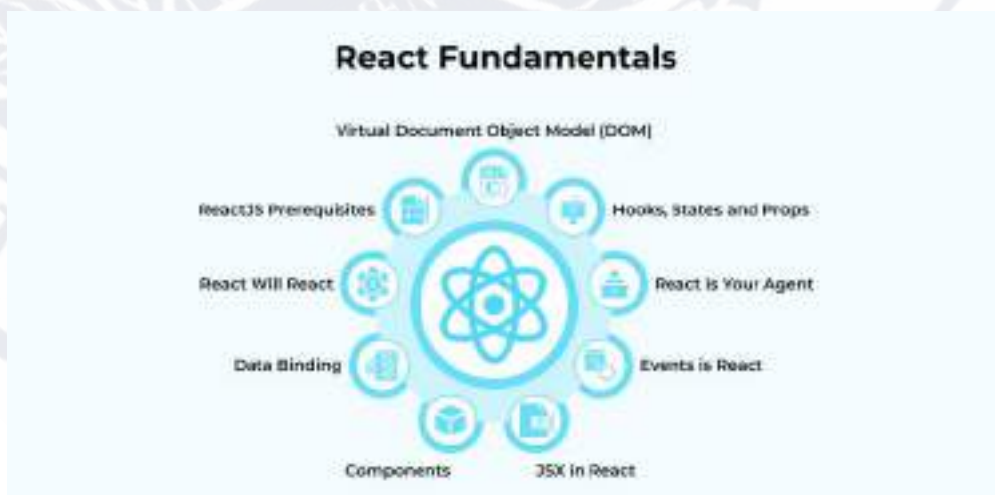


Рисунок 2.4 - Фундаментальні складові React

У центрі знаходиться логотип React, навколо якого згруповані різні аспекти та особливості цієї технології:

- Virtual Document Object Model (DOM) - віртуальне представлення DOM, що дозволяє ефективно оновлювати та змінювати користувацький інтерфейс;
- ReactJS Prerequisites - необхідні передумови, такі як знання HTML, CSS та JavaScript для роботи з React;
- Hooks, States and Props - концепції станів, властивостей та хуків для керування даними та життєвим циклом компонентів React;
- React Will React - основна ідея React про те, що він "реагує" та оновлює інтерфейс при змінах станів або властивостей;
- React is Your Agent - React виступає як агент, який керує відображенням та обробкою користувацького інтерфейсу;
- Data Binding - прив'язка даних до компонентів React для динамічного відображення;
- Events in React - обробка подій користувача в React;
- Components - компоненти як основні будівельні блоки React для створення UI;
- JSX in React - використання JSX синтаксису для написання компонентів React.

Зображення візуально представляє головні концепції та принципи роботи React, які необхідно засвоїти для ефективного використання цієї бібліотеки у веброботі.

2.4 Система управління базами даних MongoDB

MongoDB, найпопулярніша база даних документів, є NoSQL, нереляційним сховищем key-value, базою даних JSON тощо. Це надійна і багатифункціональна платформа даних для розробників з різноманітними вбудованими

функціями, необхідними для сучасних додатків, включаючи можливості машинного навчання і штучного інтелекту, потокове передавання даних, функції, тригери, безсерверну роботу, синхронізацію пристроїв і повнотекстовий пошук. Хоча MongoDB не є реляційною, вона може легко працювати з реляційними даними [35].

Зараз, у 2024 році, MongoDB використовується в безлічі різноманітних галузей, а варіанти її використання охоплюють різні сфери застосування та категорії даних, що зберігаються. Деякі з найбільших банків, автовиробників, державних установ та ігрових компаній у світі використовують MongoDB для своїх виробничих додатків. Найвідомішими користувачами MongoDB є Coinbase, Epic Games, Morgan Stanley, Adobe, Tesla, Canva, Ulta Beauty, Cathay Pacific, Dongwha та Vodafone.

MongoDB - це потужна, швидка і сучасна база даних, яка була розроблена для сучасних додатків. Завдяки гнучкій моделі документів (для даних), мові запитів на основі JavaScript, блискавичній швидкості, розширеній реплікації та масштабуванню, легкому резервному копіюванню, вдосконаленому фреймворку агрегації та підключенню до ширшої екосистеми MongoDB Atlas Cloud, вона може стати ідеальним вибором для вашого наступного проекту.

Залежно від вашого використання, MongoDB може обробляти набагато більше запитів, ніж база даних на основі SQL. Крім того, вона має багато унікальних і потужних способів запитів до даних і виконання складних агрегацій. Згідно з опитуванням Stack Overflow за 2022 рік (рис. 2.5) [36]: «MongoDB використовується в однаковій мірі як професійними розробниками, так і новачками, що навчаються кодуванню».



Рисунок 2.5 - Опитування розробників Stack Overflow щодо баз даних. 2022.

Протягом останніх шести років MongoDB посідає перше або друге місце в рейтингу «найбажаніших» технологій в категорії баз даних за версією Stack Overflow.

Запис у MongoDB називається документом. Документ - це структура даних, що складається з пар полів і значень. Документи MongoDB дуже схожі на об'єкти JSON, які складаються з пар Ключ-Значення. Значення полів можуть містити складні дані, зокрема масиви, інші документи та масиви документів. Приклад документа MongoDB наступний [37]:

```
{
  name: "Joe",
  age: 30,
  gender: "Male",
  interests: ["Maths", "Science"]
}
```

Ось декілька переваг використання документів:

- документи посилаються на власні типи даних у декількох мовах програмування;

- вбудовані документи та масиви зменшують потребу в складних з'єднаннях;
- динамічна схема підтримує плавний поліморфізм.

На рис. 2.6 показано шардинг MongoDB [38]:

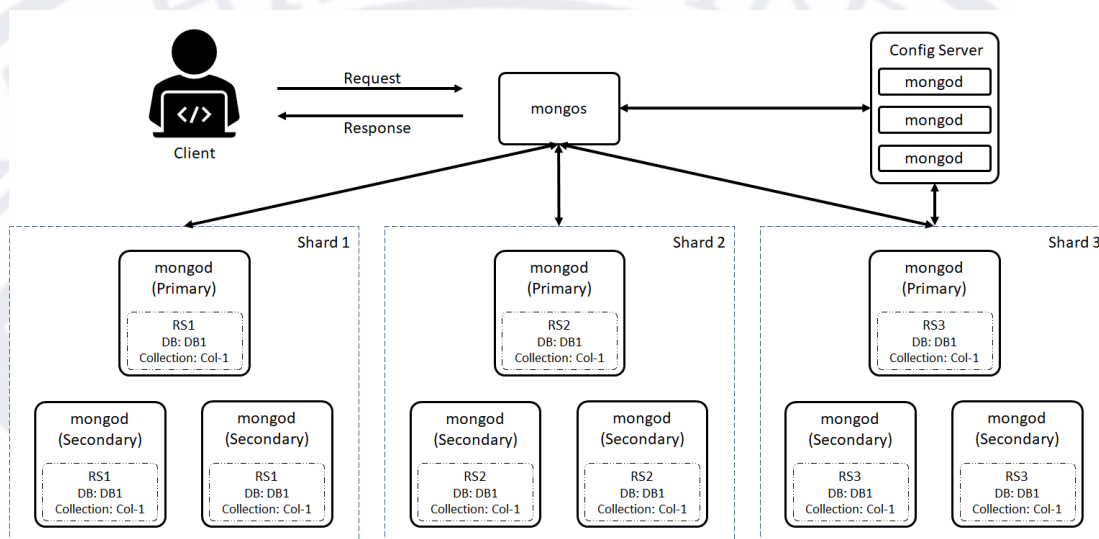


Рисунок 2.6 - Архітектура шардингу MongoDB

Mongos - це компонент, який діє як інтерфейс між клієнтом і кластером шардів. Цей екземпляр Mongos підключається до сервера конфігурації, щоб визначити, який шард має відповідати на який запит.

Config Server (Сервер конфігурації): містить метадані частини даних у кожному шарді. Він взаємодіє з усіма іншими компонентами БД, щоб гарантувати, що mongos може відповідати на запити в будь-який час.

Shard: шардинг - це концепція розподілу даних між декількома пристроями. Вона призначена для вирішення проблем масштабування і зростання в системах mongoDB.

Replica set: кожен шард розгортається як набір реплік. Кластер реплікації - це набір серверів MongoDB, які реалізують автоматизоване аварійне відновлення та реплікацію.

2.5 Додаткові інструменти VSCode та Figma

Розглянемо редактор коду Visual Studio Code. Вибір середовища розробки є важливим та незамінним кроком для кожного програміста. Від правильного вибору, у першу чергу, залежить ефективність роботи. Цей вибір залежить від ряду факторів, включаючи тип проекту, мову програмування, особисті уподобання та вимоги конкретного завдання. Зараз на ринку доступно багато середовищ програмування, кожне з яких має свої переваги та недоліки. Розробники шукають функціональність, комбінації гарячих клавіш, зразки коду, кольори та багато іншого, що дозволяє їм залишатися продуктивними. Середовище програмування – це набір програмних засобів, що включає в себе текстовий редактор, компілятор і відладчик.

Для розробника часто саме дрібниці мають вирішальне значення - вміння додавати код одним натисканням клавіші, можливість налагоджувати клієнтський і серверний код у проекті або завершення коду з урахуванням мовних особливостей. Успіх Visual Studio Code красномовно свідчить про його можливості та функціональність. Він був офіційно випущений для відкритого перегляду у квітні 2016 року і швидко став одним з найкращих редакторів за популярністю, конкуруючи з Sublime Text, Atom та UltraEdit за перше місце [39].

Visual Studio Code є кросплатформним редактором. У цьому випадку кросплатформеність означає, що версія доступна для запуску на Windows (7, 8 і 10), macOS і Linux. Процес інсталяції для кожної з них схожий, а відправна точка у всіх випадках однакова: <https://code.visualstudio.com/Download>.

Visual Studio Code є 32-розрядним додатком, незалежно від того, працює він у 64-розрядній версії Windows чи ні. Код Visual Studio базується на фреймворку під назвою Electron. Electron був розроблений для створення десктопних додатків з використанням інтерфейсних і бекенд-компонентів вебдодатків. На стороні інтерфейсу використовуються JavaScript, HTML і CSS для створення функціональності користувацького інтерфейсу. Інтерфейс рендериться за допомогою Chromium. На бекенді Node.js використовується для обслуговування компонента користувацького інтерфейсу та надання будь-якої додаткової функціональності.

Така комбінація технологій дозволяє йому працювати на трьох основних платформах: Windows, macOS та Linux.

Головна сторінка VsCode має наступний вигляд (рис. 2.7):

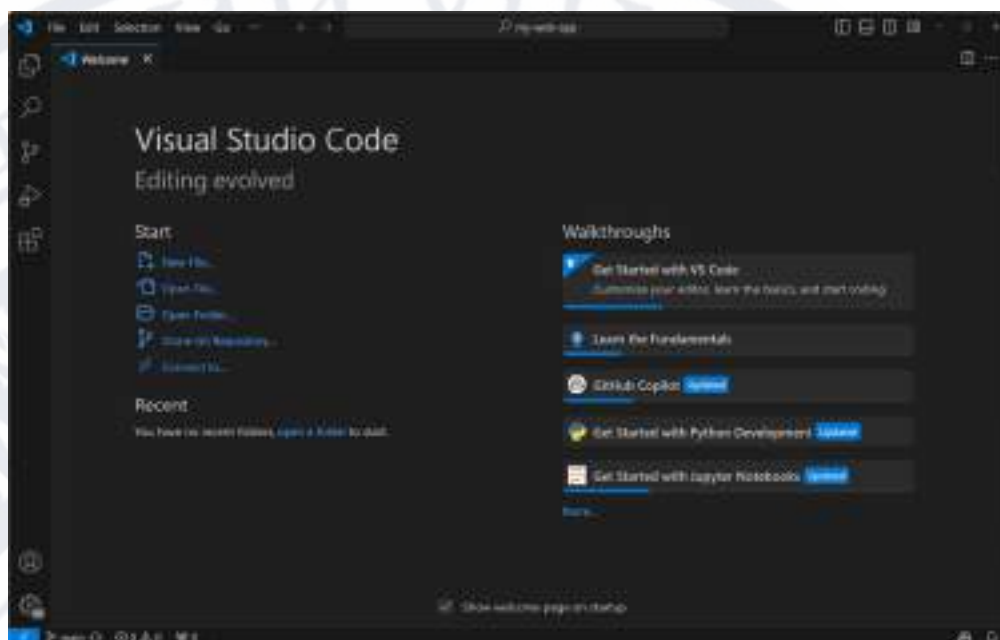


Рисунок 2.7 - Головна/Welcome page VsCode

Visual Studio Code (VSCode) – це популярний редактор коду, який пропонує безліч гарячих клавіш для прискорення та полегшення роботи програмістів. На рис. 2.8 показано основні стандартні гарячі клавіші для платформи Windows [40]:

Visual Studio Code		Ctrl+M		File management	
Keyboard shortcuts for Windows		Toggle Tab moves focus			
General		Search and replace			
Ctrl+Shift+P, F1	Show Command Palette	Ctrl+F	Find	Ctrl+N	New File
Ctrl+P	Quick Open, Go to File...	Ctrl+H	Replace	Ctrl+O	Open File...
Ctrl+Shift+N	New window/instance	F3 / Shift+F3	Find next/previous	Ctrl+S	Save
Ctrl+Shift+W	Close window/instance	Alt+Enter	Select all occurrences of Find match	Ctrl+Shift+S	Save As...
Ctrl+,	User Settings	Ctrl+D	Add selection to next Find match	Ctrl+K S	Save All
Ctrl+K Ctrl+S	Keyboard Shortcuts	Ctrl+K Ctrl+D	Move last selection to next Find match	Ctrl+F4	Close
Basic editing		Alt+C / R / W	Toggle case-sensitive / regex / whole word	Ctrl+K Ctrl+W	Close All
Ctrl+X	Cut line (empty selection)	Multi-cursor and selection		Ctrl+Shift+W	Reopen closed editor
Ctrl+C	Copy line (empty selection)	Alt+Click	Insert cursor	Ctrl+K Enter	Keep preview mode editor open
Alt+ ↑ / ↓	Move line up/down	Ctrl+Click	Insert cursor above / below	Ctrl+Tab	Open next
Shift+Alt+ ↑ / ↓	Copy line up/down	Ctrl+Alt+ ↑ / ↓	Undo last cursor operation	Ctrl+Shift+Tab	Open previous
Ctrl+Shift+K	Delete line	Ctrl+U	Undo last cursor operation	Ctrl+K P	Copy path of active file
Ctrl+Enter	Insert line below	Shift+Alt+I	Insert cursor at end of each line selected	Ctrl+K R	Reveal active file in Explorer
Ctrl+Shift+Enter	Insert line above	Ctrl+L	Select current line	Ctrl+K O	Show active file in new window/instance
Ctrl+Shift+\	Jump to matching bracket	Ctrl+Shift+L	Select all occurrences of current selection	Display	
Ctrl+]/[	Indent/outdent line	Ctrl+F2	Select all occurrences of current word	F11	Toggle full screen
Home / End	Go to beginning/end of line	Shift+Alt+↔	Expand selection	Shift+Alt+O	Toggle editor layout (horizontal/vertical)
Ctrl+Home	Go to beginning of file	Shift+Alt+↔ (drag mouse)	Shrink selection	Ctrl+= / -	Zoom in/out
Ctrl+End	Go to end of file	Shift+Alt+ (arrow key)	Column (box) selection	Ctrl+B	Toggle Sidebar visibility
		Ctrl+Shift+Alt+PgUp/PgDn	Column (box) selection page up/down	Ctrl+Shift+E	Show Explorer / Toggle focus
				Ctrl+Shift+F	Show Search
				Ctrl+Shift+G	Show Source Control
				Ctrl+Shift+D	Show Debug
				Ctrl+Shift+X	Show Extensions
				Ctrl+Shift+H	Replace in files

Рисунок 2.8 - Гарячі клавіші VsCode. Платформа - Windows.

Однією з ключових переваг VSCode є велика кількість доступних розширень (Extensions), які можуть покращити роботу з різними мовами програмування, підсвічування синтаксису, інструментами для розробки тощо.

Розширення доступні в офіційному магазині розширень VSCode, звідки їх можна легко встановлювати та керувати ними.

Розглянемо інструмент для дизайну Figma. Для дизайну є багато інструментів, які допомагають створювати різноманітні види візуальних проєктів. Для різних дизайнерських задач є відповідні інструменти рішень.

Незалежно від того, чи робите ви перші кроки в дизайні, чи вже маєте певний досвід у цій сфері, обраний вами інструмент суттєво впливає на ваш дизайнерський процес і творче самовираження. Йдеться не лише про знання всіх функцій для професійного використання, а й про відповідність інструмента, усвідомлення переваг, які він пропонує.

Figma має багатокористувацьку функціональність, яка дозволяє співпрацювати в реальному часі, а потужна функція виділяє Figma серед інших інструментів, адже вона покращує не лише роботу над дизайном, але й сам процес командної співпраці та сприяє безперебійній роботі між дизайнерами. Це, серед іншого, відрізняє її від конкуруючих продуктів. Figma - один з найдосконаліших інструментів для проектування, в якому постійно з'являються нові функції та вдосконалення. Ця прихильність до розвитку зробила Figma більше, ніж просто інструментом, вона стала незамінним помічником для багатьох дизайнерів [41].

Figma можна як завантажити на свій персональний комп'ютер, так і користуватися у звичайному веббраузері, де вона працює настількиж добре. Що дозволяє використовувати її на будь-якій операційній системі: Windows, Mac, Linux або навіть ChromeOS. Оскільки, Figma базується на хмарних технологіях, необхідне з'єднання до інтернету, щоб відбувалося автоматичне збереження змін у вашому файлі. Однак, якщо інтернет-з'єднання було втрачено, а робота над дизайном продовжувалася, Figma збереже кожне редагування у локальному кеші. Отже, як тільки ви повернетеся до мережі, всі зміни у вашому файлі будуть негайно синхронізовані.

