Міністерство освіти і науки України

Ніжинський державний університет імені Миколи Гоголя

Навчально-науковий інститут природничо-математичних, медико-

біологічних наук та інформаційних технологій

Кафедра інформаційних технологій, фізико-математичних та

 економічних наук

Освітня програма: Комп’ютерні науки

Спеціальність:122 Компʼютерні науки

**КВАЛІФІКАЦІЙНА РОБОТА**

на здобуття освітнього ступеня *бакалавр*

**Дослідження та аналіз інновацій**

**у веб-розробці, використання нових технологій, дизайну та інтерфейсів**

студента **Краснікова Сергія Олександровича**

**Науковий керівник:**

Бугаєць Наталія Олександрівна,

кандидат педагогічних наук, доцент

**Рецензент:**

Лисенко Ірина Миколаївна,

кандидат фізико-математичних наук, доцент

**Допущено до захисту:** \_\_\_\_ \_\_\_\_\_\_ 2024 р.

Завідувач кафедри

проф. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Казачков І.В.

Ніжин – 2024

**АНОТАЦІЯ**

Дослідження та аналіз інновацій у веб-розробці є актуальною темою в сучасній інформаційній технологічній сфері. Ця дослідницька робота спрямована на вивчення новітніх технологій, дизайну та інтерфейсів, які впливають на веб-розробку. Використання нових технологій у веб-розробці дає змогу створювати більш ефективні, інтерактивні та зручні веб-додатки та сайти. У процесі дослідження проведено аналіз сучасних трендів у веб-розробці, огляд новітніх інструментів та фреймворків, а також вивчення методів оптимізації дизайну та інтерфейсів для поліпшення користувацького досвіду. Результати цього дослідження можуть бути корисними для веб-розробників, дизайнерів та компаній, які прагнуть залишатися на передових позиціях.

**Ключові слова:** веб програмування, UI/UX дизайн, фреймворк, штучний інтелект, інновація.

**ABSTRACT**

Research and analysis of innovations in web development is a pertinent topic in today's technological landscape. This research aims to explore the latest technologies, designs, and interfaces that impact web development. Utilizing new technologies in web development enables the creation of more efficient, interactive, and user-friendly web applications and sites. The research encompasses an analysis of current trends in web development, an overview of cutting-edge tools and frameworks, as well as the study of methods for optimizing design and interfaces to enhance user experience. The findings of this research can be valuable for web developers, designers, and companies striving to maintain a competitive edge.

**Keywords:** web programming, UI/UX design, framework, artificial intelligence, innovation.

**ЗМІСТ**

[ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ 5](#_Toc168321003)

[ВСТУП 6](#_Toc168321004)

[Розділ 1.](#_Toc168321005) [ЕКСКУРС В ІСТОРІЮ ІНТЕРНЕТУ І ВЕБ-РОЗРОБКИ. МОВИ ПРОГРАМУВАННЯ, ЩО ВИКОРИСТОВУЮТЬСЯ ДЛЯ ВЕБ-РОЗРОБКИ 9](#_Toc168321006)

[1.1. Екскурс в історію веб-програмування 9](#_Toc168321007)

[1.2. Мови програмування для веб-розробки 10](#_Toc168321008)

[1.2.1. HTML та CSS 10](#_Toc168321009)

[1.2.2. Різниця між CSS та СSS3 11](#_Toc168321010)

[1.2.3. JavaScript 12](#_Toc168321011)

[1.2.4. Python 13](#_Toc168321012)

[1.2.5. PHP 13](#_Toc168321013)

[1.2.6. Java 14](#_Toc168321014)

[1.2.7. C# 15](#_Toc168321015)

[Висновки до першого розділу 16](#_Toc168321016)

[Розділ 2.](#_Toc168321017) [ІННОВАЦІЇ У ВЕБ-РОЗРОБЦІ 17](#_Toc168321018)

[2.1. Фреймворки і бібліотеки в веб-розробці 17](#_Toc168321019)

[2.1.1. Node.js 18](#_Toc168321020)

[2.1.2. Vue.js 20](#_Toc168321021)

[2.1.3. React JS 20](#_Toc168321022)

[2.1.4. Angular.js 21](#_Toc168321023)

[2.1.5. Django 22](#_Toc168321024)

[2.1.6. Rails 22](#_Toc168321025)

[2.1.7. Spring Boot 22](#_Toc168321026)