Figma охоплює майже все, що потрібно для створення складних інтерфейсів, від мозкового штурму та фреймворку до створення прототипів та спільного використання ресурсів. Figma також виходить за рамки дизайнерських аспектів

створення продукту і генерує CSS-код для iOS та Android для спрощення роботи розробників.

Figma - це не лише додаток для дизайну, але й платформа спільноти для обміну ідеями та рішеннями. Дизайнери з усього світу використовують Figma не лише для розробки інтерфейсів, але й для створення векторних ілюстрацій, графічного дизайну для цифрових медіа та тімбілдингу.

Кожного разу при вході до Figma, перше, що ви бачитимете - це браузер файлів (рис. 2.9). Це ваш особистий простір з величезною кількістю інформації та функцій.

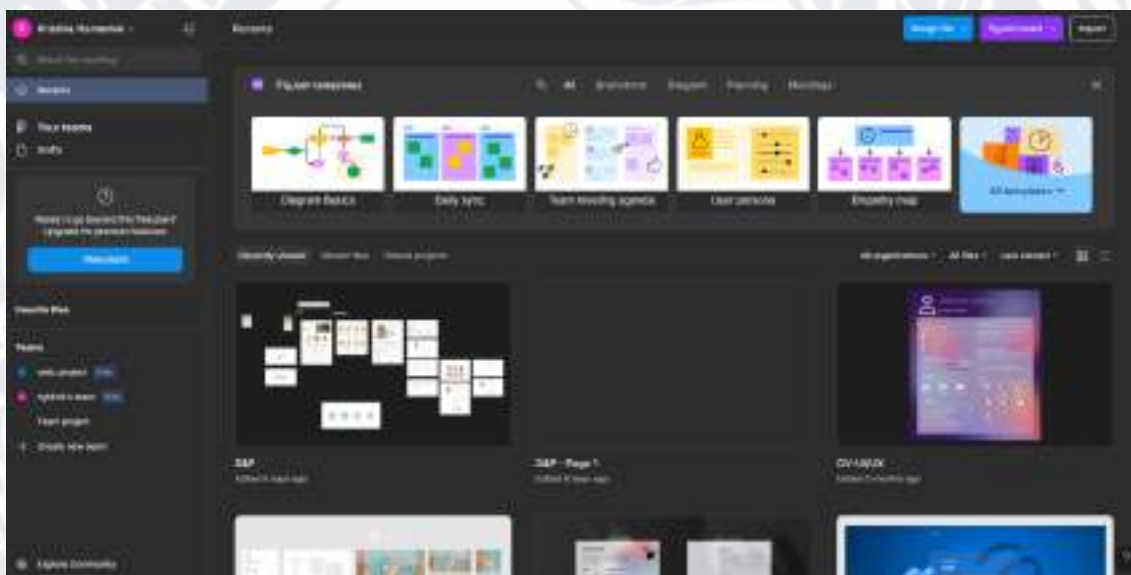


Рисунок 2.9 - Головна сторінка Figma

Створивши новий проєкт на Figma, ми побачимо горизонтальну панель інструментів вгорі (Horizontal toolbar), панель шарів ліворуч (Layers panel), панель оформлення праворуч (Design panel), а посередині - чисте полотно (рис. 2.10) [42].

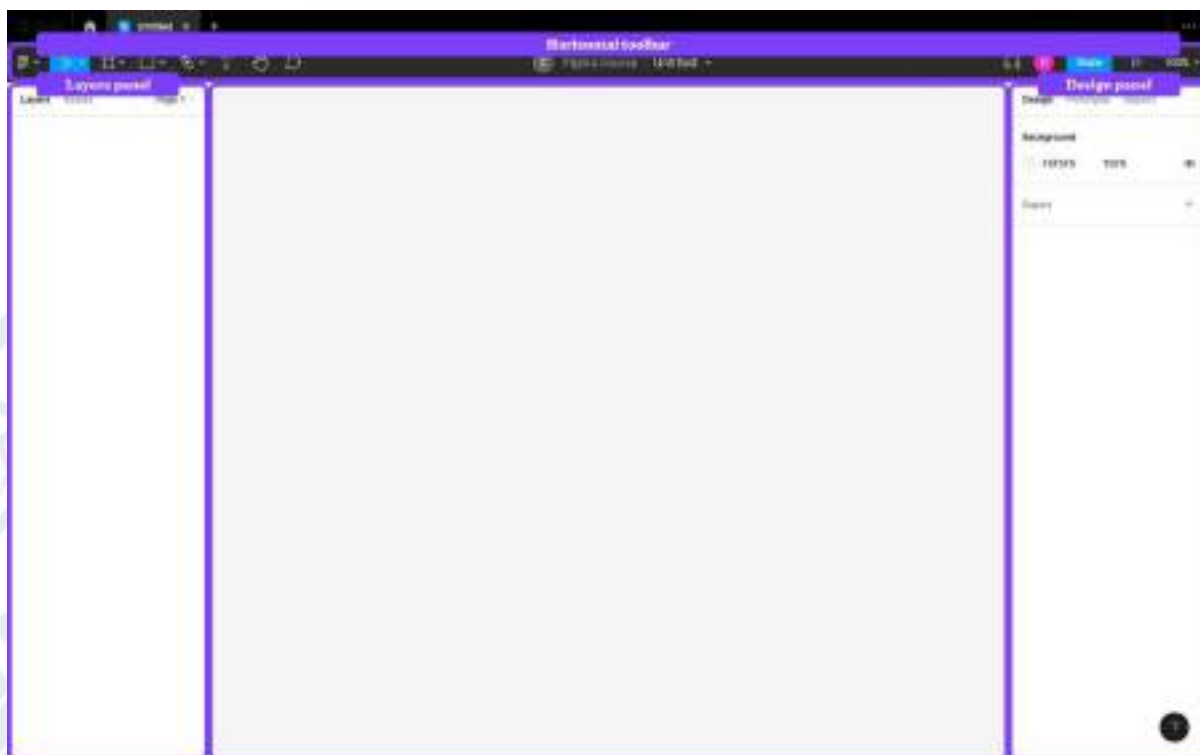


Рисунок 2.10 - Головна сторінка процесу роботи в Figma

РОЗДІЛ 3

РЕАЛІЗАЦІЯ ВЕБСЕРВІСУ ВІЙСЬКОВОГО СПОРЯДЖЕННЯ

3.2 Розробка дизайну вебсервісу військового спорядження

Сучасні вебсервіси - це не лише засіб представлення інформації, але й потужний інструмент для залучення аудиторії, впливу на користувачів та досягнення бізнес-цілей. Однак, щоб стати ефективним інструментом, вебсервіс повинен мати не лише контент, але й привабливий, орієнтований на користувача дизайн.

Колір завжди відіграє важливу роль у дизайні будь-чого. Правильно підібрані кольори можуть визначати візуальний стиль сервісу, впливати на емоційну реакцію користувачів і навіть підкреслювати фірмовий стиль компанії.

Сприйняття кольору та його значення в культурі має глибоке коріння і варіюється від однієї частини світу до іншої та від одного історичного періоду до іншого. Колір завжди був важливим засобом комунікації, символіки та самовираження.

Існують загальні ефекти, викликані різними кольорами. Наприклад, червоний: підвищує частоту серцевих скорочень і артеріальний тиск, стимулює енергію та витривалість, викликає сильні емоції. Синій – знижує частоту серцевих скорочень і кров'яний тиск, сприяє спокою та розслабленню, зменшує стрес і тривогу. Зелений - асоціюється з природою та відродженням. Сприяє відчуттю розслаблення і гармонії а також, зменшує напругу і стрес. Жовтий асоціюється з радістю, оптимізмом та енергією. Він піднімає настрій і покращує психічне здоров'я. Фіолетовий – відомий своїми духовними та творчими конотаціями. Він може викликати почуття самоаналізу та глибокого внутрішнього контакту. Помаранчевий асоціюється з енергією, теплом і комфортом. Він підвищує активність і стимулює почуття оптимізму. Білий символізує чистоту, простоту та інновації. Він створює відчуття свіжості та відкритості. Чорний асоціюється з серйозністю, таємничістю і владою. Він може викликати відчуття глибини і

таємничості. Сірий сприятливо впливає на людину, дарує відчуття спокою і захищеності, знімає напругу, розслабляє.

За основу був обраний білий колір (HEX: #FFFFFF). Далі обрано два відтінки зеленого (HEX: #99BC85 та HEX: #4F6F52) у поєднанні з відтінками сірого, найчастіше буде вживатися (HEX: #E9EBED) та темніший сірий для тексту (рис. 3.1).



Рисунок 3.1 - Основні обрані кольори для сайту

Не менш важливим є вибір правильного шрифту для сайту – це також частина того, як він виглядає і сприймається користувачем. Шрифти можуть викликати певні асоціації та емоції, тому важливо, щоб вони були підібрані відповідно до характеристик бренду та цілей вебсайту.

Для основного шрифту на сайті був обраний - Poiret One (Посилання на шрифт: <https://fonts.google.com/specimen/Poiret+One?query=poiret+one>). А також для створення логотипу – Cardo (Посилання на шрифт: <https://fonts.google.com/specimen/Cardo?query=Cardo>).

Було розроблено наступний логотип магазину (рис. 3.2):



Рисунок 3.2 - Логотип онлайн-магазину

Логотип складається з назви "Defense & Power" та зображення радару, поміщеного замість літери "o" в слові "Power". Даний шрифт надає йому строгий

та офіційний вигляд. Логотип створює асоціації зі сферою оборони, безпеки та військової техніки. Також, у нижній частині додано напис з типом магазину.

Навігаційна панель сайту має наступний вигляд для користувачів, які не здійснили авторизацію на сайт (рис. 3.3), а також для тих, хто створив та увійшов у свій обліковий запис (рис. 3.4):



Рисунок 3.3 - Навігаційна панель для не авторизованих користувачів



Рисунок 3.4 - Навігаційна панель для авторизованих користувачів

Головна сторінка сайту (рис. 3.5) має чітку та просту структуру з акцентом на зображенні та відповідає типовому оформленню інтернет-магазинів. Поруч з фото розміщений заголовок та текст, що закликають відвідувачів скористатися широким асортиментом якісних військових товарів для забезпечення безпеки та комфорту. Є також дві кнопки - "Замовити зараз" та "Дізнатися більше".

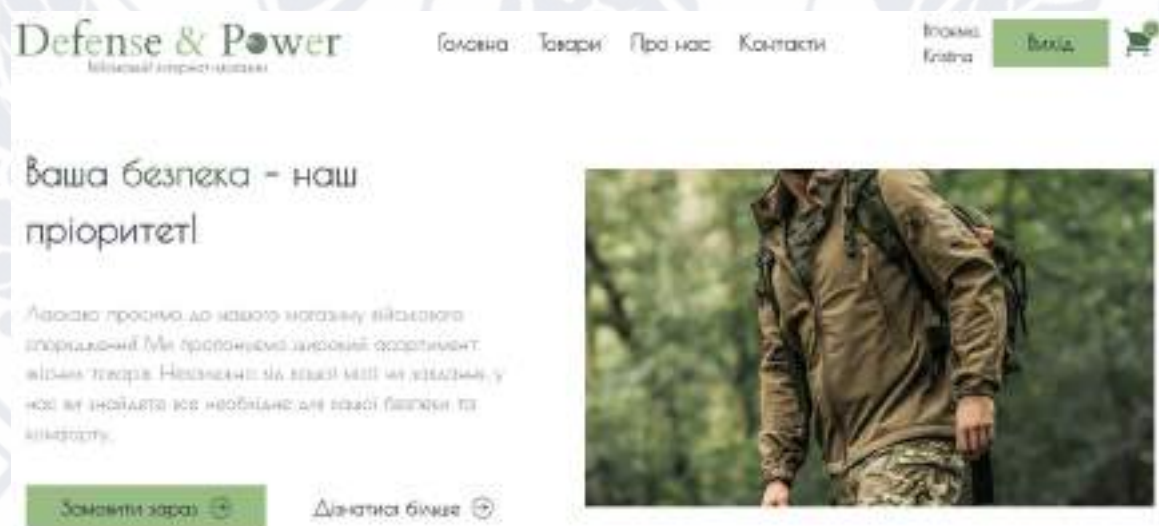


Рисунок 3.5 - Головна сторінка сайту

При прокрутці головної сторінки розташована секція з бестселерами магазину (рис. 3.6).



Рисунок 3.6 - Секція на головній сторінці з бестселерами

Нижче розташована інформація про магазин (рис. 3.7), та контакти (рис. 3.8), до цих розділів можна швидко потрапити натиснувши відповідні посилання «Про нас» та «Контакти» на навігаційній панелі.

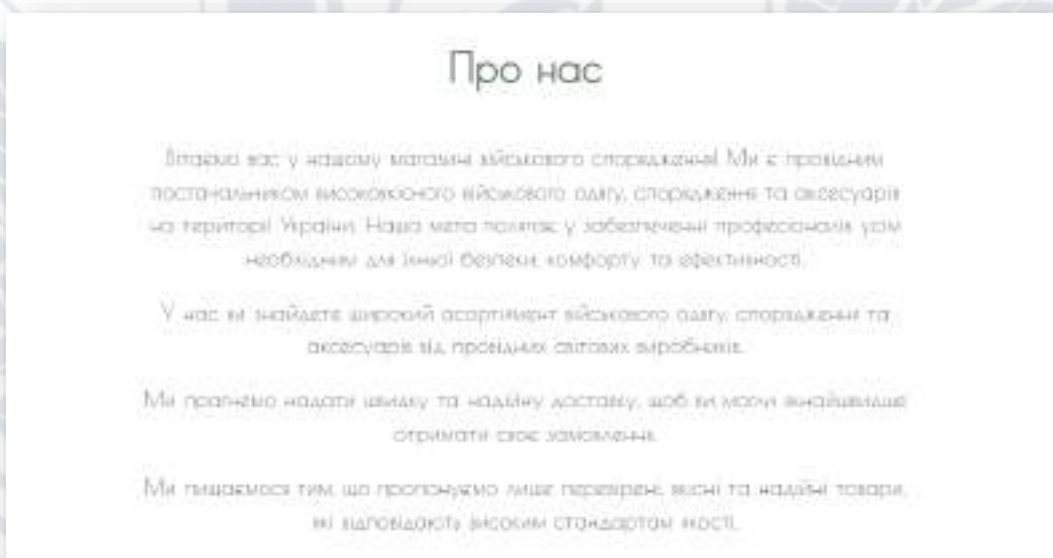


Рисунок 3.7 - Секція з інформацією про магазин



Рисунок 3.8 - Секція контактів магазину

У кінці кожної вебсторінки магазину знаходиться футер (рис. 3.9).



Рисунок 3.9 - Футер вебсайту

Перейшовши на вкладку «Товари» ми бачимо список наявних товарів у магазині, також тут знаходиться поле пошуку та сортування товарів за категоріями (рис. 3.10).

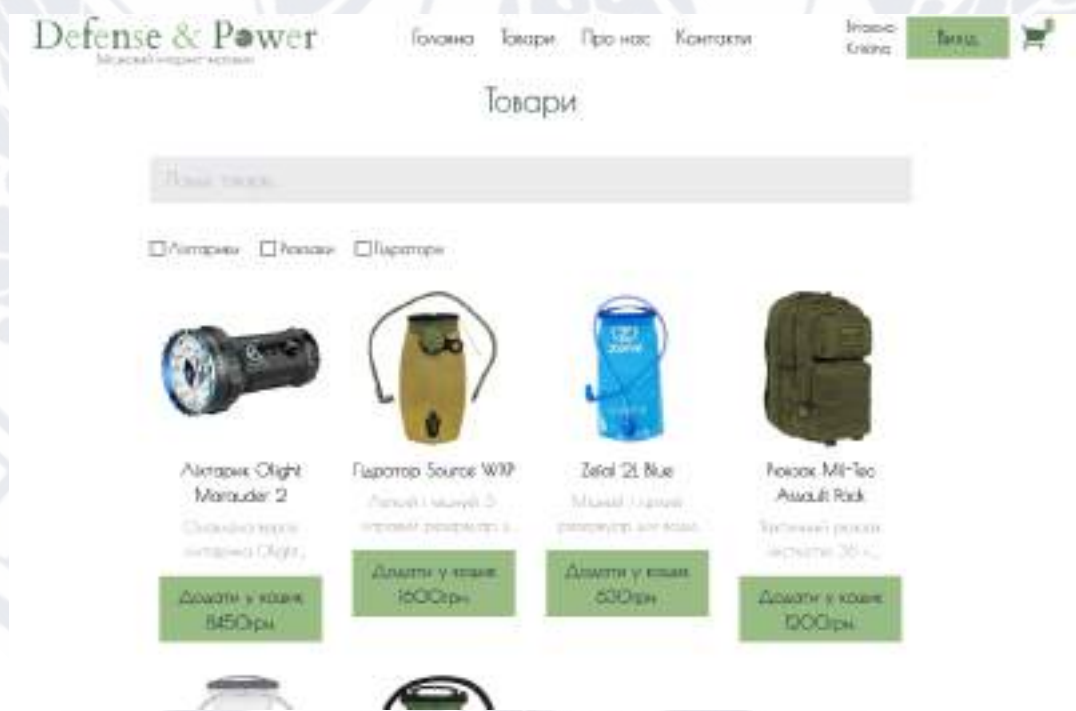


Рисунок 3.10 - Вкладка товари

Defense & Power

Блогер Товары Про нас Контакты

Вход Регистрация

Профіль Замовлення

Результати

Ім'я та Прізвище: test2

E-mail: test2@example.com

Номер телефону: Номер телефону

Почтовий ящик: Місто: Почтовий ящик Миколаїв

Адреса: Адреса

Зберегти

Рисунок 3.12 - Сторінка редагування профілю користувача

У вкладці «Замовлення» користувач бачить інформацію про свої здійсненні покупки (рис. 3.13).

Профіль Замовлення

Замовлено	test2@example.com	0800-00-00-0000	Показати
Не замовлено	test2@example.com	0800-00-00-0000	Показати

Рисунок 3.13 - Вкладка "Замовлення"

Для користувачів, які є адміністраторами тут знаходяться три додаткових вкладки: «Категорії», «Товари» та «Користувачі». Тобто це наша адміністративна панель (рис. 3.14).

Профіль Категорії Товари Користувачі Замовлення

Рисунок 3.14 - Адміністративна панель

У вкладці «Категорії» відбувається створення та редагування наявних категорій (рис. 3.15).

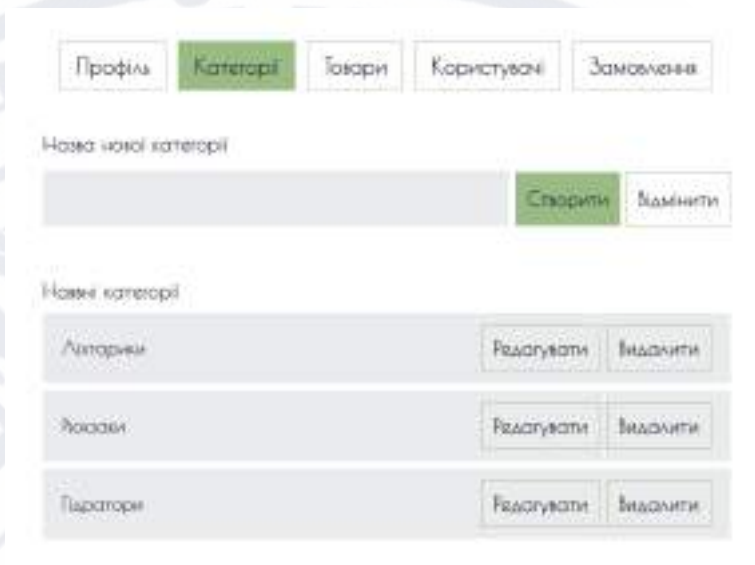


Рисунок 3.15 - Створення та редагування категорій

У вкладці «Товари» відбувається створення та редагування наявних товарів (Рис. 3.16, 3.17, 3.18, 3.19, 3.20).

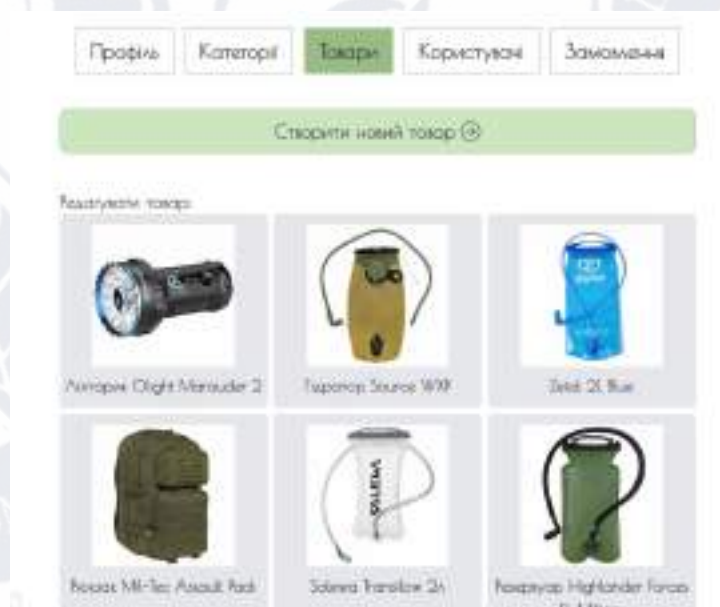


Рисунок 3.16 – Вкладка адміністратора «Товари»

Рисунок 3.17 - Сторінка створення нового товару

Рисунок 3.18 - Додання розміру та додаткової вартості товару

Рисунок 3.19 - Додавання додаткових фото

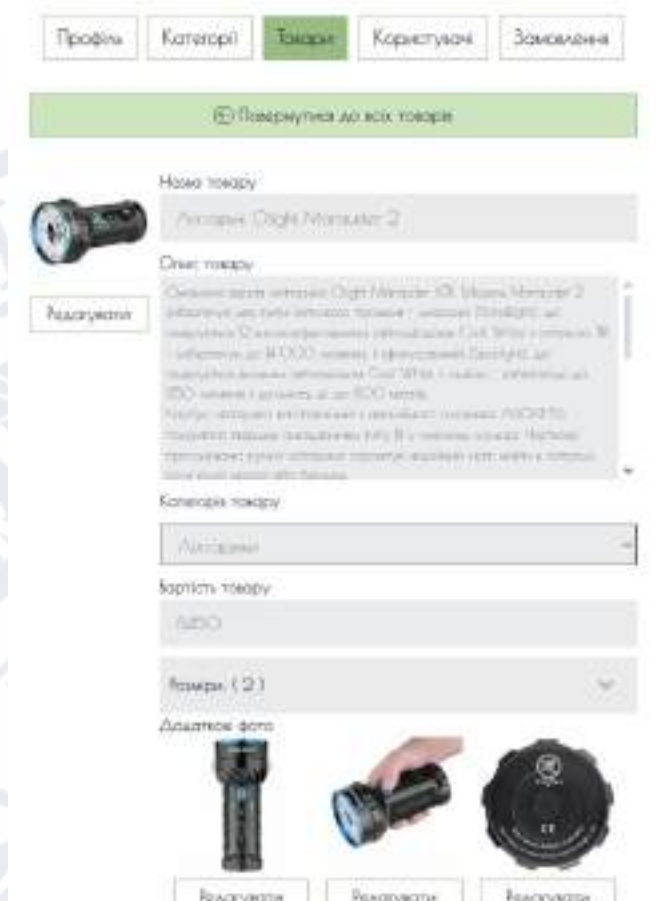


Рисунок 3.20 - Редагування наявного товару

На вкладці «Користувачі» адміністратори бачать усіх користувачів та можуть редагувати їхню інформацію за необхідністю, а також надавати їм права адміністратора (рис. 3.21, 3.22).

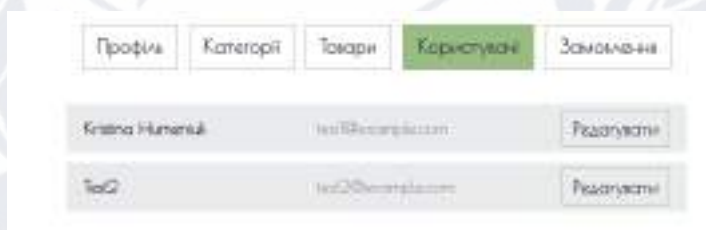


Рисунок 3.21 - Вкладка "Користувачі"

Рисунок 3.22 - Сторінка адміністратора для реагування інформації користувача

При натисненні на іконку кошика на навігаційній панелі, ми переходимо до нашого кошика та оформлення замовлення (рис. 3.23).

Рисунок 3.23 - Сторінка кошика та оформлення замовлення

Сторінка входу та реєстрації мають простий вигляд (рис. 3.24, 3.25).



Рисунок 3.24 - Сторінка авторизації

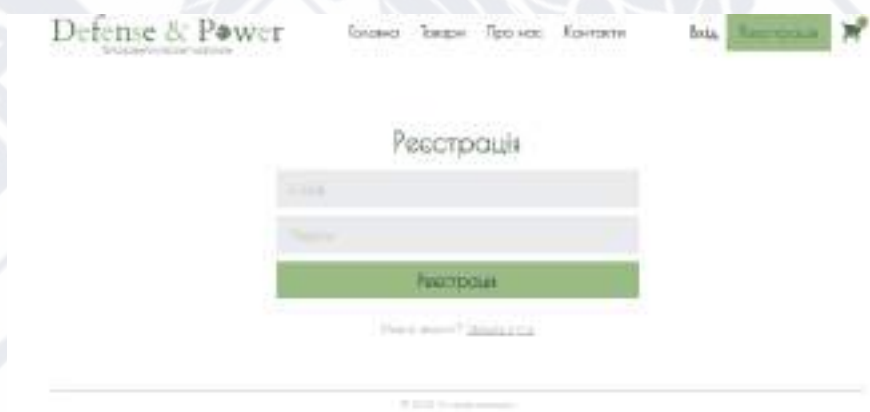


Рисунок 3.25 - Сторінка реєстрації

На цьому етапі розробка дизайну закінчена. В ході цього розділу ми розглянули ключові аспекти дизайну, такі як вибір кольору та шрифту, які займають пріоритетну роль у створенні практичного в функціонуванні та естетичного дизайну. Загальний дизайн відображає характер та цілі проекту, створюючи приємне та зручне середовище для користувачів.

3.3 Програмна реалізація вебсервісу військового спорядження

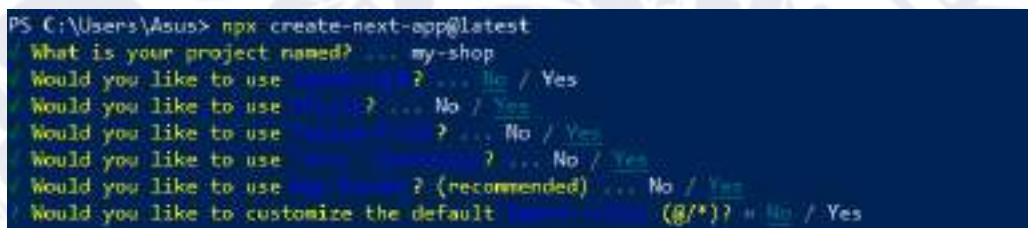
Розробка починається зі створення проекту `nuxt.js`, але для того щоб його створити, для початку треба завантажити та встановити останню наявну версію `node.js`. Це можна зробити на офіційному сайті за посиланням: <https://nodejs.org/en/>. Після проходження стандартних кроків встановлення, відкриваємо Windows PowerShell та перевіряємо чи встановлення пройшло успішно ввівши команду, яка має повернути нам встановлену версію `node.js` (рис. 3.26):



```
PS C:\Users\Asus> node -v
v20.9.0
```

Рисунок 3.26 - Перевірка версії node.js

Перейдемо безпосередньо до створення нашого проекту. Ми вводимо наступну команду `npx create-next-app@latest`. Далі вписуємо ім'я нашого проекту та натискаємо Enter та обираємо необхідні нам параметри рис. 3.27. Після підтвердження створюється проект у відповідному місці.



```
PS C:\Users\Asus> npx create-next-app@latest
What is your project named? ... my-shop
Would you like to use TypeScript? ... No / Yes
Would you like to use ESLint? ... No / Yes
Would you like to use Tailwind CSS? ... No / Yes
Would you like to use a CSS framework? ... No / Yes
Would you like to use a CSS framework? ... (recommended) ... No / Yes
Would you like to customize the default import alias? (g/*)? ... No / Yes
```

Рисунок 3.27 - Створення проекту Next.js

Відкриємо створений проект у обраному середовищі програмування, тобто у VSCode. Структура проекту матиме наступний вигляд (рис. 3.28):

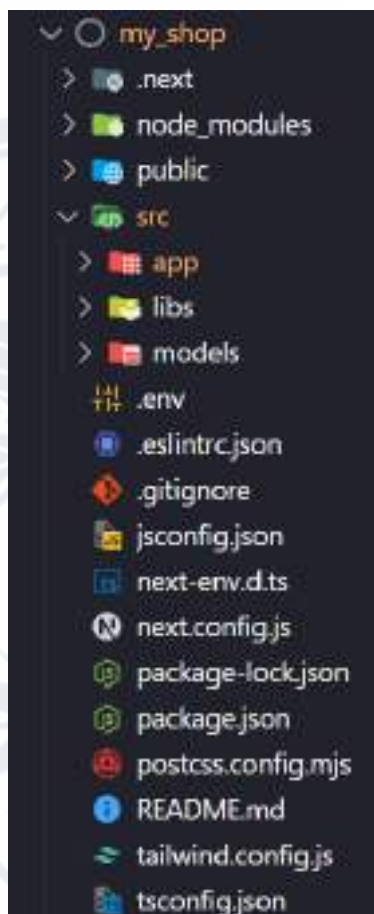


Рисунок 3.28 - Структура створеного проекту

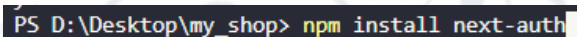
Розглянемо основні папки та файли:

- `node_modules` - тут зберігаються всі встановлені npm-пакети та залежності проекту;
- `public` - статичні файли, такі як зображення, CSS, які доступні безпосередньо через URL;
- `src` - папка із вихідним кодом програми;
- `app` - тут зберігаються React-компоненти та сторінки додатку;
- `libs` - бібліотечні функції, утиліти;
- `models` - вміщує моделі даних, інтерфейси тощо;
- `.env` - файл з змінними оточення;
- `.eslintrc.json` - конфігурація линтера ESLint;
- `.gitignore` - список файлів та папок, що ігноруються Git;
- `jsconfig.json` - конфігурація для JavaScript в середовищі розробки;

- next-env.d.ts - типізація для Next.js;
- next.config.js - файл конфігурації Next.js;
- package-lock.json, package.json - відомості про залежності проекту;
- postcss.config.mjs - конфігурація PostCSS;
- README.md - файл з інструкціями та описом проекту;
- tailwind.config.js - конфігурація Tailwind CSS;
- tsconfig.json - конфігурація TypeScript.

Ця структура забезпечує організацію коду призначена для побудови масштабованих React-додатків. Для запуску режиму розробки використовуємо команду *npm run dev*.

Для розробки авторизації необхідно у самому проекті підвантажити пакунок next-auth (рис. 3.29).



```
PS D:\Desktop\my_shop> npm install next-auth
```

Рисунок 3.29 - Встановлення next-auth

Щоб додати NextAuth.js до проекту, потрібно створити файл з назвою [...nextauth].js в app/api/auth. Він містить динамічний обробник маршрутів для NextAuth.js, який також буде містити всі глобальні конфігурації NextAuth.js. (рис. 3.30).

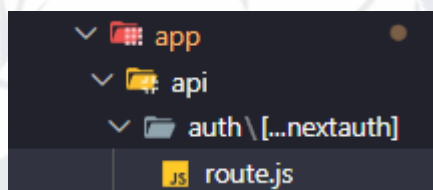


Рисунок 3.30 - Файл для NextAuth.js

Далі за допомогою MongoDB потрібно створити нашу базу даних, до якої ми будемо від'єднуватися. Переходимо безпосередньо до її створення, обираємо безкоштовну версію та натискаємо створити (рис. 3.31).

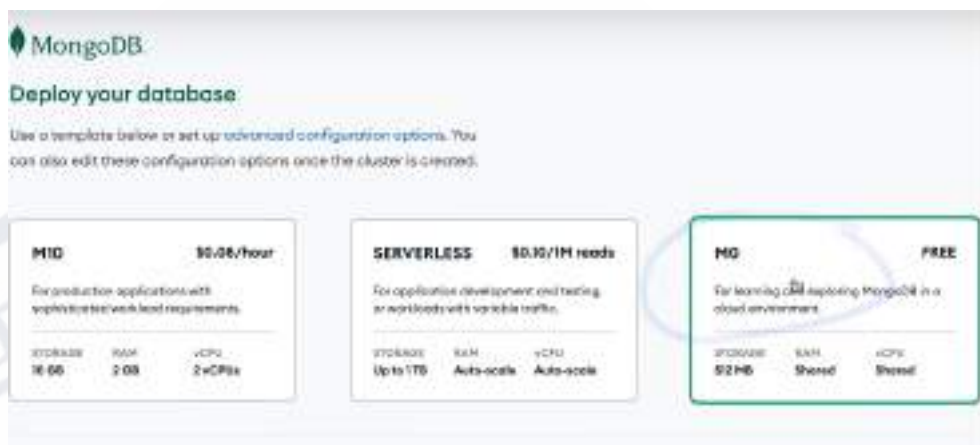


Рисунок 3.31 - Створення бази даних. Крок 1.

Далі вводимо ім'я користувача, пароль копіюємо, будемо використовувати його в проєкті, він буде зберігатися у файлі .env (рис. 3.32).

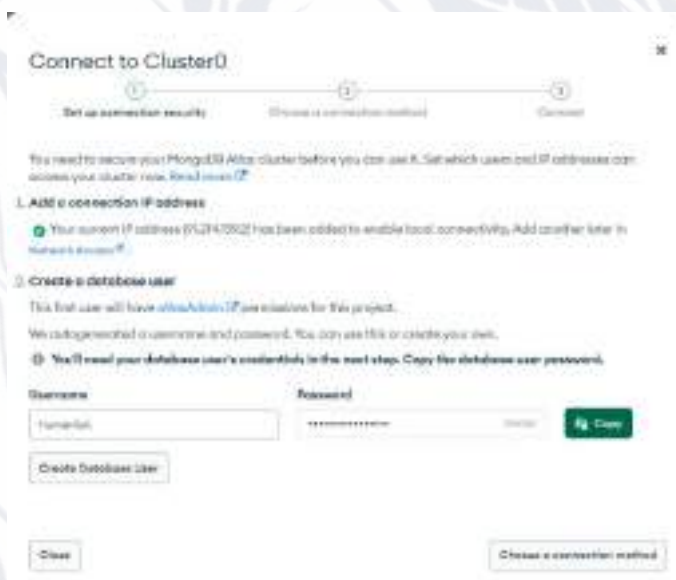


Рисунок 3.32 - Створення бази даних. Крок 2.

Далі обираємо метод підключення, у даному випадку це MongoDB for VS Code знову ж таки зберігаємо пароль та натискаємо готово (рис. 3.33).