[2.1.8. Laravel 23](#_Toc168321027)

[2.2. Кібербезпека 23](#_Toc168321028)

[2.3. Доповнена та віртуальна реальність у веб-додатках 25](#_Toc168321029)

[2.3.1. Доповнена реальність (AR) 25](#_Toc168321030)

[2.3.2. Віртуальна реальність (VR) 26](#_Toc168321031)

[2.3.3. Майбутнє AR/VR для веб-розробки 27](#_Toc168321032)

[2.4. Безсерверна архітектура 27](#_Toc168321033)

[Висновки до другого розділу 28](#_Toc168321034)

[Розділ 3.](#_Toc168321035) [ШТУЧНИЙ ІНТЕЛЕКТ У ВЕБ-ПРОГРАМУВАННІ 30](#_Toc168321036)

[3.1. Штучний інтелект 30](#_Toc168321037)

[3.2. Переваги та недоліки штучного інтелекту 33](#_Toc168321038)

[3.3. Як працює ШІ 35](#_Toc168321039)

[3.4. Інструменти та сервіси ШІ 37](#_Toc168321040)

[3.5. Сфери застосування ШІ 39](#_Toc168321041)

[3.6. Засоби штучного інтелекту для веб-розробки 46](#_Toc168321042)

[3.6.1. Процеси веб-розробки за допомогою ШІ 48](#_Toc168321043)

[3.6.2. Приклади інтеграції технологій ШІ з веб-розробкою 48](#_Toc168321044)

[3.6.3. Компоненти упровадження ШІ у веб-розробку 51](#_Toc168321045)

[3.6.4. Конфіденційність даних 53](#_Toc168321046)

[3.6.5. Прогнозовані тенденції та інновації застосування ШІ у веб-розробці 53](#_Toc168321047)

[Висновки до третього розділу 55](#_Toc168321048)

[Розділ 4.](#_Toc168321049) [ТРЕНДИ UI/UX ДИЗАЙНУ 56](#_Toc168321050)

[4.1. Динамічні мікроанімації 56](#_Toc168321051)

[4.2. Технологія 3D 58](#_Toc168321052)

[4.3. Оригінальна типографіка 59](#_Toc168321053)

[4.4. Мобільна адаптація 59](#_Toc168321054)

[4.5. Голосовий інтерфейс користувача 61](#_Toc168321055)

[4.6. Персоналізація 62](#_Toc168321056)

[Висновки до четвертого розділу 63](#_Toc168321057)

[ВИСНОВКИ 66](#_Toc168321058)

[СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 69](#_Toc168321059)

[ДОДАТКИ 72](#_Toc168321060)

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕНЬ

AI Artificial Intelligence

ШІ Штучний інтелект

GPT Generative pre-trained transformer

WF Web framework

AR Augmented Reality

VR Virtual Reality

ВСТУП

**Актуальність теми.** Світ веб-розробки постійно розвивається, і бути в курсі останніх тенденцій веб-розробки має вирішальне значення як для компаній, так і для розробників. Дивлячись у 2024 рік, можна побачити захоплюючі досягнення, які сформують майбутнє веб-розробки. Поінформованість про ці тенденції веб-розробки допоможе створювати сучасні та ефективні веб-сайти, починаючи від нових технологій і закінчуючи мінливими очікуваннями користувачів.

Таким чином, щоб бути конкурентоспроможними, розробники повинні йти в ногу з найновішими тенденціями та технологіями веб-розробки через майбутні тенденції веб-розробки, починаючи від появи прогресивних веб-додатків і оптимізації голосового пошуку до підвищеного значення доступності та кібербезпеки.

Згідно з нещодавнім опитуванням SWEOR , 85% користувачів вважають, що веб-сайт має бути оптимізованим для мобільних пристроїв, тоді як 38% перестануть використовувати веб-сайт, якщо макет непривабливий. Крім того, дослідження показують, що веб-сайти з швидшим часом завантаження мають вищий коефіцієнт конверсії, потенційно зростаючи на 7% за кожну секунду збереження. Ці статистичні дані підкреслюють важливість упровадження найкращих тенденцій веб-розробки у 2024 році для задоволення потреб користувачів і досягнення успіху в бізнесі [15].

Швидкий темп технологічного прогресу ставить перед розробниками веб-сайтів та додатків низку викликів і можливостей. Використання новітніх технологій, інноваційних дизайнів та інтерфейсів дає змогу створювати продукти, які не лише відповідають потребам користувачів, а й визначають нові стандарти віртуального середовища.