Рисунок 3.33 - Створення бази даних. Крок 3.

Якщо все виконано вірно у списку буде видно створену БД (рис. 3.34).



Рисунок 3.34 - Створена БД

Тепер під'єднуємо БД для авторизації та реєстрації. Файл з маршрутом для аутентифікації матиме приблизно наступний код (рис. 3.35):


```

export const authOptions = {
  secret: process.env.SECRET,

  providers: [

    CredentialsProvider({
      name: 'Credentials',
      id: 'credentials',
      credentials: {
        username: { label: "Email", type: "email", placeholder: "test@example.com" },
        password: { label: "Password", type: "password" },
      },
      async authorize(credentials, req) {
        const email = credentials?.email;
        const password = credentials?.password;

        mongoose.connect(process.env.MONGO_URL);
        const user = await User.findOne({email});
        const passwordOk = user && bcrypt.compareSync(password, user.password);

        if (passwordOk) {
          return user;
        }

        return null;
      }
    })
  ]
}

```

Рисунок 3.35 - Auth route

Функція для реєстрації показана на рис. 3.36. Цей код є обробником запиту POST для створення нового користувача у системі.

```

export async function POST(req){
  const body = await req.json();
  mongoose.connect(process.env.MONGO_URL);
  const pass = body.password;
  if (!pass?.length || pass.length < 5) {
    new Error('password must be at least 5 characters');
  }

  const notHashedPassword = pass;
  const salt = bcrypt.genSaltSync(10);
  body.password = bcrypt.hashSync(notHashedPassword, salt);

  const createdUser = await User.create(body)
  return Response.json(createdUser);
}

```

Рисунок 3.36 - Функція для створення нового користувача

При наступному написанні коду, дані будуть синхронізуватися з базою даних, які можна переглянути за допомогою вкладки Collections у БД, де буде видно створені «таблички» та дані в них (рис. 3.37).



Рисунок 3.37 – Таблички та дані в БД

На рис. 3.38 наведено схему бази даних.

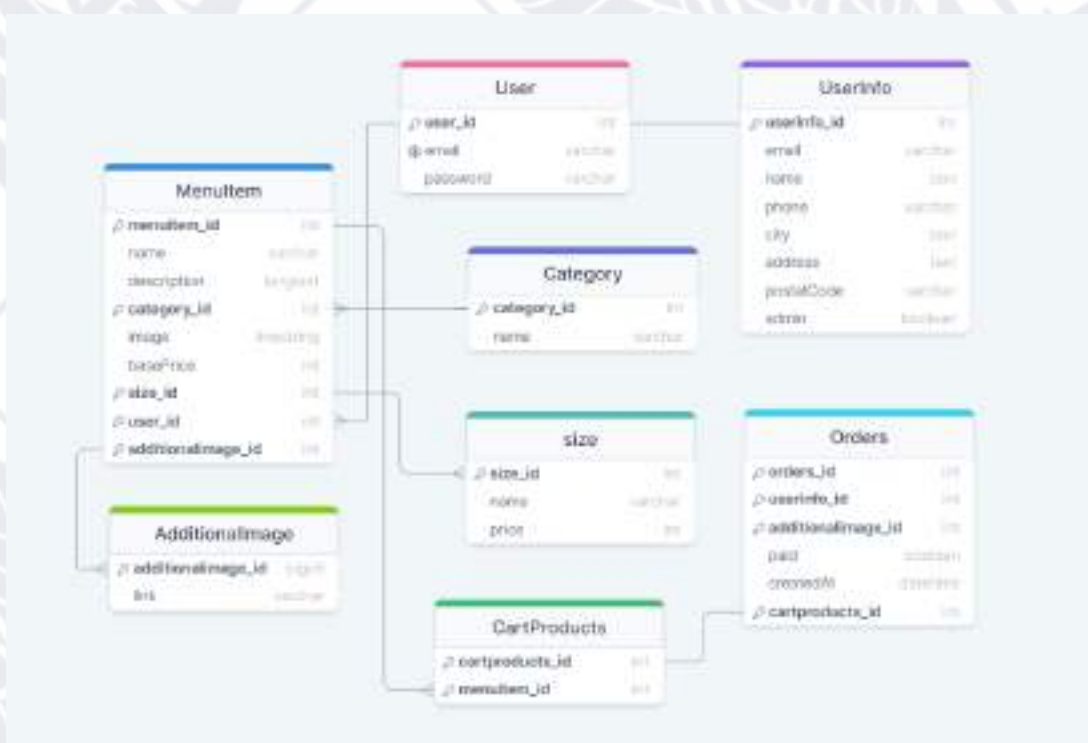


Рисунок 3.38 – Схема БД

У ході розробки використовувалася бібліотека react-hot-toast - це бібліотека для React, яка дозволяє легко відображати спливаючі повідомлення (toast) у вебдодатку. Вона пропонує простий та зручний інтерфейс для відображення повідомлень про успіх, помилку чи інші повідомлення користувачеві.

На рис. 3.39 показано використання даної бібліотеки в одній з функцій (тут безпосередньо створення та оновлення категорії).


```

async function handleCategorySubmit(e) {
  e.preventDefault();
  const creationPromise = new Promise(async(resolve, reject) => {
    const data = {name: categoryName};
    if (editedCategory) {
      data._id = editedCategory._id;
    }
    const response = await fetch('/api/categories', {
      method: editedCategory ? 'PUT' : 'POST',
      headers: {'Content-Type': 'application/json'},
      body: JSON.stringify(data),
    });
    setCategoryName('');
    fetchCategories();
    setEditedCategory(null);
    if(response.ok) resolve();
    else reject();
  });
  await toast.promise(creationPromise, {
    loading: 'editedCategory'
      ? 'Визначення категорії...'
      : 'Створення нової категорії...',
    success: 'editedCategory' ? 'Категорія оновлена!' : 'Категорія створена!',
    error: 'Виникла помилка...'
  });
}

```

Рисунок 3.39 - Приклад використання бібліотеки react-hot-toast

При створенні чи оновленні категорій видно спливаючі відповідні повідомлення успіху або помилки у верхній частині вебсервісу (рис. 3.40).



Рисунок 3.40 - Результат роботи бібліотеки

На рис. 3.41 наведено структуру розробленого вебсервісу.



Рисунок 3.41 - Структура вебсервісу

ВИСНОВКИ

У бакалаврській роботі проведено дослідження та розробку вебсервісу для онлайн-замовлень військового спорядження. Основною метою було створення ефективного та корисного інструменту, який би дозволив військовим, без зайвого клопоту та витрат часу, придбати необхідне їм спорядження. Для досягнення цієї мети були вирішені наступні завдання:

- проведено аналіз потреб військових у спеціальному спорядженні;
- вивчено існуючі вебсервіси для замовлення спорядження та проведено їх аналіз;
- розроблено дизайн вебсервісу для замовлення військового спорядження;
- створено базу даних для даного вебсервісу;
- реалізовано функціонал та проведено тестування даного вебсервісу.

Розроблений вебсервіс для онлайн замовлень військового спорядження забезпечує зручний та інтуїтивно зрозумілий інтерфейс, що спрощує процес замовлення спорядження та заощаджує час користувачів. Створений програмний продукт повністю задовольняє поставлені цілі та завдання.

У процесі розробки вебсервісу для онлайн замовлень військового спорядження було використано ряд потужних інструментів, які сприяли успішному втіленню поставлених завдань та досягненню мети проекту.

Для клієнтської або фронтенд-розробки вебсервісу обрано бібліотеку React. Було використано CSS-фреймворк Tailwind, який дозволяє швидко та легко створювати дизайни для стилізації та редагування користувацького інтерфейсу. Для серверної частини додатку було обрано Next.js, адже Next.js забезпечує високопродуктивні, масштабовані вебдодатки. Для зберігання даних було обрано MongoDB, оскільки це надійна та гнучка NoSQL-база даних, яка чудово інтегрується з JavaScript-додатками. Для роботи над дизайном та створення прототипів було використано інструмент Figma, що дозволив створити простий та

зручний інтерфейс. Та використовувалося, усім відоме, середовище програмування VSCode.

Використання таких інструментів, як React, Tailwind CSS, Next.js, MongoDB та Figma, дозволило забезпечити високу якість та ефективність розробки вебсервісу для онлайн замовлень військового спорядження, що відповідає потребам користувачів та сучасним стандартам веброзробки.

СПИСОК ЛІТЕРАТУРИ

1. Bistricher Rachel What is a website? Definition + examples. 2023. URL: <https://www.wix.com/blog/what-is-a-website>.
2. Rouse Margaret Website. 2020. URL: <https://www.techopedia.com/definition/5411/website>.
3. Yalanska Marina Web Design: 16 Basic Types of Web Pages. URL: <https://blog.tubikstudio.com/types-of-web-pages/>.
4. Noviantika G. What Is the Difference Between a Blog and a Website: Content, Monetization, and Updates. 2024. URL: <https://www.hostinger.com/tutorials/difference-between-blog-and-website>.
5. Sydney Go What Is an Ecommerce Website? A Complete Guide for 2024. 2024. URL: <https://www.semrush.com/blog/ecommerce-website/>.
6. Tomasis Rebecca What is an eCommerce website and why do you need one. 2024. URL: <https://www.wix.com/blog/what-is-an-ecommerce-website>.
7. Intuit Mailchimp E-commerce Website. 2021. URL: <https://mailchimp.com/marketing-glossary/ecommerce-website/#:~:text=An%20ecommerce%20website%20is,logistics%2C%20and%20provide%20customer%20service>.
8. Shwake Emily 27 common types of websites (with templates to get you started). 2023. URL: https://www.wix.com/blog/types-of-websites?utm_source=google&utm_medium=cpc&utm_campaign=13708482660^124757113592^search%20-%20dsa&experiment_id=^530755701287^&gad_source=1&gclid=CjwKCAjwrcKxBhBMEiwAIVF8rC1ZDgO307gP0nlU8YIJZMDXNTTHSSAkDQd0RgCViEyNJRt8SN6w8RoCOSAQA_vD_BwE#viewer-7ssgt.
9. Unbounce What is a landing page?. URL: <https://unbounce.com/landing-page-articles/what-is-a-landing-page/#:~:text=In%20digital%20marketing%2C%20a%20landing,similar%20places%20on%20the%20web>.

10. Webflow Team Landing pages versus websites: Key differences & how to use them effectively. 2023. URL: <https://webflow.com/blog/landing-pages-vs-websites>.
11. Yumpu Definition of an educational website. 2023. URL: <https://www.yumpu.com/en/document/view/68021521/definition-of-an-educational-website>.
12. Manning Katie A Guide to the Most Popular Types of Websites. 2023. URL: <https://www.loop-digital.co.uk/popular-types-of-websites/#educational-websites>.
13. Wearecom Corporate website. URL: <https://www.wearecom.fr/en/dictionnaire/corporate-site/#:~:text=Corporate%20website,to%20its%20customers%20or%20users>.
14. RGB Agency What is a corporate website? 2022. URL: <https://rgbweb.studio/blog/what-is-a-corporate-website/>.
15. Rajkotupdates news, Medium What does news websites do? 2023. URL: <https://medium.com/@utcmanik/what-does-news-websites-do-bc4d10abd833>.
16. Julie C. Meloni HTML, CSS and JavaScript. Sams Publishing. 2018. C 2-3.
17. DuRocher David HTML and CSS QuickStart Guide: The Simplified Beginners Guide to Developing a Strong Coding Foundation, Building Responsive Websites, and Mastering the Fundamentals of Modern Web-design. 2021. C. 47-48.
18. Robson Elisabeth Head First HTML and CSS: A Learner's Guide to Creating Standards-Based Web Pages. O'Reilly Media. 2012. C. 17.
19. Ackermann Philip JavaScript: The Comprehensive Guide to Learning Professional JavaScript Programming. Rheinwerk Computing. 2022. C. 64-65.
20. Suehring Steve, Valade Janet PHP, MySQL, JavaScript & HTML5 All-In-One For Dummies. For Dummies. 2013. C. 187-188.
21. Nicholas C. Zakas Understanding JavaScript Promises. Lulu.com. 2022. C. 1-3.

22. Geeksforgeeks JavaScript Tutorial. 2024. URL:
<https://www.geeksforgeeks.org/javascript/>.
23. Bhat Kartik Ultimate Tailwind CSS Handbook. Orange Education Pvt Ltd.
 2023. C. 33.
24. Rappin Noel Modern CSS with Tailwind Flexible Styling Without the Fuss.
 Pragmatic Bookshelf. 2021. C. 10-11.
25. Gerchev Ivaylo Tailwind CSS: Craft Beautiful, Flexible, and Responsive
 Designs. SitePoint. 2022. C. 3-5.
26. Khandelwal Dev Mastering Responsive Web Design with Tailwind CSS.
 Medium. 2023. URL: <https://slyro.medium.com/mastering-responsive-web-design-with-tailwind-css-8beee9f9f596>.
27. Riva Michele Real-World Next.js: Build scalable, high-performance, and
 modern web applications using Next.js, the React framework for production.
 Packt Publishing. 2022. C. 4-5.
28. Banks Alex Learning React: Modern Patterns for Developing React Apps. O'-
 Reilly Media. 2020. C. 280.
29. Romesh Tharaka Why Choose Next.js - Top 5 Performance Benefits. .Cult.
 2023. URL: <https://cult.honeypot.io/reads/top-nextjs-performance-benefits/>.
30. Barklund Morten, Mardan Azat React Quickly. Manning. 2023. C. 2-3.
31. Osmani Addy Learning JavaScript Design Patterns. O'Reilly Media. 2023.
 C.181-254.
32. Khan Qaifi ReactJS for Jobseekers. BPB Publications. 2023 C. 149-166.
33. Sarrion Eric Master React in 5 Days. Apress. 2023. C. 3-21
34. Dhanani Hardik A Definitive Guide to ReactJS Fundamentals. TatvaSoft. 2023.
 URL: <https://www.etatvasoft.com/blog/reactjs-fundamentals/>.
35. Aleksendrić Marko, Borucki Arek, Domingues Leandro, Hammad Malak Abu,
 Hannouch Elie, Nair Rajesh, Palmer Rachelle Mastering MongoDB 7.0. Packt
 Publishing. 2024. C. 26.
36. Jenkins Justin MongoDB for Jobseekers. BPB Publications. 2023. C. 1-20.
37. Ashok Chaubal Pinakin Mastering Mean Stack. BPB Publications. 2023. C. 7.

38. Prabhu Vignesh Kumar Rajagopal Introduction to MongoDB. Digital Varys.
URL: https://digitalvarys.com/introduction-to-mongodb/#Importance_of_MongoDB_Technology.
39. Johnson Bruce Visual Studio Code. Wiley. 2019. C. 7-8.
40. Visual Studio Code Tips and Tricks. Visual Studio Code. 2024. URL: <https://code.visualstudio.com/docs/getstarted/tips-and-tricks>.
41. Staiano Fabio Designing and Prototyping Interfaces with Figma. Packt Publishing. 2022. C. 4-5.
42. Schwarz Daniel The Designer's Guide to Figma: Master Prototyping, Collaboration, Handoff, and Workflow. SitePoint. 2023. C. 12-13.

ДЕКЛАРАЦІЯ

про дотримання академічної доброчесності

Я, _____

Повністю вказується ПІБ та статус (посада для працівників, освітня (освітньо-наукова) програма – для здобувачів вищої освіти)

що нижче підписалась/підписався, розуміючи та підтримуючи загальнови-
знані засади справедливості, доброчесності та законності,

ЗОБОВ'ЯЗУЮСЬ:

дотримуватися принципів та правил академічної доброчесності, що визна-
чені законодавством України, локальними нормативними актами Донецького на-
ціонального університету імені Василя Стуса, положеннями, правилами, умо-
вами, визначеними іншими суб'єктами, та не допускати їх порушення.

ПІДТВЕРДЖУЮ:

що мені відомі положення статті 42 Закону України «Про освіту»;

що у даній роботі не представляла/представляв чийсь роботи повністю або
частково як свої власні. Там, де я скористалася/скористався працею інших, я зро-
била/зробив відповідні посилання на джерела інформації;

що дана робота не передавалась іншим особам і подається вперше, не пору-
шує авторських та суміжних прав закріплених статтями 21-25 Закону України
«Про авторське право та суміжні права», а дані та інформація не отримувались в
недозволений спосіб.

УСВІДОМЛЮЮ:

що ця робота може бути перевірена університетом на плагіат або інші пору-
шення академічної доброчесності, в тому числі з використанням спеціалізованих
сервісів;

що у разі порушення академічної доброчесності, до мене можуть бути за-
стосовані процедури, передбачені законодавством України та Кодексом академі-
чної доброчесності та корпоративною етикою Донецького національного універси-
тету імені Василя Стуса, іншими локальними нормативними актами універси-
тету, та я можу бути притягнута/притягнутий до академічної відповідальності.

(дата)

(підпис)

**II МІЖНАРОДНА НАУКОВО-
ПРАКТИЧНА КОНФЕРЕНЦІЯ**

**II INTERNATIONAL SCIENTIFIC
AND PRACTICAL CONFERENCE**



**Сучасний стан та основні
пріоритети розвитку економіки**

**Current state and main priorities
of economic development**



**14 квітня 2024 р.
м. Одеса, Україна**

**April 14, 2024
Odesa, Ukraine**

ЗМІСТ
TABLE OF CONTENTS

СЕКЦІЯ 1. ЕКОНОМІКА
SECTION 1. ECONOMICS

Антонюк Д., Коляда О.

Використання штучного інтелекту для зміцнення підприємств
в умовах євроінтеграції: виклики та можливості. 10

Бойко О. В.

Властиві персонал підприємств складові екологічного рюкзак... 14

Вішнівецька К. О., Ширяєва Л. В.

Транснаціональна природа холдингових компаній.. 17

Машенко М. А., Волкодав В. Ю., Швець А. Д.

Ринок криптовалют і відображення окремих
біржових індексів галузей США як фактор
розвитку малого та середнього підприємства..... 20

Негляд А. В.

Етапи оцінки рівня фінансової конкурентоспроможності
підприємств в галузевих структурах..... 25

Онешко С. В.

Показники прибутковості компаній
у сфері інформаційних технологій..... 28

Паляниця В. В.

Strategic development of enterprises
in the conditions of transformations..... 32

Паляниця С. В.

Peculiarities of modern entrepreneurial
activity in the conditions of transformations..... 34

Сергієнко О. А., Рилов Д. Ю., Самусь П. О.

Тенденції розвитку ринку діагностичних пристроїв... 36

Стасовська Г.

Стратегічні аспекти формування інтелектуальних
активів інноваційного підприємництва.. 40

Трипольська Г. С.

Оцінка видатків на запровадження енергоакуюлюючих
потужностей в секторі домогосподарств України..... 44

Chiz, L., Nikolaieva, K.

Implementation of financial control
in the enterprise management system..... 46

Шинькович А. В.

Геополітичний фактор у визначенні
поняття продовольчої безпеки..... 50

Shyriaieva, L., Fedorov, O.

Global experience in implementation of blockchain
technology for the protection of financial information..... 55

СЕКЦІЯ 2. МІЖНАРОДНІ ВІДНОСИНИ**SECTION 2. INTERNATIONAL RELATIONS**

Запущляк В. З., Івашук О. В.

Багатовекторність економічної дипломатії
України в умовах глобальної невизначеності..... 60

Новак О. Є.

Transformation of national agri-food
systems: innovation dynamics..... 63

СЕКЦІЯ 3. ОБЛІК І ОПОДАТКУВАННЯ**SECTION 3. ACCOUNTING AND TAXATION**

Болмат О. В.

Відображення в обліку списання
запасів в умовах військового стану... 68

Демидюк В. О.

Відображення ризиків непогашення дебіторської
заборгованості в інтегрованій звітності.. 71

Сук П. Л.

Обернений метод амортизації необоротних активів
на основі чистого доходу від основної діяльності..... 74

СЕКЦІЯ 4. ФІНАНСИ, БАНКІВСЬКА СПРАВА ТА СТРАХУВАННЯ**SECTION 4. FINANCE, BANKING AND INSURANCE**

Кісельова Е. С.

Розвиток особистого страхування в Україні... 80

Ковернінська Ю. В.

Переваги застосування інформаційних технологій у
фінансовому аналізі діяльності суб'єктів господарювання. 82

Новосад Д. І.

Блокчейн як інноваційний фінансовий інструмент
для забезпечення сталого економічного розвитку..... 85

Шемаєва Л. Г.

Переваги та недоліки боргових методів фінансування
бюджетного дефіциту в умовах війни.....89

СЕКЦІЯ 5. МЕНЕДЖМЕНТ

SECTION 5. MANAGEMENT

Босак А. О., Пугач А. І.

Дослідження теоретичної структури
інтелектуального капіталу..... 93

Казанцева А. Є., Дорошенко Г. О.

Основні підходи до управління
організаційною культурою підприємства.....96

Медяник Ю.Г., Жихарєва В. В.

Ефективні методи управління адаптацією
персоналу в сучасних умовах..... 99

Понедельченко К. В., Пенська І. О.

Застосування інструментарію менеджменту
при технічній підготовці інноваційного проєкту.....103

Равлінко З. П.

Практичні аспекти оцінювання результативності
безпекової діяльності торговельного підприємства. 107

СЕКЦІЯ 6. ПУБЛІЧНЕ УПРАВЛІННЯ ТА АДМІНІСТРУВАННЯ

SECTION 6. PUBLIC MANAGEMENT AND ADMINISTRATION

Вороніна Ю. Є.

Формування громадянської активності молоді.....111

СЕКЦІЯ 7. ТУРИЗМ ТА ГОТЕЛЬНО-РЕСТОРАННА СПРАВА

SECTION 7. TOURISM, HOTEL AND RESTAURANT BUSINESS

Гопанчук Л. М.

Сучасні тенденції гостинності.114

Москвічова О. С.

Функціональний підхід в управлінні
ризиками готельного підприємства... 116

СЕКЦІЯ 8. СІЛЬСЬКЕ ГОСПОДАРСТВО

SECTION 8. AGRICULTURE

Калинка А. К., Лесик О. Б.

Продуктивність бугайців нової популяції буковинського зонального типу м'ясного комолого сименталу худобина на раціонах з високим рівнем енергії з використанням різних сумішей концентрованих в умовах передгір'я Карпат..... 122

СЕКЦІЯ 9. ІНЖЕНЕРІЯ ТА ТЕХНОЛОГІЇ

SECTION 9. ENGINEERING AND TECHNOLOGY

Гуменюк К. В., Січко Т. В.

Адаптивний дизайн веб-сайтів: принципи та методи..... 128

СЕКЦІЯ 10. ПРАВО

SECTION 10. LAW

Кірик А. Ю.

Альтернативні методи вирішення спорів у сфері захисту прав споживачів..... 131

Рибалко В. О.

Щодо недоцільності зупинення дисциплінарного провадження стосовно судді до вирішення кримінальної справи..... 134

СЕКЦІЯ 11. ПЕДАГОГІКА

SECTION 11. PEDAGOGY

Драгунова В. В.

Реалії та пріоритети проектування консалтингової діяльності в закладі вищої освіти в умовах невизначеності..... 138

Семенова І.

Основні пріоритети учителів фізичної культури використання інноваційних технологій..... 141

УДК 004.05

JEL Classification: L86

Гуменюк К. В.,

здобувачка вищої освіти,

Донецький національний університет

імені Василя Стуса, м. Вінниця

Січко Т. В.,

канд. техн. наук, доцент

кафедри інформаційних технологій,

Донецький національний університет

імені Василя Стуса, м. Вінниця

АДАПТИВНИЙ ДИЗАЙН ВЕБ-САЙТІВ:

ПРИНЦИПИ ТА МЕТОДИ

Веб-сайти стають все більш важливим елементом сучасного інтернет- простору, адаптивний дизайн є ключовим фактором для забезпечення їхньої ефективності та доступності для користувачів. Адаптивний веб-сайт – це під-хід до розробки веб-сайтів, який забезпечує їхню коректну роботу на різних пристроях та екранах, а також оптимальний вигляд та взаємодію з користувачем незалежно від розміру екрана. Впровадження адаптивних веб-сайтів [1] вже стало звичним явищем.

В основі адаптивного дизайну лежать три основні принципи [2]. Для певних дизайнів можуть використовуватися й інші принципи, але ці три є основними для всіх адаптивних сайтів: гнучкі сіткові системи (Fluid Grid Systems), використання гнучких зображень (Fluid Image Use), медіа-запити (Media Queries) [3].

Fluid Grid Systems – це методологія веб-дизайну, яка базується на викорис-танні гнучких сіток для створення адаптивних і респонсивних макетів веб-сторі-нок. У цьому підході елементи на сторінці розміщуються у відносно гнучких

колонках, які можуть змінювати свою ширину залежно від розміру екрану пристрою, на якому відображається веб-сторінка. Головна мета використання fluid grid systems – забезпечити оптимальне відображення веб-сайту на різних пристроях, таких як комп'ютери, планшети та смартфони. Це досягається за допомогою використання відсоткових значень ширини елементів, а також медіа-запитів, які реагують на розмір екрана і змінюють розміщення та стилізацію елементів відповідно до цього розміру екрана.

Fluid images – це методологія розробки веб-дизайну, яка дозволяє забезпечити адаптивність зображень на веб-сайтах. Основна ідея полягає в тому, щоб зображення масштабувалися відповідно до розміру екрану або контейнера, в якому вони відображаються. Отже, ми можемо створити одне зображення і навчити браузер масштабувати його відповідно до розміру контейнера. Для інших зображень, таких як іконки, можна використовувати файли SVG. Ці формати файлів легкі, їх можна масштабувати до будь-якої величини без втрати якості.

Media Queries – це інструкції для зміни макета сайту на основі певних умов. Наприклад, двоколонковий підхід може бути непрактичним на екрані смартфона. Можна використовувати медіа-запит, щоб вказати браузеру змінити розташування колонок на екрані, якщо розмір екрану менший за певний розмір. Цей конкретний розмір, при якому макет змінюється, називається "точкою розриву".

Розглянемо популярні методи створення адаптивного веб-сайту. Підхід, спрямований на мобільні пристрої (Mobile-First Approach)[4] – розробка дизайну починається для мобільних пристроїв і поступово удосконалюється макет для великих екранів. Це забезпечує безперебійну роботу на всіх пристроях і мінімізує час завантаження сайту.

Використовуйте фреймворки CSS, такі як Bootstrap або Foundation, які надають адаптивні сітки та компоненти "out of the box", для швидкої розробки та забезпечення адаптивності.

Використання адаптивних блоків в поєднанні з медіа-запитами та іншими методами адаптивного дизайну, можуть створювати гнучкі та адаптивні макети

та елементи, які реагують на пристрій користувача, розмір екрану та його положення.

Використовуйте масштабовані розміри шрифтів (Scalable Typography), щоб зберегти читабельність і розбірливість на різних пристроях.

Мета-тег Viewport – це HTML-тег, який використовується для управління тим, як веб-сторінка відображається на пристроях із різними розмірами екрана та різною роздільною здатністю. Цей тег дає змогу веб-розробникам вказати масштабування, ширину та початкове положення веб-сторінки.

Сьогодні поширеною практикою є створення веб-сайтів або додатків, які адаптують свій користувацький інтерфейс залежно від браузера або пристрою. Адаптивний веб-дизайн має вирішальне значення в сучасному цифровому середовищі, враховуючи різноманітність пристроїв та вподобань користувачів. Дотримання цих принципів і найкращих методів допомагає створювати веб-сайти, які надають зручний і зрозумілий користувацький інтерфейс на різних пристроях із різними характеристиками екрану. Оскільки більшість людей взаємодіють з веб-сайтами через мобільні пристрої, користувачі очікують, що веб-сайти будуть адаптивними. Отже, адаптивний веб-дизайн – це не вибір, це фундаментальна вимога до веб-розробки в сучасному різноманітному цифровому просторі.

Список використаних джерел

1. Choudhury R.G., Kumar A., Varshney D., Jain P., Malhotra V. Context Sensitive Reflow of HTML Objects, IEEE, Atlanta: GA, USA, 2013. URL: <https://ieeexplore.ieee.org/document/6690007?arnumber=6690007>.
2. Marcotte Ethan Responsive Web Design, A Book Apart, 2014. С. 13-64.
3. Interaction Design Foundation Responsive Design: Best Practices, 2023. URL: <https://www.interaction-design.org/literature/article/responsive-design-let-the-device-do-the-work>.
4. Wroblewski Luke Mobile First, A Book Apart, 2021. С. 48-108.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Одеський національний технологічний університет
Університет Інформатики і прикладних знань, м.Лодзь, Польща
Інститут комп'ютерної інженерії, автоматизації, робототехніки та
програмування ім.П.Н.Платонова

XXIV Всеукраїнська науково-технічна конференція
молодих вчених, аспірантів та студентів

«СТАН, ДОСЯГНЕННЯ ТА ПЕРСПЕКТИВИ
ІНФОРМАЦІЙНИХ СИСТЕМ І ТЕХНОЛОГІЙ»

Матеріали конференції



Одеса

18-19 квітня 2024 р.

ЗМІСТ

Список організацій, представники яких взяли участь у роботі конференції	18
Розділ 1: Математичне і комп'ютерне моделювання складних процесів	20
1. Analysis of searching methods for explosive objects using information technology and computer modeling. Сотник С.В., Придятько Д.Р. (Харківський національний університет радіоелектроніки)	20
2. Neural network approximation of odes and ODE systems. Fediaieva Y., Stehun A. (Odessa I.I.Mechnikov National University)	22
3. Comparative analysis of Nist, Diehard and Testu01 tests for assessment of statistical characteristics of generated sequences. Kikh M., Niemkova O. (Lviv Polytechnic National University)	24
4. Using models inspired by nature to control of complex processes. Munteanu S. (Technical University of Moldova)	26
5. Furniture modeling in 3DS MAX. R. Ismailova, Ainukatova A. (Turan University, Almaty, Republic of Kazakhstan)	29
6. Analysis of the impact of flash land structure on the forming quality of complex aircraft forgings. Zhang Xiang, Borysevych V. (Aerospace University -Kharkiv Aviation Institute , Kharkiv, Ukraine)	31
7. Вплив збурень на процес диференціальної гри переслідування. Бардан А.О. (Чернівецький національний університет імені Юрія Федьковича)	33
8. Моделювання випробувального комплексу для дослідження ходової частини техніки та підготовки екіпажів з водіння. Веретенников І.М., Кот В.В. (Національний технічний університет -Харківський політехнічний інститут)	34
9. Ефективне автоматичне управління процесами сушіння зерна: інформаційна основа та її реалізація. Гапонюк І.О. (ТОВ «ЗАВОД ЕЛЕВАТОРНОГО ОБЛАДНАННЯ», м. Одеса)	36
10. Моделі системного аналізу. Голенко М. К., Кучер Є. М. (Університет митної справи та фінансів)	38
11. Антиплоска задача теорії пружності для нескінченної смуги, що послаблена тріщиною. Зайцев М.Д., Журавльова З. Ю. (Одеський національний університет імені І. І.Мечникова)	40
12. Аналіз перспектив оптимізації бізнес-процесів через Cloud Networking. Крушельницька М. О., Сахарова С. В. (Одеський національний технологічний університет)	42
13. Використання програмних продуктів для технології бізнес-аналітики. Кузевич Є.В. (Вінницький торговельно-економічний інститут Державного торговельно-економічного університету)	43
14. Аналіз часу виконання та ефективності алгоритмів сортування для мови Python. Кучма	45

Ю.В. (компанія GoIT)	
15. Автоматизація оцінювання розміру програмного забезпечення на ранніх етапах роботи над проектом. Латанська Л.О., Макарова Л.М., Каіров В.О., Крамаренко А.С. (Національний університет кораблебудування імені адмірала Макарова)	46
16. Основи методу балансування навантаження в інфраструктурі як послугі (IAAS). Лисенко С.М., Гандзій Д.В. (Хмельницький національний університет)	48
17. Основи удосконаленого методу керування постачання IT-інфраструктур згідно з технологією Блокчейн. Лисенко С.М., Саух О.Е. (Хмельницький національний університет)	50
18. До питання моделювання магнітних аномалій. Макаренко Н.В., Крячок О.С. (Інститут кібернетики імені В.М. Глушкова НАН України)	52
19. Напрямки моделювання у MATLAB. Мельник О.Ю. (Вінницький торговельно-економічний інститут Державного торговельно-економічного університету)	54
20. Метод автоматизації завантаження та підготовки метеоданих для системи РОДОС.	55
Новіков А.М. (Інститут проблем безпеки атомних електростанцій Національної академії наук України)	
21. Algorithm of method for evaluating the effectiveness of a Web Node using analytical hierarchy processing. Орехов С.В., Dominov D.O., Bahatskyi N.S. (Національний технічний університет «Харківський політехнічний інститут»)	57
22. Усереднення в багаточастотних системах із залежністю частот від повільних змінних на півосі. Пастула М.О., Ривак М.П. (Чернівецький національний університет імені Юрія Федьковича)	60
23. Підходи та принципи розрахунку проводки системи керування літальним апаратом на статичну міцність. Пелих В.П. (Національний аерокосмічний університет імені М. Є. Жуковського "Харківський авіаційний інститут")	61
24. Метод для автоматизованого визначення показників живучості систем відповідального призначення. Приймак Н.І., Жук Ю.П. (Національний університет-Львівська політехніка))	63
25. Особливості та інструменти моделювання процесів реагування на надзвичайні ситуації. Прищепа В.О., Заороний А. О. (Національний університет «Чернігівська політехніка»)	65
26. Дослідження коливальних рухів поїзда, викликаних нерівностями залізничної колії. Решетнікова П.Е., Заковоротний О. Ю. (Національний технічний університет «Харківський політехнічний інститут»)	67
27. Комп'ютерне моделювання напружено-деформованого стану деталей автотранспорту. Рудик О.Ю., Войтюк І.С., Герасимчук М.І., Ніколаєнко В.В. (Хмельницький національний університет)	68

університет)	
28. До застосування комп'ютерних технологій в розробці безпілотних літальних апаратів. Слишик Т.О., Чумак І.О., Сохацький А.В. (Університет митної справи та фінансів)	70
29. Комп'ютерні технології в транспортних апаратах типу екраноплан . Телуєва В.С., Сохацький А.В. (Університет митної справи та фінансів)	72
30. Моделювання розподілу ресурсів в умовах надзвичайної ситуації. Федорчук Є.Н., Білошицький Я.О., Панченко О.А. (Національний університет «Львівська політехніка»)	74
31. Моделювання функціонування радіотехнічної системи для збільшення дальності дії системи управління безпілотного летального апарату. Чернявський О.Ю., Герасимов С.В., Марущенко В.В. (Національний технічний університет -Харківський політехнічний інститут)	76
32. Загальний вид задачі теплопровідності для двохшарового циліндру. Шимченко В.В. (Одеський національний університет імені І. І. Мечникова)	78
33. Математична модель фізики польоту пташки у грі -FLAPPY BIRD . Шняга В.М., Чехмиструк Р.Ю. (Вінницький національний технічний університет)	80
Розділ 2: Управління, обробка та захист інформації	82
1. Electronic document management in the era of digitization: features and practical application . Akhmetov A. (Turan University, Almaty, Republic of Kazakhstan)	82
2. Analysis of collecting data process on products at different stages of production. Сотник С.В., Єчевський А.Д. (Харківський національний університет радіоелектроніки)	84
3. Порівняльний аналіз методів генерації тестових даних для реляційних баз даних. Башкіров М.О. (Харківський національний університет радіоелектроніки)	86
4. Алгоритми розпізнавання шкідливого програмного забезпечення для ОС ANDROID. Бобик Д.О., Яцків В.В. (ЗУНУ)	88
5. Захист інформації незалежної ветеринарної лабораторії. Богомолова П.Б. (Одеський державний аграрний університет)	89
6. Розробка системних рішень для захисту від кіберзагроз. Болтач С.В., Голочалов Д.Л. (Одеський національний технологічний університет)	91
7. Основи методу виявлення кібератак соціальної інженерії із застосуванням телефону. Бохоцько О.О., Лисенко С.М. (Хмельницький національний університет)	92
8. Захист від кіберзагроз: сучасні підходи. Бутенко Т.А., Тутов Д.В. (Державний біотехнологічний університет)	94
9. Нормативно-правове регулювання кібербезпеки в Україні та світі. Варава В.С. (Державний торговельно-економічний університет)	96