Задля розуміння інновацій у веб-розробці, необхідно оглянути різноманітні аспекти, що становлять цей обширний дисциплінарний простір. Нові технології, такі як штучний інтелект, блокчейн, інтернет речей (IoT), віртуальна та доповнена реальність, та інші, змінюють спосіб, якими ми сприймаємо та взаємодіємо зі світом через веб. Аналіз цих технологій допомагає розробникам та дослідникам краще розуміти їхні можливості та обмеження, а також спрогнозувати їх вплив на майбутні тренди у веб-розробці.

Нові технології відкривають двері для інноваційних підходів до дизайну та інтерфейсів. Удосконалення користувацьких інтерфейсів, упровадження відповідного та доступного дизайну, а також розвиток персоналізованих веб-досвідів стають ключовими напрямами веб-розробки. Дослідження в області інтерфейсів та дизайну допомагає зрозуміти, які взаємодії та візуальні концепції ефективніше приваблюють та задовольняють користувачів.

Ще одним важливим аспектом дослідження інновацій у веб-розробці є аналіз ринку та конкурентного середовища. Визначення потреб та очікувань аудиторії, оцінка конкуренції та ідентифікація ніш для розвитку стають критичними етапами в розробці успішних веб-продуктів.

Таким чином, дослідження та аналіз інновацій у веб-розробці створюють фундаментальну базу для створення сучасних та ефективних веб-продуктів. Вони допомагають розробникам розуміти та використовувати нові технології, вдосконалювати дизайн та інтерфейси та адаптуватися до змін у веб-середовищі.

**Об’єкт дослідження** – веб-розробка.

**Предмет дослідження** – інновації у веб-розробці, використання нових технологій, дизайну та інтерфейсів.

**Мета кваліфікаційної роботи** полягає у вивченні, дослідженні та аналізі сучасних інновацій у веб-розробці на основі використання новітніх технологій, дизайну та інтерфейсів.

Відповідно до вказаного об’єкту, предмету і для досягнення поставленої мети визначені такі **завдання** роботи:

1. Провести огляд літератури та джерел інформації з питань сучасних тенденцій у веб-розробці, включаючи наукові статті, книги та веб-ресурси.
2. Дослідити та проаналізувати нові технології, що використовуються в веб-розробці, зокрема JavaScript-фреймворки, CSS-препроцесори, серверні технології та інші.
3. Вивчити та проаналізувати сучасні дизайнерські тенденції у веб-розробці, зокрема мікроанімації, 3D-графіку.
4. Розглянути питання впливу інноваційних технологій та дизайну на користувацький досвід та ефективність веб-проектів.
5. Зробити висновки та сформулювати рекомендації щодо подальшого розвитку веб-розробки та використання інновацій у цій галузі.

**Методи дослідження:** огляд джерел інформації, аналітичні методи опрацювання даних, методи аналізу, синтезу, порівняння, узагальнення та систематизації, спостереження та моделювання.

**Наукова новизна** роботи полягає в оновленні розуміння сучасних тенденцій у веб-розробці та інноваційних підходів до використання новітніх технологій, дизайну та інтерфейсів. Ця робота спрямована на заповнення прогалин у розумінні та застосуванні новітніх підходів у веб-розробці, а також на сприяння розвитку більш інноваційних та ефективних веб-проектів у майбутньому.

Робота має **практичне значення**, оскільки результати дослідження будуть корисними веб-розробникам та компаніям для ефективного впровадження нових технології в свої веб-проекти, що сприятиме поліпшенню їх функціональності та конкурентоспроможності. Рекомендації, розроблені на основі аналізу, допоможуть розробникам створювати веб-інтерфейси, які будуть більш ефективними, зручними для використання та приємними для користувачів. Застосування новітніх технологій та дизайну дає змогу створювати веб-сайти та додатки, які забезпечують високий рівень задоволення та залученості користувачів. Також результати дослідження можуть надихнути інших розробників до впровадження новаторських підходів та розвитку веб-технологій, що сприятиме появі нових можливостей у цій сфері.

Результати дослідження були представлені під час доповіді на студентській науково-практичній конференції «Молодь у науці», що відбулась 16-24 травня 2024 р. у Ніжинському державному університеті імені Миколи Гоголя.