10. Проблеми контролю якості даних в розподілених інформаційних системах. Геряк Ю.М., Берко А.Ю. (Національний університет "Львівська політехніка")	98
11. Investigation Of PostgreSQL Extensions For Work With Coordinates Of Objects On The Map. Головачов М.О. (Вінницький Національний Технічний Університет)	100
12. Криптовалюта і блокчейн: технології, правовий статус, інвестиції. Деркач Т.М., Неїжмак К.О. (Національний університет «Полтавська політехніка імені Юрія Кондратюка»)	102
13. Exploring Of Java HTTP Client Implementations. Доценко В.С. (Вінницький національний технічний університет)	103
14. Інструменти OSINT framework. Живилю Є.О., Дамян М.Ю. (Національний університет «Полтавська політехніка імені Юрія Кондратюка»)	105
15. Practice using neural network technologies in developing information and educational applications. Заволович Д.О., Хошаба О.М. (Вінницький національний технічний університет)	107
16. Towards SQL injection attacks detection using machine learning. Копп А.М., Чуйко Я.М. (Національний технічний університет «Харківський політехнічний інститут»)	108
17. Вплив налаштувань конфігураційних параметрів Apache Hadoop та Apache Spark на продуктивність режимів розгортання: стратегії та рекомендації. Коптілов Н.С. (Харківський Національний Економічний Університет ім. С.Кузнеця)	110
18. Програмне забезпечення для аналізу виконуваних файлів на предмет подібності із використанням нейронної мережі -NEUROVER . Макарова Л.М., Камінський С.С., Бризгалов М.В. (Національний університет кораблебудування імені адмірала Макарова)	112
19. Спеціалізація автоматизованих видавничо-редакційних веб платформ публікування наукових досліджень. Мороз Р.Б. (Українська академія друкарства)	114
20. Feasibility of using handshake domains compared with classic DNS. Павлюк О.-Ю.С. (Національний університет «Львівська політехніка»)	116
21. Безпека вхідної автентифікації в системах електронного розкладу навчальних закладів: виклики та заходи захисту.. Пастух С.В. (Одеський національний технологічний університет)	117
22. Класифікація загроз для інформаційно-комунікаційних систем. Пелюх О.І., Єсіна М.В. (Харківський національний університет імені В. Н. Каразіна)	119
23. Види аналізу шкідливого програмного забезпечення. Ревнюк О.В., Улічев О.С. (Європейський Університет)	121
24. Кібервійна: битва за кіберпростір у російсько-Українському конфлікті. Сакалюк О.Ю.,	123

Зігура Т.М. (Одеський національний технологічний університет)	
25. Оптимізація та забезпечення ефективної роботи систем електронного розкладу навчальних занять з використанням баз даних. Скоблова М.О. (Одеський національний технологічний університет)	125
26. Проблеми вразливостей та перспективи розвитку хмарних технологій. Усенко М.П., Бандоріна Л.М. (Український державний університет науки і технологій)	126
27. Методи поширення шкідливого програмного забезпечення. Фесенко Т.М., Топчій Ю.П. (Національний університет «Полтавська політехніка імені Юрія Кондратюка»)	128
28. Some problems in managing server computing resources using deep machine learning tools.. Хошаба О.М. (Вінницький національний технічний університет)	130
29. The use of mathematical methods and models in determining the expediency of choosing protective structures.. Хошаба О.М., Гайдаш О.С. (Вінницький національний технічний університет)	133
30. The modern innovations of developing an accounting software tool for service station suppliers and customers. Хошаба О.М., Луков В.А. (Вінницький національний технічний університет)	134
31. Main directions of software development in the field of drone control.. Хошаба О.М., Майданюк А.В. (Вінницький національний технічний університет)	135
32. Well-known methods of analysis to increase the effectiveness of implementing cottage plots. Хошаба О.М., Мартиненко Р.І. (Вінницький національний технічний університет)	137
33. Methods of increasing the efficiency of using currency operations on the Forex market.. Хошаба О.М., Остапенко Я.А. (Вінницький національний технічний університет)	138
34. Comparative characteristics of break-even point determination models in economics and business analysis.. Хошаба О.М., Свентух А.О. (Вінницький національний технічний університет)	140
35. Overview of modern authentication methods for microcontrollers. Чупа Н.Р., Чупа Т.Р. (Національний університет "Львівська політехніка")	141
36. Дослідження методів контролю та корекції помилок інформації в комп'ютерних системах обробки даних, що функціонують в системі залишкових класів. Янко А.С., Сабельнікова П.С. (Національний університет «Полтавська політехніка імені Юрія Кондратюка»)	143
Розділ 3: Нові інформаційні технології в освіті	145
1. Decision support system for planning admissions committees in higher education institutions. Melnic R. (Technical University of Moldova)	145
2. Efficient task management for academic projects: integrating modern methodologies. Pohorieltsev P.M., Smotrych A. J. (Одеський національний технологічний	147

університет)	
3. Identification and analysis of factors influencing the scheduling process in the distance learning environment in Ukraine. Sytnik O.O., Vdovitchenko O.V. (National Aerospace University "Kharkiv Aviation Institute")	149
4. Децентралізований тайм-менеджмент освітнього процесу в університеті. Баденко Д.В., Яланецький В.А. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	151
5. Інтеграція інформаційних технологій у математичні проекти для підвищення зацікавленості учнів до STEM-освіти. Брюхович М.В. (Харківський національний педагогічний університет імені Г.С. Сковороди)	153
6. Веб-форум кафедри. Буряківський С.В., Свинчук О.В., Бандурка О.І. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	154
7. Використання технологій віддаленого навчання для забезпечення вивчення вибіркового модулю з інформатики «Веб-технології» учнями загальноосвітніх навчальних закладів. Гавриленко Б.М., Дубич К.П. (Рівненський державний гуманітарний університет)	155
8. Розробка навчальної комп'ютерної програми «Англійська для програмістів». Гарбарчук І. С., Бабиш С. М. (Рівненський державний гуманітарний університет)	156
9. Використання онлайн-курсів, платформ для дистанційного навчання та хмарних технологій в освіті. Гармаш К.О., Дивак В.В. (Державний торговельно-економічний університет)	158
10. Моделювання наочних матеріалів до навчальної дисципліни «Геометричне моделювання та методи візуалізації» у середовищі BLENDER. Голінський Ю.В. (Одеський національний технологічний університет)	160
11. Використання інформаційних технологій в освітньому процесі здобувачів вищої освіти за змішаною формою навчання. Данилюк Н.М. (Національний університет "Острозька академія")	162
12. Методичні аспекти вивчення технологій Front-End розробки у курсі інформатики 10 – 11 класів. Корольов О.В. (Житомирський державний університет ім. Івана Франка)	164
13. Система моніторингу виконання завдань в навчально-виховному процесі кафедри. Котова А.А., Мангулі Ю.Д., Свинчук О.В., Бандурка О.І. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	165
14. Переваги та недоліки використання нейромереж у роботі вчителя інформатики. Лойко Ю.В., Дубич К.П. (Рівненський державний гуманітарний університет)	168

15. Педагогічні прийоми та практики застосування технологій VR та гейміфікації припідготовці до іспиту з водіння. Малюга А.І. (Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»)	170
16. Цифрові технології реалізації мікронавчання на предметі «Інформатика» у закладах загальної середньої освіти. Матвійчук І.О., Дубич К.П. (Рівненський державний гуманітарний університет)	172
17. Виклики пов'язані з використанням штучного інтелекту у науці та освіті. Новіков А.М. (Інститут проблем безпеки атомних електростанцій НАН України), Новікова Г.В., Денисенко І.В. (Національний університет «Кієво-Могилянська академія»)	174
18. Розробка анімаційних візуалізацій для платформ дистанційного навчання в STEM освіті. Олейнікова І.В., Лагода О.А., Бабута В.Є. (Київський національний університет технологій та дизайну)	176
19. Role of artificial intelligence in the educational field. Остапенко К.О. (Національний Технічний Університет "Харківський Політехнічний Інститут")	177
20. Задачі оптимізації виробничих процесів. Петров В.М. (Одеський національний технологічний університет), Познар С.С. (Агро-симо-машбуд), Жданов О.О. (ОДАБА)	179
21. Застосування інформаційних технологій на основі SolidWorks для проектування пристосувань ремонту автотранспорту. Рудик О.Ю., Мадера Р.О., Титаренко С.Б., Видиш К.А. (Хмельницький національний університет)	181
22. Прогнозування результатів вступної кампанії до закладів вищої освіти на основі моделей машинного навчання. Страхов Є.М., Чачко Н.Л. (Одеський національний університет імені І.І. Мечникова)	183
23. Освітня платформа HUMAN як інструмент інноваційної роботи шкіл під час дистанційного навчання. Туваєва А.Ю., Лебідь О.Ю. (Університет митної справи та фінансів)	185
24. Інформаційна технологія підтримки прийняття рішень з розвитку інфраструктури віртуальної країни. Царенко О.П., Нуждіна М.І. (Одеський національний університет імені І. І. Мечникова)	186
25. Використання середовищ віртуальної реальності для створення імерсивного навчального досвіду. Яценяк Д.В. (Тернопільський національний педагогічний університет імені Володимира Гнатюка)	188
Розділ 4: Проектування інформаційних систем та програмних комплексів	191
1. Development of a job search Website for people with visual impairments. Brovko A.O. (Simon Kuznets Kharkiv National University of Economics)	191

2. Prospects of using AOSP-based SINGLE-board computers for the modernization of railway transport in Ukraine. Dmytro Hlavchev, Mykyta Popello, Yuliia Lishuk (Національний технічний університет «Харківський політехнічний інститут»)	193
3. Functioning of Gossip-based and classic Kademlia protocols. Kichmarenko O.D., Yezhkova A.G. (ОНУ ім. І. І. Мечникова)	194
4. Functional design of the mobile application "SimuLearn". Kim Y., Alimbekova A.T., Zeynegabylov A.A., Rakhmanov R.M. (Turan University, Almaty, Republic of Kazakhstan)	196
5. Joint development of IT projects using the example of Github and Visual Studio. Kim Yekaterina, Kan A.Ye., Gavrilova A.L. (Turan University, Almaty, Republic of Kazakhstan)	196
6. Principles of designing the user interface of mobile currency exchange applications. Kim Yekaterina, Yegai I.A., Pan A.A. (Turan University, Almaty, Republic of Kazakhstan)	199
7. Principles of polite crawling for collecting weather data from Websites. Liutenko I., Kravets Y. (National Technical University «Kharkiv Polytechnic Institute»)	200
8. Analysis of knowledge representation methods in decision-making computing systems. Struna V., Kapustianski M., Ursu A. (Technical University of Moldova)	202
9. Практика захисту інтелектуальної власності від несанкціонованого доступу в хмарі AWS. Антонова А.Р. (Одеський національний технологічний університет)	204
10. Дослідження особливостей розробки інтернет-магазинів. Антонова А.Р., Маленков І.М. (Одеський національний технологічний університет)	205
11. Інформаційна управляюча система керування особистими фінансами та часом. Багрій А.Ю., Снігур Т.С. (Одеський національний технологічний університет)	207
12. Програмний тренажер для навчання музиці. Бєлий М. О., Владімірова В.Б. (Одеський національний технологічний університет)	209
13. Використання GOOGLE-сервісів у сфері надання адміністративних послуг. Борей Л. А., Волкова А.Ю. (Одеський національний технологічний університет)	211
14. Проектування та розробка Веб-платформи для автоматизованого створення телеграм-ботів за допомогою інструменту конструювання сценаріїв. Брильянт І.А., Чехмєструк Р.Ю. (Вінницький Національний Технічний Університет)	212
15. Телеграм бот для сканування файлів та URL на наявність вірусних компонентів. Власов А.О., Снігур Т.С. (Одеський національний технологічний університет)	213
16. Проектування архітектури програмної системи для комп'ютерного моделювання розумних об'єктів. Воробйов В.С., Ковалюк Т.В. (Київський національний університет імені Тараса Шевченка)	216
17. An empirical approach to integrated process quality assessment in the life cycle of variable software systems. Гамзаєв Р.О., Ткачук М.В. (Харківський національний університет імені	217

В.Н. Каразіна)	
18. Аналіз напрямків розвитку задачі розпізнавання емоцій людей у соціальних мережах. Ганчев С.С., Ковалюк Т.В. (Київський національний університет імені Тараса Шевченка)	219
19. Дослідження використання циклу, звичайної рекурсії та хвостової у мові програмування SCALA. Глинчук Л.Я. (Волинський національний університет імені Лесі Українки)	221
20. Метод опису метаданих об'єктів для мобільної кроссплатформної CRM-системи на платформі BAF. Глімбовський Б.В., Боровик О. В. (Хмельницький національний університет)	223
21. Розробка інтернет-магазину з продажу вінілових платівок. Гузій А.К., Швець Н.В. (ВСП «Фаховий коледж промислової автоматики та інформаційних технологій Одеського національного технологічного університету»)	224
22. Оптимізація продуктивності веб-сайтів. Гуменюк К.В., Січко Т.В. (Донецький національний університет імені Василя Стуса)	226
23. Метод оптимізації паралельної обробки даних згідно з кластерною архітектурою CUDA. Дзюбчик О. Л. (Хмельницький національний університет)	228
24. Ідентифікація та аналіз тенденцій розвитку інформаційних систем для бізнесу в Україні. Дроздов В. О., Сахарова С. В. (Одеський Національний Технологічний Університет)	230
25. Інформаційна система забезпечення навчально-виховної діяльності катедри. Дячук А.О., Свинчук О.В., Бандурка О.І. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	231
26. Методичні підходи до моделювання пристроїв на 8-бітних мікроконтролерах в програмному середовищі PROTEUS. Заєць О.Ю., Малежик М.П. (Український державний університет імені Михайла Драгоманова)	233
27. Розробка концептуальної моделі багатовимірного інформаційного простору для управління конфігураціями мікросервісних застосунків. Зінов'єв Д. В. (Харківський національний університет імені В.Н. Каразіна)	235
28. Аналіз роботи розробленого додатку для керування комп'ютером за допомогою жестів, що використовує Arduino та ультразвукові датчики. Ісайко С.В., Корнієнко Ю.К. (Одеський національний технологічний університет)	237
29. Особливості дослідження інтернет-залежності у студентів ВНЗ. Котлик С.В., Соколова	238
О.П., Мойсєєва І.О. (Одеський національний технологічний університет)	

30. Моделі, методи інформаційної технології підтримки рішень в системі управління логістичними операціями під час військового стану . Кузін Ю.В., Лифар В.О., Модестова Т.В. (Східноукраїнський національний університет імені Володимира Даля)	239
31. Крос-платформні рішення розробки мобільних застосунків. Ляшук Т.Г., Шроль Т.С. (Рівненський державний гуманітарний університет)	241
32. Проектування бази даних веб-додатку для вивчення дисципліни «Чисельні методи». Майдіков І.М., Корнієнко Ю.К. (Одеський національний технологічний університет)	242
33. Розробка бібліотеки для забезпечення надійного обміну повідомленнями в розподілених інформаційних системах. Макрушин А.М., Свинчук О.В., Бандурка О.І. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	244
34. Проектування інформаційної системи для розрахунку рівня освітлення. Малюкін О.В, Мельников О.Ю. (Донбаська державна машинобудівна академія)	245
35. Розробка Веб-сайту для бізнесу з помічником на базі Штучного Інтелекту. Могілей О.Р. (Одеський національний технологічний університет)	247
36. Використання архітектури MVC при розробці програмного засобу ведення електронної комерції. Москаленко С.С., Котлик С.В. (Одеський національний технологічний університет)	248
37. Система перерозподілу задач навантаження енергетичної мережі. Мулик О.В., Свинчук О.В., Бандурка О.І. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	250
38. Ефективне прийняття рішень в бойових умовах на основі градієнтного спуску. Назаркевич М. А., Олексів Н.Т. (NULP)	251
39. Моделі інформаційно-аналітичної оцінки програмного коду онтологічними засобами. Ненахов К.Д., Горбова О.В. (Український державний університет науки та технологій)	253
40. Методика створення інформаційної системи. Овдій А.А, Жуковецька С.Л. (Одеський національний технологічний університет)	254
41. Блокчейн платформи для підготовки та проведення виборів. Олійник Є.О. (Національний аерокосмічний університет імені М. Є. Жуковського «Харківський авіаційний інститут»)	256
42. Модель рівнів зрілості вебдодатків: від статичних сторінок до сучасних фреймворків. Острецов Д.І. (Луганський національний університет імені Тараса Шевченка)	258
43. Програмна підтримка обробки зображень. Рибалов А.Б., Владімірова В.Б. (Одеський національний технологічний університет)	259