Розділ 1

ЕКСКУРС В ІСТОРІЮ ІНТЕРНЕТУ І ВЕБ-РОЗРОБКИ. МОВИ ПРОГРАМУВАННЯ, ЩО ВИКОРИСТОВУЮТЬСЯ ДЛЯ ВЕБ-РОЗРОБКИ

Поширення Інтернет-технологій у масовій культурі може бути пояснене їхнім внеском у візуальні електронні медіа. Зростання популярності Інтернету у масовій культурі корелює з його історією як візуального середовища. Щоб населення зрозуміло потенціал Інтернету, воно потребувало графічного інтерфейсу. Персональний комп'ютер не став би настільки поширеним без монітора, і до цього часу не знайшов би широкого застосування, поки електронні таблиці, текстові процесори і відеоігри не привернули користувачів. Так само користувачі не почали би використовувати Інтернет до тих пір, поки не з'явилися візуально-орієнтовані веб-браузери на початку 1990-х років. І незважаючи на значні покращення в технологіях та дизайні, деякі з ранніх методів веб-дизайну залишалися актуальними протягом багатьох років.

1.1. Екскурс в історію веб-програмування

Веб-програмування сформувалося в результаті еволюції комп'ютерних технологій та потреб користувачів. Ось кілька ключових етапів у формуванні веб-програмування:

* *початок Інтернету (1960-і роки).*Перші кроки веб-програмування пов'язані з розробкою мережі ARPANET, яка була передвісником Інтернету. ARPANET був створений як дослідницький проект Міністерства оборони США для забезпечення зв'язку між науковими та військовими установами;
* *створення WorldWideWeb та перших браузерів (1990-ті роки).* У 1990 році Тім Бернерс-Лі створив перший веб-браузер WorldWideWeb (пізніше перейменований на Nexus), що відкрив нові можливості для взаємодії з інформацією в Інтернеті. Це був початок масового застосування веб-технологій;
* *поява HTML та CSS.* Мова розмітки гіпертексту HTML (HyperText Markup Language) виникла для визначення структури веб-сторінок, а мова каскадних таблиць стилів CSS (Cascading Style Sheets) – для їхнього оформлення. Ці мови дали можливість розробникам створювати веб-сторінки з різноманітним виглядом та функціоналом;
* *поява JavaScript.* Мова програмування JavaScript була створена у 1995 році і стала ключовою для створення інтерактивних елементів на веб-сторінках, що дало можливість розробникам створювати динамічні та більш інтерактивні веб-додатки;
* *розвиток серверних технологій.* Виникнення мов програмування для серверного середовища, зокрема PHP, Python, Ruby, Java, дозволило створювати складні веб-додатки, які можуть взаємодіяти з базами даних та забезпечувати різноманітні сервіси користувачам [6].

Ці етапи сприяли появі та розвитку веб-програмування, яке стало необхідною складовою частиною сучасного Інтернету, визначаючи його функціонал та зручність користування.

1.2. Мови програмування для веб-розробки

Для веб-розробки використовуються різні мови програмування, які мають свої унікальні особливості та застосування.

1.2.1. HTML та CSS

HTML (HyperText Markup Language) – мова розмітки гіпертексту, що використовується для створення структури веб-сторінок. Нею визначається, як вміст сторінки буде організований та інтерпретований браузером. HTML складається з тегів та їх атрибутів, що визначають такі елементи на сторінці як заголовки, параграфи, списки, таблиці, зображення та посилання, які допомагають структурувати та відображати інформацію на веб-сторінці.

Структура HTML дуже важлива для SEO (Search Engine Optimization), оскільки пошукові системи використовують її для індексації та ранжування сторінок.

Каскадні таблиці стилів CSS (Cascading Style Sheets) використовуються для оформлення та стилізації веб-сторінок. За допомогою них визначають вигляд та відображення елементів HTML.

CSS складається з правил, які описують розмір, колір, шрифт, відступи, рамки та інші властивості елементів HTML. Використання CSS допомагає розділити структуру та оформлення веб-сторінок, що робить код більш чистим та підтримуваним. Завдяки CSS можна легко змінювати вигляд всієї веб-сторінки, змінюючи лише кілька правил.

У результаті узгодженого використання HTML та CSS розробники створюють красиві, добре структуровані та інтерактивні веб-сайти, які надають користувачам приємний досвід перегляду.