44. Аналіз систем контролю версій в програмуванні. Романюк О.Н., Тітова Н. В., Мазур В.В (Вінницький національний технічний університет), Романюк С.О. (Національний університет «Одеська політехніка»), Котлик С.В. (Одеський національний технологічний університет)	261
45. Дослідження алгоритму тестування можливостей навантаження програмного забезпечення. Руднева Д.О., Єсіна М. В. (Харківський національний університет імені В.Н. Каразіна)	263
46. Android-застосунок для тестових фінансових операцій з криптовалютами. Сажин О.І., Шевченко І.В. (Національний аерокосмічний університет ім. М.Є. Жуковського «ХАІ»)	265
47. Навігація світом САПР: порівняльний аналіз Altium, Orcad та Mentor Graphics для проектування друкованих плат. Семенов В.В. (Чорноморський національний університет імені Петра Могили)	266
48. Ризики управління командою ІТ в комерційній установі в умовах військового стану. Серік О.А., Гайдаєнко О.В (Національний університет кораблебудування ім. адмірала Макарова)	268
49. Розробка програмного забезпечення для вивчення іноземної мови. Сімонов М.О. (ФКПАІТ ОНУ)	269
50. Розробка системи моніторингу параметрів повітря в приміщенні на основі мікроконтролера. Стоянова Р.В., Бондарь М.Д. (Фаховий коледж промислової автоматики та інформаційних технологій ОНУ)	272
51. Інформаційно-вимірювальна система якості повітря навколишнього середовища на базі IoT. Суслук Б.В. (Хмельницький національний університет), Грига В.М. (Прикарпатський національний університет імені Василя Стефаника)	273
52. Сучасні підходи проектування програмного забезпечення. Таволжан Д.О., Ковалюк Т.В. (Київський національний університет імені Тараса Шевченка)	275
53. Особливості використання фреймворку Kivu для розробки кросплатформних додатків. Тимощук А.А., Ліщина Н. М. (Луцький національний технічний університет)	277
54. Блокчейн-базована система підтримки криптовалютних операцій для ОС ANDROID. Тимчук П.В. (Хмельницький національний університет), Грига В.М. (Прикарпатський національний університет імені Василя Стефаника)	279
55. Програмне забезпечення для обчислення параметрів функціональної стійкості мережевих інформаційних систем на основі графового аналізу. Ткаченко Р.О., Свинчук О.В., Бандурка О.І. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	281
56. Використання TDD-підходу для розроблення MVP Android-застосунку для клієнтів мережі аптек. Труш М.С., Шевченко І.В. (Національний аерокосмічний університет ім. М.Є. Жуковського «ХАІ»)	283

57. Проектування програмного комплексу для імітації поведінки розумних об'єктів наприкладі розумних окулярів. Фуркало Д.Ю., Ковалюк Т.В. (Київський національний університет імені Тараса Шевченка)	284
58. Завдання ВЕБ-базованих систем, побудованих на основі Web-To-Print. Хорошевська І.О. (Харківський національний економічний університет імені Семена Кузнеця), Хорошевський О.І. (Харківський національний університет радіоелектроніки)	286
59. Розповсюдження Uniform system of accounts for the Lodging Industry на тлі постійних змін туристичного сектору. Шекепа С.С., Трач О.Р. (Одеський національний технологічний університет)	288
Розділ 5: Комп'ютерні телекомунікаційні мережі та технології	291
1. Review of research on the influence of mobile learning technologies on the development of oral speech when studying foreign languages. Ismailova R., Kistaudaev SH.N. (Turan University, Almaty, Republic of Kazakhstan)	291
2. WebRTC: a revolution in browser communication. Ismailova R., Umirdekov K. (Turan University, Almaty, Republic of Kazakhstan)	292
3. Use of hog descriptor in classification of remote sensing images. Rybnytskyi M., Kryvenko S., Lukin V., Shulga I. (National Aerospace University – "Kharkiv Aviation Institute")	294
4. Виявлення атак на протокол безпеки WPA3 у мережах стандарту IEEE 802.11 за допомогою сигнатур. Банах Р.І., Піскозуб А.З. (Національний університет "Львівська політехніка")	296
5. Стан і перспективи розвитку комп'ютерних телекомунікаційних мереж та технологій. Бандоріна Л.М., Дружин І. Є. (Український державний університет науки і технологій)	298
6. Роль технічного письменника в документуванні API. Богуцький Д.В., Горбова О.В. (Український державний університет науки і технологій)	300
7. Еволюція мережі Інтернет: від IPv4 до IPv6. Вихрист О.В. (Харківський національний університет радіоелектроніки)	301
8. Веб-сервіс для фрілансерів: аналітичний огляд, функціонал та структура. Деркач Т.М., Дмитренко Т.А., Дмитренко Андрій .О. (Національний університет «Полтавська політехніка імені Юрія Кондратюка»)	303
9. Основи методу оптимізації енергоспоживання інфраструктури Дата-центрів із використанням віртуальних машин. Карцан А.Р., Лисенко С.М. (Хмельницький національний університет)	305
10. Реалізація резервування інтернет-з'єднання на основі ROUTEROS. Кудінов Є.О. (Харківський національний автомобільно-дорожній університет)	307

11. Технологія віртуальної реальності: взаємодія з користувачем. Кулик Ю.-М. Р. (Національний університет –Львівська політехнікаII)	309
12. Моделювання функціональної надійності мережі зв'язку з встановленням з'єднання. Нєнов О.Л., Босенко І. С. (Одеський національний технологічний університет)	311
13. Ранжування вузлів комутації та каналів зв'язку при моделюванні мережінфокомунікацій. Нєнов О.Л., Миценко А.В. (Одеський національний технологічний університет)	313
14. Декомпозиція мережі на підмножині шляхів обміну інформацією. Нєнов О.Л., Холодняк М.К. (Одеський національний технологічний університет)	315
15. Історія PBR процедурних текстур та їх базові принципи. Протасов Д. Ю., Сахарова С.В. (Одеський національний технологічний університет)	317
16. Використання сучасних оптичних технологій у телекомунікаційних мережах. Рибалов А.Б., Рибалов Б.О. (Одеський національний технологічний університет)	318
17. Стекова та реєстрова архітектура віртуальної машини. Слушна Н.В. (Одеський національний технологічний університет)	320
18. Інтеграція SOFTWARE-DEFINED networks та хмарних технологій для ефективного управління мережами майбутнього. Шабров М.Ю. (Одеський національний технологічний університет)	321
19. Мультирівнева модель даних щодо кібербезпеки критичних інфраструктур: адміністративно-правове регулювання. Шкітов А.А. (Відкритий міжнародний університет розвитку людини «УКРАЇНА»)	322
20. Метод та програмно-технічні засоби оптимізації IoT інфраструктури із застосуванням концепції туманних обчислень. Шудрик А.О. (Хмельницький Національний Університет)	324
21. Застосування хмарних сервісівв нафтогазовій галузі. Юрчишин О.В. (Управліннянафтопромислового сервісу ПАТ„Укрнафтall), Юрчишин В.М. (Івано-Франківський національний технічний університет нафти і газу)	326
Розділ 6: Штучний інтелект і автоматизація робототехнічних систем	328
1. Emotional words uttered in exceptional circumstances. Borozan O. (Technical University of Moldova)	328
2. Pattern recognition in image sets using convolutional neural networks. Myshkovskiy Y.I. (Lviv Polytechnic National University)	330
3. Use of Uavs and Ugvs in swarm systems and their interaction. Rabiichuk I., Fechan A. (Національний університет "Львівська політехніка")	332
4. Штучний інтелект як ефективний засіб для аналізу вимог до програмного забезпечення. Бєляєв О.І. (Державний біотехнологічний університет)	34

5. Застосування нейронних мереж до класифікації зображень. Блажко М.О. (Дніпровський національний університет ім. Олеся Гончара)	336
6. Програма для автоматичного виявлення фейкових новин у соціальних мережах . Волосенко В.Ф., Чехмestрук Р.Ю. (Вінницький національний технічний університет)	337
7. Роль штучного інтелекту в маркетинговій діяльності. Гарна У.Ю., Ілляшенко С.М. (Національний технічний університет "Харківський політехнічний інститут")	338
8. Інтеграція штучного інтелекту в мультимедійний дизайн. Грицай С. Д. (Київський національний університет технологій та дизайну)	339
9. Інформаційна система безпеки розумного будинку. Гуйда О.Г., Черненко О.С., Омецинська Н. В. (Таврійський національний університет імені В. І. Вернадського)	341
10. Проблеми довгострокового навчання нейронних мереж. Євтушенко О.С., Заковоротний О. Ю. (Національний технічний університет «Харківський політехнічний інститут»)	343
11. Інтелектуальні системи як універсальність ветеринарії. Жмай А., Чайковська К., Шалаягіна О. (Одеський Державний Аграрний Університет)	344
12. Дослідження алгоритмів машинного навчання для автоматизації робототехнічних систем у задачах навігації. Жульковський О.О., Вохмянін Г.Я., (Дніпровський державний технічний університет), Жульковська І.І. (University of Customs and Finance)	346
13. Додаток для аналізу та класифікації музичних жанрів на основі аудіозаписів з використанням алгоритмів машинного навчання. Іванченко А.В., Чехмestрук Р.Ю. (Вінницький національний технічний університет)	347
14. Практичні застосування інтеграції штучного інтелекту в процес освіти. Капітон А.М., Гладкий С.С., Пророк М.Ю. (Національний університет «Полтавська політехніка імені Юрія Кондратюка»)	348
15. БПЛА – інноваційний тренд розвитку DER BAUER. Кіхай С.С. (Одеський Державний Аграрний Університет)	350
16. Дослідження впливу профілактичних заходів на рівень захворюваності за допомогою штучних нейронних мереж. Козуб Д.С., Мельников О.Ю. (Донбаська державна машинобудівна академія)	351
17. Прогнозування ефективності просування сайту за допомогою штучних нейронних мереж. Кривінченко Д.Р., Мельников О.Ю. (Донбаська державна машинобудівна академія)	353
18. Посилення кібербезпеки через штучний інтелект: можливості, виклики та перспективи. Куплевацька С.В., Мартинюк Г.В. (Маріупольський державний університет)	355
19. Використання штучного інтелекту для розвитку CRM-систем. Курилах А.С., Капітон	357

А.М. (Національний університет "Полтавська політехніка імені Юрія Кондратюка")	
20. Спосіб підвищення ефективності моделей штучного інтелекту. Лактіонов О.І. (Національний університет "Полтавська політехніка імені Юрія Кондратюка")	358
21. Практичне застосування безпілотних літальних апаратів для скиду об'єктів в Україні. Магеровський Д.В. (Національний університет "Львівська політехніка")	359
22. Використання патерну проектування команда при розробці програмного забезпечення для керування мобільним роботом. Москва В.В., Ушкаренко О.О. (Національний університет кораблебудування ім. адм. Макарова)	361
23. Створення та оптимізація моделі нейронної мережі для автоматичного розпізнавання дорожніх знаків на зображеннях. Назарчук Б.Г., Мороз І. П. (Рівненський державний гуманітарний університет)	363
24. AI In logistics and supply chain automation. Нечай Д.Л. (Національний університет «Львівська політехніка»)	365
25. Research of deep learning technology for building recognition. Подорожняк А.О., Lolenko A. (Національний технічний університет "Харківський політехнічний інститут")	366
26. Кіберфізична система моніторингу мікроклімату виробничих приміщень з функцією дистанційного контролю . Поташнік М.О. (Хмельницький національний університет)	367
27. Особливості трансферної моделі Bidirectional Encoder Representations From Transformers. Прочухан Д.В. (Харківський національний університет радіоелектроніки)	368
28. Метод та система нейромережевого відображення текстових послідовностей. Пшеничко О.С. (Хмельницький національний університет)	369
29. Аналіз та прогнозування аварійності на транспорті за допомогою нейронної мережі. Ситніков В.С., Лаврухін В.В., Жеребкін С.Є., Босовський В.О., Войтов В.М. (Національний університет "Одеська політехніка")	370
30. Системи первинного індикування параметрів зберігання виробничих залишків. Сторожук Д.І. (Українська академія друкарства)	372
31. Порівняльний аналіз ефективності архітектур Resnet-101 та Resnext-101 в задачах медичної діагностики. Прочухан Д.В. (Харківський національний університет радіоелектроніки)	375
32. Дослідження методів розпізнавання пошкодження наземних транспортних засобів. Телішевський П.А. (Національний університет «Львівська політехніка»)	376
33. Інтерактивні технології на основі великих мовних моделей. Чернявський Р.А., Крайник Я.М. (Чорноморський національний університет ім. Петра Могили)	378

34. Інтеграція машинного навчання у бізнес-процеси: автоматизація та ефективність. Томашевський О.М. (Національний університет «Львівська політехніка»)	379
35. Застосування генеративного штучного інтелекту для створення текстур. Чікменьов С.С., Жуковецька С.Л. (Одеський національний технологічний університет)	381
Розділ 7: Комп'ютерні ігри і WEB-дизайн	383
1. Application of MYSQL database in Web programming. Blagojce Najdovski (Faculty of Biotechnical Sciences)	383
2. Дизайн інтерфейсу в успішних мобільних додатках: аналіз тенденцій та стратегій.Алексєєнко К.В., Антоненко С.В. (Дніпровський національний університет імені О. Гончара)	386
3. Основні принципи WEB-дизайну. Бикова П.О., Свинчук О.В., Бандурка О.І.(Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	388
4. Проектування якісного та зручного дизайну інтерфейсу в комп'ютерній грі. Бовкун М.Ф. (Дніпровський національний університет імені Олеся Гончара)	399
5. Розробка інтегрованого онлайн-порталу для біженців з персоналізованими рекомендаціями. Богачук Д.В., Чехместрук Р.Ю. (Вінницький національний технічний університет)	391
6. Створення ігрового 3D персонажу. Босенко Л.С. (Одеський національний технологічний університет)	392
7. Еволюція комп'ютерного зору. Водяницький В.М., Юскович-Жуковська В.І. (Приватний вищий навчальний заклад "Міжнародний економіко-гуманітарний університет імені академіка Степана Дем'янчука", м. Рівне)	393
8. Лінійне програмування в системі управління ресурсами в іграх. Гальцев Д.Ю. (Одеський національний технологічний університет)	396
9. Основні типи UX/UI дизайну в комп'ютерних іграх. Дорош Р.О. (Житомирський державний університет імені Івана Франка)	397
10. Інтерактивні ігри в навчальному процесі. Захарчук Є.М., Свинчук О.В., Бандурка О.І. (Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського»)	399
11. Аналіз нововведень ігрового рушія Unreal Engine 5 порівняно з Unreal Engine 4. Комісаров В.Р. (Київський національний університет імені Тараса Шевченка)	400
12. Дослідження проблем переносу моделі з 3D-редактора в ігровий рушій. Крисань І.С., Жуковецька С.Л. (Одеський національний технологічний університет)	402
13. Чи дійсно комп'ютерні ігри несуть загрозу людству?. Кручковський М.С. (Університет митної справи та фінансів)	403

14. Проектування інтерфейсу онлайн версії настільної гри «Монополія». Левківський А.П. (ЖДУ імені Івана Франка)	405
15. Еволюційне моделювання у відеоіграх. Ленартович В.Г. (Національний технічний університет «Харківський політехнічний інститут»)	407
16. Про застосування геометричного моделювання у CAD-системах та дизайні. Ломовцев П.Б., Бойцова О.С. (Одеський національний технологічний університет)	408
17. Дослідження особливостей кібербезпеки комп'ютерних ігор. Мелешко Є.В., Варченко І.В. (Центральноукраїнський національний технічний університет)	409
18. Інструменти для створення ігрового штучного інтелекту в різних ігрових рушіях . Мелешко Є.В., Варченко І.В. (Центральноукраїнський національний технічний університет)	411
19. Leveraging tokens as marketing instrument in WEB3 Gaming. Ostapenko K. O. National Technical University "Kharkiv Polytechnic Institute"	413
20. Розробка дизайну веб-застосунку засобами React Bootstrap. Поліщук П.А. (Національний Університет "Одеська Політехніка")	414
21. Основні вимоги для побудови програмних систем комп'ютерної графіки. Романюк О.Н., Завальнюк Є.К. (Вінницький національний технічний університет)	416
22. Місце і роль комп'ютерної графіки в процесах сучасності. Ростоцька А.М. (Одеський державний аграрний університет)	418
23. Історія розвитку інструментів для розробки відеоігор з використанням ігрового рушія Source. Симака О.С. (Київський національний університет імені Тараса Шевченка)	420
24. Дослідження стратегічної гри реального часу з нейронною мережею. Ситніков В.С., Мінаков О.О., Діленко О.В., Молодожен Ю.М., Калиненко Д.О. (Національний університет "Одеська політехніка")	422
25. Комп'ютерні ігри та веб-дизайн: злиття технологій у світі розваг. Хмара Д.О., Шинкарьова В.А. (Університет митної справи та фінансів)	423
26. Використання ігрового рушія Unity для побудови кінематографічного простору та переваги використання на ринку. Шаповал В.В. (Київський національний університет імені Тараса Шевченка)	425
27. Формування майбутнього рендерингу в Unity через впровадження передових технологій, вдосконалення процесів та вплив на візуальну якість та імерсивність віртуальних середовищ. Шаповал В.В., Симака О.С. (Київський національний університет імені Тараса Шевченка)	427
28. Лендинг на основі інтегрованого підходу для успішного просування сайту. Шевцова Н.В., Семенюк Р.В. (Рівненський державний гуманітарний університет)	429

29. Розробка гри "Forgotten Thoroughfare". Шевченко І.О. (Національний аеро-космічний університет ім. М.Є. Жуковського)	431
30. Впровадження механік ХОРРОПу в кулінарну гру. Шестопапов С.В., Азатханов А.А., Репало Д.С. (Одеський національний технологічний університет)	431
31. Дослідження ключових особливостей історичних ігор жанру «RPG». Шестопапов С.В., Бурикін І.С., Костюк Д.С. (Одеський національний технологічний університет)	434
32. Дослідження ключових особливостей ігрових персонажів в стилі Аніме. Шестопапов С.В., Карпунін А.А. (Одеський національний технологічний університет)	436
Розділ 8: Бібліометрика. Інформатизація навчального, наукового, дослідницького процесів	438
1. Автоматизоване обслуговування користувачів науково-технічної бібліотеки ОНТУ. Гриньків С.Й., Кривенко В.С. (Одеський національний технологічний університет)	438
2. Науково технічна бібліотека ОНТУ — сьогодні. Пошук інформації для користувачів закладу вищої освіти.. Лобакова Л. П., Коваль З. Н. (Одеський національний технологічний університет)	440
3. Синергія музею та науково-технічної бібліотеки ОНТУ. Оставненко І.А., Златіна О.А. (Одеський національний технологічний університет)	441
4. Автоматизація бібліотечних технологічних процесів (з досвіду роботи науково-технічної бібліотеки Одеського Національного Технологічного Університету). Савченко Л.В., Брагінська Н.А. (Одеський національний технологічний університет)	442
5. Застосування технологій ші в діяльності науково-технічної бібліотеки ЗВО. Сиволап О.С., Ольшевська О.В. (Одеський національний технологічний університет)	444
6. Автоматизація бібліотечних процесів відділу комплектування. Скутаренко О.Л. (Одеський національний технологічний університет)	445
7. Бібліометричний аналіз у відділі наукової обробки та каталогізації документів. Тимчишина М.В., Шмігельська О.В. (Одеський національний технологічний університет)	447
Розділ 9: Інформаційні технології у медицині	449
1. Biometrics: Trends and Prospects. Kim E.R., Rogozhin P.G (Turan University, Almaty, Republic of Kazakhstan)	449
2. Впровадження моделі DeepFM для покращення алгоритмів синтезу рекомендацій у медичних системах. Герус О.О. (Національний лісотехнічний університет України)	450
3. Порівняння ефективності моделей штучного інтелекту для діагностики захворювань	452
легень. Гітис В.Б., Вареник В.В. (Донбаська державна машинобудівна академія)	
4. Медичні програми і пристрої. Роль мобільних програм для здоров'я та фітнесу у сучасному суспільстві: переваги, недоліки і перспективи. Горбачов О.С. (Донбаська державна машинобудівна академія)	454

5. Інформаційно-комунікаційні технології в телереабілітації . Гусєва-Божаткіна В.А., Шакула А.І (Національний університет кораблебудування імені адмірала Макарова)	455
6. Розробка чат-боту для підтримки програми –Домедична допомога. Алгоритм М.А.Р.С.Н. для цивільнихІІ. Живило І.О., (Харківського національний університет імені В.Н. Каразіна), Шпіка К.А. (Національний університет "Полтавська політехніка імені Юрія Кондратюка")	457
7. Розроблення алгоритму роботи системи дистанційного моніторингу психофізіологічного стану людини. Жульковський О.О., Волошина К.Р., Волошин Р.В. (Дніпровський державний технічний університет)	459
8. Оптимізація ресурсного управління в медичних закладах за допомогою ІТ-технологій. Катреча Л.В., Міценко С.А. (Державний торговельно-економічний університет)	461
9. Сучасні інноваційні технології в медицині. Козурман В.П. (Університет митної справи та фінансів)	463
10. Застосування байєсівської структури для аналізу зображень магнітно-резонансної томографії. Кравченко П.К. (Чорноморський національний університет ім. Петра Могили)	465
11. Проблеми та перспективи використання програмних застосунків у галузі охорони здоров'я. Лейбак Д.В., Кательніков Д.І. (Вінницький національний технічний університет)	467
12. Використання оперативних потужностей телемедицини при сонографічному обстеженні у пацієнтів з атеросклерозом артерій каротидного басейну. Сегін Н.Т. (Івано-Франківський національний медичний університет)	468
13. Кіберфізична система діагностики раку молочної залози з використанням нейромережі. Сіпайло А.О. (Хмельницький національний університет)	470
14. Gathering medical data from patients using wearable devices. Слоневський Є.О. (Національний технічний університет «Харківський політехнічний інститут»)	472
15. Features of using artificial intelligence for screening vitamin D deficiency in adults. Страхов Є.М., Корхова А. С. (Одеський національний університет імені І. І. Мечникова)	474
16. Розробка програмного засобу для діагностики органічних ушкоджень мозку при посттравматичних стресових розладах у учасників бойових дій. Трунова А.І., Висоцька О.В., Білецька С.Є. (Національний аерокосмічний університет ім. М. Є. Жуковського «Харківський авіаційний інститут»)	475
17. Розробка структури мікропроцесорної системи вимірювання пульсу людини методом фотоплетізмографії. Ушкаренко О.О., Савун І.А. (Національний університет кораблебудування ім. адм. Макарова)	477

18. Розробка мобільного застосунку-персонального консультанта з приготування їжі на платформі Android з використанням технологій JAVA. Щербацький Б.І., Кательніков Д.І.(Вінницький національний технічний університет)	479
Розділ 10: 3D моделювання та 3D друк	481
1. Modeling design of mobile robotic platform. Сотник С.В., Зарубін І.С. (Харківський національний університет радіоелектроніки)	481
2. Modeling of potting greenhouse design. Сотник С.В., Кирпота Ф.В. (Харківський національний університет радіоелектроніки)	483
3. Використання 3D плагінів для створення 3D моделей в межах інструменту BLENDER. Данилюк М.М. (Національний університет «Львівська Політехніка»)	485
4. Технологія створення 3D моделі поршневої системи компресора холодильної установки для навчальних цілей. Зінченко А.Ф. (Одеський національний технологічний університет)	487
5. Utilization of 3d-printing in architecture and construction. Клягін-Ізовцев П.А., Braterska N.M. (Харківський національний університет міського господарства імені О. М. Бекетова)	488
6. Нарощування 3D-розмірних електронних компонентів з фотополімерних компондів за допомогою лазерного випромінювання. Костін Д.О. (Харківський національний університет радіоелектроніки)	490
7. Про особливості створення тривимірних моделей механізмів минулих років. Котлик С.В., Соколова О.П. (Одеський національний технологічний університет)	492
8. Особливості відтворення об'єктів великого розміру методом фотограметрії. Кравченко О.В., Жуковецька С.Л. (Одеський національний технологічний університет)	494
9. Технологія створення фотореалістичної візуалізації 3d моделі автомобіля з подальшою анімацією у середовищі Blender. Лисковецький В.В. (Одеський національний технологічний університет)	496

УДК 004.03

ОПТИМІЗАЦІЯ ПРОДУКТИВНОСТІ ВЕБ-САЙТІВ

ГУМЕНЮК К.В.

(humeniuk_k@donnu.edu.ua),

СІЧКО Т.В.(t.sichko@donnu.edu.ua)

Донецький національний університет імені Василя Стуса

Робота присвячена огляду стратегій та інструментів для оптимізації продуктивності вебсайтів. Дослідження цих аспектів допоможе веброзробникам та власникам вебсайтів підвищити ефективність своїх проєктів, забезпечивши позитивний досвід взаємодії з сайтом користувачів.

Швидкість завантаження вебсторінок є ключовим фактором для успіху

вебсайтів. При стрімкому розвитку інтернет-технологій і зростаючій конкуренції в онлайн-середовищі, користувачі мають високі очікування щодо швидкодії та продуктивності вебсайтів. Час завантаження сторінок впливає на перше враження користувачів, їхню здатність знайти потрібну інформацію та загальне враження від взаємодії з вебсайтом.

Вебпродуктивність у першу чергу відноситься до швидкості завантаження вебсайту. Це важливо, тому що коротший час завантаження покращує користувацький досвід роботи із сайтом на всіх типах інтернет-з'єднань. Це допомагає в досягненні таких простих цілей, як залучення більшої кількості користувачів до відвідування та читання контенту сайту, або ж спонукання користувачів до дії. Повільні вебсайти випробовують терпіння користувачів і можуть призвести до того, що вони покинуть сайт ще до того, як побачать, що він може запропонувати [1].

Розглянемо стратегії та інструменти для прискорення завантаження сторінок вебсайтів. Одним із найпоширеніших способів прискорення завантаження сторінок є оптимізація зображень. Вона включає використання форматів зображень з високим ступенем стиснення, як от JPEG або WebP, а також зменшення розмірів зображень без втрати якості [2]. Інструменти, як от Photoshop, TinyPNG та ImageOptim дозволяють автоматизувати цей процес.

Файли CSS та JavaScript можуть бути значно зменшені шляхом мініфікації, яка видаляє зайві пробіли, коментарі та інші зайві символи. Крім того, їх можна компресувати за допомогою спеціальних інструментів, наприклад, UglifyJS для JavaScript та CSSNano для CSS [3]. Компресія додається до мініфікації шляхом використання різних алгоритмів стиснення, які дозволяють зменшити розмір файлів без втрати їхньої функціональності.

Content Delivery Network (CDN) – це розподілена система серверів, розташованих у різних місцях по всьому світу, яка призначена для ефективного доставки контенту користувачам. Ця технологія дозволяє зменшити фізичну відстань між сервером і кінцевим користувачем, що призводить до скорочення часу завантаження сторінок і покращення продуктивності вебсайту. [4]. Популярні CDN, як от Cloudflare, Amazon CloudFront і Akamai, надають високоякісні послуги доставки контенту з високим рівнем доступності та швидкодії.

Однак оптимізація розміру сторінки – лише один аспект. Інший ключовий фактор – це швидкість відповіді серверів. Для забезпечення швидкої реакції серверів можна скористатися кешуванням, яке дозволяє зберігати копії сторінок або їх елементів (таких як HTML-сторінки, зображення, CSS-файли, JavaScript) на сервері, що дозволяє уникнути повторного генерування сторінок для кожного запиту.

Крім того, використання асинхронного завантаження ресурсів дозволяє сторінці завантажуватися частково, що полегшує швидке відображення контенту користувачеві. Це особливо корисно при завантаженні великих медіафайлів, як от відео або картинки в галереях. Асинхронне завантаження дозволяє браузеру завантажувати ресурси паралельно і не блокує процес завантаження інших елементів сторінки. Як результат, користувачі можуть бачити вже доступний контент і взаємодіяти з ним, навіть якщо ще не завантажені всі елементи сторінки.

Необхідно також враховувати мобільну оптимізацію. З огляду на те, що більшість користувачів використовують мобільні пристрої для доступу до інтернету, важливо забезпечити швидку роботу вебсайту на мобільних платформах. Це можна досягти за допомогою адаптивного дизайну, який автоматично пристосовується до розміру екрану пристрою, та мінімізації використання ресурсів, які можуть бути обмежені на мобільних пристроях.

Отже, оптимізація продуктивності вебсайтів – це складний процес, що включає різноманітні стратегії та інструменти, вона стає необхідною умовою для того, щоб

забезпечити конкурентоспроможність та задоволення потреб користувачів. Важливо постійно вдосконалювати вебсайт та використовувати новітні технології для забезпечення оптимального користувацького досвіду та ефективної роботи.

СПИСОК ЛІТЕРАТУРИ

1. Jeremy L. Wagner Web Performance in Action - Building fast web pages, Manning, 2017. С. 1-3.
2. Maximiliano Firtman Hacking Web Performance, O'Reilly Media, 2018. С. 17-21.
3. Steve Souders Even Faster Web Sites: Essential Knowledge for Frontend Engineers, Oreilly & Associates Inc, 2009. С. 29-35.
4. Ilya Grigorik High Performance Browser Networking, O'Reilly Media, 2013. С. 235-236.



Міністерство освіти і науки України
Донецький національний університет імені Василя Стуса
Київський національний університет будівництва і архітектури
Черкаський державний технологічний університет
Київський національний торговельно-економічний університет
Львівський національний університет імені Івана Франка
Тернопільський національний технічний університет імені Івана Пулюя
Громадська організація «Освітня фундація продуктового ІТ»

ІНФОРМАЦІЙНИЙ ЛИСТ

**Вельмишановні колеги,
запрошуємо Вас взяти участь у**

**У ВСЕУКРАЇНСЬКІЙ НАУКОВО-ПРАКТИЧНІЙ
КОНФЕРЕНЦІЇ ЗДОБУВАЧІВ ВИЩОЇ ОСВІТИ ТА
МОЛОДИХ ВЧЕНИХ**

«Прикладні інформаційні технології»

24 травня 2024 року



За адресою, м. Вінниця, вул. 600-річчя, 21

ВАРІАНТИ ВИКОРИСТАННЯ NLP

Донецький національний університет імені Василя Стуса, м. Вінниця

У сучасному світі штучний інтелект (ШІ) і технології обробки природної мови (NLP) [1] виступають у ролі ключових факторів розвитку інновацій та автоматизації. NLP відкриває безліч можливостей для вдосконалення комунікації та оптимізації бізнес-процесів [2]. Існує безліч сценаріїв, де ці технології вже знаходять своє застосування та принесли суттєві покращення. Інновації розпізнавання мови та розуміння контексту дають змогу створювати продукти, які взаємодіють із користувачами більш інтуїтивно та ефективно.

Обробка природної мови (NLP) належить до галузі комп'ютерних наук, а точніше, до галузі штучного інтелекту (ШІ), яка займається наданням комп'ютерам здатності розуміти текст і усне мовлення так само, як це роблять люди. NLP поєднує комп'ютерну лінгвістику — моделювання людської мови на основі правил — зі статистичними методами, машинним навчанням (МН) і моделями глибокого навчання [3]. Разом ці технології дають змогу комп'ютерам обробляти людську мову у вигляді тексту або голосових даних і розуміти її повне значення, із мотивами того, хто говорить чи пише.

Обробка природної мови є рушійною силою машинного інтелекту в багатьох сучасних сферах застосування. Розглянемо декілька варіантів використання NLP. Одним зі способів є виявлення спаму. Ви можете не вважати розпізнавання спаму завданням NLP, але більшість технологій використовують можливості NLP для розпізнавання текстів, щоб сканувати електронні листи на наявність ознак, які часто вказують на спам або фішинг. Такими ознаками можуть бути зловживання фінансовими термінами, погрози, недоречна наполегливість, неправильне написання назв компаній та багато іншого. Виявлення спаму — одне з небагатьох завдань NLP.

Ще одним варіантом є машинний переклад. Google Translate — приклад широко доступної технології NLP у практичному застосуванні. Насправді ефективний машинний переклад передбачає більше, ніж просто заміну слів однієї мови на слова іншої. Ефективний переклад повинен точно передавати зміст і відтінок вхідної мови і перекладати його з тим самим змістом і бажаним ефектом. Сучасні інструменти машинного перекладу успішно працюють над цією задачею, покращуючи якість перекладу та забезпечуючи більш правильне відтворення сенсу і стилістичних особливостей вихідного тексту. Результативність машинного перекладу можна визначити не лише за точністю передачі інформації, але і за здатністю зберегти відтінок емоцій та індивідуальний стиль автора.

Віртуальні агенти, як от Siri від Apple та Alexa від Amazon, використовують розпізнавання мови для ідентифікації голосових команд і формування природної мови, для реагування на запити відповідними діями чи корисними коментарями. Ця техніка базується на глибокому навчанні та обробці природної мови (NLP), де моделі вивчають синтаксичні і семантичні структури мови для кращого розуміння користувацьких запитів. Чат-боти демонструють таку ж саму техніку, реагуючи на введені текстові повідомлення. Найліпші серед них використовують алгоритми машинного навчання, які не лише адаптуються до конкретного користувача, але й аналізують та узагальнюють широкий спектр лінгвістичних концепцій.

Ще одним зі способів є узагальнення тексту, що використовує методи NLP для обробки величезних обсягів цифрового тексту і створення підсумків та коротких описів для дослідницьких баз даних або для завантажених користувачів, які не мають часу на читання повного тексту. Найкращі програми для узагальнення тексту використовують методи семантичного аналізу та генерації природної мови (NLG) [4], щоб додати до результатів узагальнення корисну інформацію та висновки.

Аналіз думок у соціальних мережах із використанням технології обробки природної мови (NLP) став необхідним інструментом для багатьох компаній у сучасному бізнесі. Ця технологія дає змогу вивчати та розуміти мову, яку користувачі використовують у своїх постах, відгуках та рецензіях, щоб виявити їхні ставлення, емоції та вподобання. Завдяки аналізу думок у соціальних мережах за допомогою NLP компанії мають можливість отримувати цінні інсайти щодо сприйняття своїх продуктів, акцій та подій. Розкриття реальних відгуків та реакцій користувачів допомагає підприємствам адаптувати свої стратегії розвитку, виправляти недоліки та покращувати якість своїх товарів та послуг.

Технологія обробки природної мови знаходить широке застосування в різних сферах. NLP вдосконалює комунікацію, оптимізує бізнес-процеси та сприяє автоматизації завдань, що раніше вимагали значних людських ресурсів. Завдяки постійному розвитку та вдосконаленню, обробка природної мови продовжує відкривати нові можливості для технологічного прогресу. Використання NLP стає необхідністю для компаній, які прагнуть залишатися конкурентоспроможними та адаптуватися до бізнес-середовища, що швидко змінюється.

Список використаних джерел

1. Jurafsky D., Martin J. Speech and Language Processing: An Introduction to Natural Language Processing, Pearson, 2008. С. 265-337.
2. Eisenstein J. Introduction to Natural Language Processing, MIT Press, 2019. С. 48-59.
3. Goldberg Y. Neural Network Methods in Natural Language Processing, Morgan and Claypool Publishers, 2017. С. 2-6.

Areces C., Koller, A., Striegnitz K. Referring expressions as formulas of description logic, Association for Computational Linguistics, 2008. URL: https://www.researchgate.net/publication/228370565_Referring_Expressions_as_Formulas_of_Description_Logic.