1.2.2. Різниця між CSS та СSS3

CSS і CSS3 – це версії мови каскадних таблиць стилів, що використовуються для візуального оформлення веб-сторінок. Однак, CSS3 є розширеною версією CSS з новими функціями та можливостями. Різниця між CSS і CSS3 полягає в:

* *синтаксисі і специфікації.*  CSS – це перша версія мови і має базовий функціонал та синтаксис, що складається з властивостей, значень і селекторів. CSS3 – це розширена версія CSS, яка доповнена новими модулями, властивостями і функціями;
* *підтримка браузерів.* Більшість браузерів підтримують CSS у великій мірі, оскільки це базова технологія для веб-розробки. Підтримка CSS3 може відрізнятися в залежності від конкретних функцій і властивостей. Деякі нові можливості можуть не підтримуватися старішими версіями браузерів;
* *нові функції і властивості.* Оригінальний CSS складається з базового набору властивостей і можливостей, таких як робота з кольорами, шрифтами, межами і т.д. CSS3 містить багато нових функцій, таких як анімація, переходи, тіні, градієнти, гнучкі шрифти, адаптивні макети і багато іншого;
* *підтримка анімації та ефектів.* Оригінальний CSS підтримує базові анімаційні можливості, але має обмежену підтримку для складних анімацій та переходів. CSS3 має розширені можливості анімації, такі як @keyframes, що використовується для створення складних і плавних анімаційних ефектів без використання JavaScript;
* *медіазапити.* В CSS підтримка медіазапитів є обмеженою, що ускладнює створення адаптивних дизайнів. CSS3 має розширені можливості медіазапитів, які дозволяють більш точно контролювати вигляд веб-сторінки в залежності від характеристик пристрою, таких як розмір екрану, орієнтація та роздільна здатність.

Узагальнюючи, CSS3 є більш розширеною та функціональною версією CSS з багатьма новими можливостями, які спрощують веб-розробку та дозволяють створювати більш динамічні та естетичні веб-сторінки.

1.2.3. JavaScript

JavaScript є однією з найбільш важливих мов програмування для веб-розробки. Особливості JavaScript у веб-програмуванні:

* *динамічні веб-сторінки*. За допомогою засобів JavaScript створюють динамічний зміст на веб-сторінках, що реагує на дії користувача без перезавантаження сторінки. До таких динамічних елементів на сторінці відносять анімацію, зміну контенту, валідацію форм, реакцію на події миші та клавіатури, та багато іншого;
* *клієнтська валідація форм.* Скрипт мовою JavaScript може перевіряти введені дані користувача безпосередньо в браузері перед їх відправленням на сервер, що допомагає зменшити кількість помилок та полегшує взаємодію з формами;
* *AJAX.* Використовуючи JavaScript, виконують асинхронні запити до сервера без перезавантаження сторінки. Це дає змогу оновлювати частини сторінки без зміни всього її вмісту, що полегшує роботу з великими обсягами даних та підвищує швидкість веб-сайту;
* *розробка веб-додатків.* JavaScript використовується для розробки клієнтських додатків (SPA – Single Page Applications), що мають багато переваг: зменшення часу завантаження, більш інтерактивний інтерфейс та покращена відповідь на дії користувача;
* *розробка ігор.* JavaScript використовується для створення браузерних ігор, що працюють у веб-браузері без потреби встановлення додаткового програмного забезпечення;
* *реактивні фреймворки.* JavaScript підтримує велику кількість реактивних фреймворків та бібліотек, зокрема React.js, Vue.js, Angular, що дозволяє розробникам створювати складні клієнтські додатки з допомогою компонентного підходу.

Загалом, JavaScript є ключовою технологією для веб-розробки, що надає засоби для створення багаторівневих, інтерактивних та ефективних веб-додатків.

1.2.4. Python

Python використовується у веб-програмуванні для розробки веб-додатків та веб-сайтів. Ключові аспекти використання Python у веб-розробці:

* *фреймворки для веб-розробки.* Python має різноманітні фреймворки, що полегшують створення веб-додатків. Найпопулярніші з них: Django і Flask;
* *широкі можливості.* Python має велику кількість бібліотек та модулів для веб-розробки. Наприклад, бібліотека Requests використовується для взаємодії з веб-службами, а бібліотека BeautifulSoup для розбору HTML-коду;
* *розробка API.* Python часто використовується для розробки API (Application Programming Interface) для взаємодії між різними компонентами програмного забезпечення або для взаємодії з клієнтами. Flask та Django дозволяють швидко створювати RESTful API;
* *машинне навчання та аналітика.* Python – це популярна мова програмування для реалізації алгоритмів машинного навчання та обробки даних. Веб-додатки, що використовують аналітику або машинне навчання, часто базуються на Python;
* *використання у великих проектах.* Python широко використовується у великих веб-проектах, таких як Instagram, Pinterest та Dropbox, що свідчить про його потужність та масштабованість.

Узагальнюючи, Python – це потужний та гнучкий інструмент для веб-розробки, який забезпечує широкі можливості та простоту використання.

1.2.5. PHP

PHP (Hypertext Preprocessor) – це мова програмування, яка широко використовується для веб-розробки. Особливості PHP під час використання у веб-програмуванні:

* *простий синтаксис.* Одним з головних переваг PHP є його легкість у освоєнні. Він має простий синтаксис, схожий на мови, такі як C або Perl, що дає змогу розробникам швидко створювати веб-додатки;
* *вбудована підтримка веб-розробки.* Мова PHP спеціально розроблена для створення динамічних веб-сайтів. Вона містить вбудовані функції та бібліотеки для роботи з HTTP-запитами, роботи з формами, роботи з базами даних та іншими важливими аспектами веб-розробки;
* *розширені можливості.* PHP має багато розширень та фреймворків, які полегшують розробку великих та складних веб-додатків. Найпопулярніші фреймворки включають Laravel, Symfony, CodeIgniter та Yii;
* *підтримка баз даних.* За допомогою PHP реалізується підтримка багатьох реляційних баз даних, таких як MySQL, PostgreSQL, SQLite та інших, що дозволяє легко взаємодіяти з базами даних для зберігання та опрацювання даних на веб-сайті;
* *популярність і велика спільнота.* PHP є однією з найпопулярніших мов програмування для веб-розробки, і вона має велику активну спільноту розробників, які надають підтримку, документацію та розширення;
* *швидкість розробки.* Багато функцій PHP спрямовані на прискорення процесу розробки, такі як шаблони (наприклад, Smarty), що дає можливість розробникам легко розділяти логіку та представлення веб-сторінок.

Хоча PHP має свої переваги, ця мова також має свої недоліки, зокрема проблеми з безпекою та низька масштабованість. Однак з використанням правильних інструментів та практик PHP може бути потужним інструментом для створення різноманітних веб-додатків та сайтів.

1.2.6. Java

Мова програмування Java використовується в веб-розробці завдяки своїм перевагам і різноманітним можливостям. Переваги Java:

* *платформа незалежна від обладнання.* Java використовує віртуальну машину Java (JVM), що дозволяє цій мові працювати на будь-якій операційній системі. Це робить Java ідеальним вибором для створення веб-додатків, які можуть працювати на різних платформах;
* *багатопотоковість.* У Java підтримується багатопотоковість, що є умовою для створення швидкодіючих та ефективних веб-додатків, які можуть опрацьовувати багато запитів одночасно;
* *широкий вибір фреймворків.* У веб-розробці з Java широко використовуються різноманітні фреймворки, такі як Spring, Hibernate, Struts, JSF та інші. Ці фреймворки полегшують розробку веб-додатків, забезпечуючи різноманітні інструменти та бібліотеки;
* *безпека.* Java має вбудовані механізми безпеки для створення веб-додатків, захищених від різних типів атак, переповнення буфера тощо;
* *широкі можливості застосування.* Java має багато розширень і бібліотек для різних задач веб-розробки, включаючи роботу з базами даних, створення веб-сервісів, обробку запитів HTTP та багато іншого;
* *масштабованість.* Java добре підходить для створення великих та масштабованих веб-додатків, які можуть опрацьовувати великі обсяги даних та велику кількість користувачів.

Хоча Java може бути більш складною мовою програмування для веб-розробки порівняно з іншими мовами, вона надає розробникам широкий набір інструментів та можливостей для створення потужних веб-додатків.

1.2.7. C#

C# є потужною мовою програмування, яка використовується для створення різноманітних програмних застосунків, включаючи веб-додатки. У використанні C# у веб-програмуванні є свої переваги:

* *ASP.NET.* ASP.NET є фреймворком для створення веб-додатків за допомогою мови програмування C#. ASP.NET дозволяє розробникам швидко будувати веб-додатки та використовувати різні підходи, такі як Web Forms, MVC (Model-View-Controller), а також Web API для розробки веб-служб;
* *широкі можливості застосування.* C# має багато функцій та можливостей, щоб створювати різноманітні веб-додатки, включаючи ефективні, масштабовані та безпечні реалізації;
* *інтеграція з іншими технологіями.* C# може легко інтегруватися з іншими технологіями та сервісами, такими як бази даних, хмарні сервіси, сервіси кешування та багато інших, що розширює можливості розробки;
* *широка підтримка від Microsoft.* C# та ASP.NET отримують широку підтримку від Microsoft, що включає в себе інструменти розробки, документацію, форуми та інші ресурси, які полегшують розробку та підтримку веб-додатків на цій платформі;
* *масштабованість та продуктивність.* Завдяки ефективності мови та фреймворку, веб-додатки, розроблені з використанням C#, можуть бути масштабованими та продуктивними, що робить їх ідеальними для побудови великих та складних веб-проектів.

Отже, використання C# у веб-програмуванні розширює можливості розробників у процесі створення потужних та ефективних веб-додатків.

Висновки до першого розділу

Існує велика кількість мов програмування для веб-розробки, включаючи JavaScript, Python, PHP, Java, Ruby та інші. Кожна мова має свої переваги і недоліки, що дає можливість розробникам вибирати той інструмент, який найбільш підходить для їхнього проекту.

JavaScript та Python, володіють потужними засобами і можливостями використання та одночасно є досить простими для вивчення і застосування. Це полегшує розробку веб-додатків для початківців та допомагає швидко розгортати проекти. Також ці мови користуються великою популярністю та мають великі спільноти розробників. Це означає, що є багато документації, бібліотек, фреймворків та ресурсів, які полегшують розробку та підтримку веб-додатків.

Деякі мови, наприклад Java та C#, мають велику продуктивність та ефективність, що дозволяє створювати великі та масштабовані веб-додатки. Вони підходять для великих команд розробників та підприємств.

Часто веб-розробка складається з цілих технологічних стеків, які містять різні мови програмування, фреймворки, бібліотеки та інструментів. Вибір правильних компонентів для технологічного стеку є важливою складовою успішного проекту. У загальному, вибір мови програмування в веб-розробці залежить від конкретних потреб проекту, вмінь розробника та його вподобань, а також від характеристик проекту, таких як масштаб, складність та продуктивність.

Розділ 2

ІННОВАЦІЇ У ВЕБ-РОЗРОБЦІ

Майбутнє веб-розробки виглядає яскравим, як ніколи. У 2024 році з’являється більше технологій, інструментів та фреймворків, тому необхідно бути в курсі останніх тенденцій. Для збереження конкурентоспроможності треба знати та вміти своєчасно використовувати найбільш актуальні інструменти розробника.

2.1. Фреймворки і бібліотеки в веб-розробці

Веб-фреймворки є життєво важливими інструментами для сучасних розробників. Вони забезпечують основу для кодування та багаторазових модулів, щоб зробити процес розробки простішим і швидшим. Використовуючи веб-фреймворк, розробники можуть зосередитися на унікальних аспектах своїх програм замість написання повторюваного коду [7].

Веб-фреймворк (WF) або фреймворк веб-додатків (WAF) — це програмний фреймворк, розроблений для підтримки розробки веб-програм, включаючи веб-сервіси, веб-ресурси та веб-інтерфейси API. Веб-платформи забезпечують стандартний спосіб створення та розгортання веб-додатків у мережі [3].

Фреймворк веб-розробки — це структурований набір інструментів, бібліотек і угод, призначених для оптимізації та спрощення процесу створення веб-додатків. Ці фреймворки забезпечують основу та попередньо встановлену структуру, що допомагає розробникам зосередитися на функціях, що стосуються конкретної програми, а не кодувати з нуля.

Фреймворки веб-розробки зазвичай включають компоненти для зовнішньої та внутрішньої розробки. Для фронтенду вони пропонують інструменти для створення адаптивних інтерфейсів користувача, обробки взаємодії користувачів і керування потоком даних. Для серверної частини вони забезпечують структуру для серверної логіки, взаємодії з базами даних та інших завдань, пов’язаних із сервером.

Розглянемо деякі ключові аспекти фреймворків веб-розробки:

* Попередньо написаний код: фреймворки забезпечують структуру для веб-додатків, тож не потрібно кодувати з нуля, включаючи основні функції для керування даними та їх відображення на веб-сторінках.
* Бібліотеки та інструменти: вони часто постачаються з бібліотеками для доступу до бази даних, фреймворків шаблонів і керування сеансами, і вони часто сприяють повторному використанню коду.
* Функції безпеки: Frameworks також пропонують функції безпеки для захисту від загроз, таких як впровадження SQL, міжсайтовий сценарій (XSS) та інші.
* Ефективність і швидкість: забезпечуючи структуроване середовище, фреймворки можуть зробити процес розробки швидшим і ефективнішим.
* Архітектура MVC: багато фреймворків дотримуються шаблону Model-View-Controller (MVC), який розділяє модель даних із бізнес-правилами (Model), інтерфейсом користувача (View) і введенням користувача (Controller).
* Спільнота та підтримка: більшість фреймворків веб-розробки мають сильні спільноти та підтримку. Це означає, що доступна велика кількість документації, форумів і інструментів сторонніх розробників.

2.1.1. Node.js

Node.js — це міжплатформне середовище виконання JavaScript із відкритим вихідним кодом, створене Раяном Далем у 2009 році. Завдяки асинхронному керованому подіями середовищу виконання JavaScript Node.js розроблено для розробки масштабованих мережевих програм. В Node.js є багато фреймворків для прискорення процесу розробки, кожен з яких має свої переваги та особливості.

Adonis.js – це повнофункціональний фреймворк, побудований на основі принципів MVC (Model-View-Controller). Він надає розробникам зручні інструменти для створення веб-додатків. Наприклад, він має ORM (Object-Relational Mapping) для спрощення роботи з базами даних, механізм маршрутизації, що дозволяє легко налаштовувати маршрути в додатку, а також підтримку автентифікації, яка допомагає в управлінні процесом аутентифікації користувачів. В цілому, Adonis.js робить процес розробки веб-додатків більш простим і ефективним за рахунок надання потужних інструментів і концепцій, з якими можна працювати.

З AdonisJs є можливість створювати потужні веб-додатки з використанням принципів MVC

Головною особливістю з його допомогою потужні веб-додатки з використанням принципів MVC. Популярним серед розробників цей фреймворк роблять:

* *Розширюваність:* в AdonisJs є безліч плагінів та розширень, для розширення функціональності додатків.
* *Підтримка WebSocket:* в AdonisJs є вбудована підтримка протоколу WebSocket, що дає можливість створювати додатки, що працюють в реальному часі та забезпечує двосторонній зв'язок клієнт-сервер.
* *Зручність тестування:* AdonisJs підтримує модульне тестування контролерів, сервісів і маршрутів, та інших компонентів додатків.
* Завдяки фреймворку AdonisJs розробники можуть швидко розгорнути додаток, використовуючи готові компоненти замість написання повторюваного коду та сконцентруватися на розробці бізнес-логіки.

Express.js – це платформа сервера веб-додатків Node.js, спеціально розроблена для створення односторінкових, багатосторінкових і гібридних веб-додатків. Він легкий та гнучкий. Використовується для розробки веб-додатків з використанням Node.js.

Express.js — це мінімальний і гнучкий фреймворк веб-додатків Node.js, який надає надійний набір функцій для веб- і мобільних додатків, забезпечуючи рішення, орієнтоване на продуктивність. Він використовується для створення RESTful API і відомий своєю простотою, швидкістю та масштабованістю, використовується розробниками для оптимізації розробки на сервері.

Загалом Express ідеально підходить для розробки складних функцій із меншим кодом і більшою гнучкістю.

Проміжне програмне забезпечення в Express.js також допомагає опрацьовувати запити та відповіді, що полегшує реалізацію функцій аналізу вхідних запитів або налаштування файлів cookie.

Хоча інші фреймворки підходять для складних програм, Express.js залишається неперевершеним за своєю швидкістю та універсальністю у веб-розробці.

Ідеально підходить для масштабованих веб-додатків, він пропонує швидке налаштування, що робить його ідеальним для проектів, що включають API, мікросервіси та програми реального часу. Express.js прискорює розробку, забезпечуючи як гнучкість, так і безперебійну роботу користувача.

Одними з ключових переваг Express є потужні можливості маршрутизації, помічники HTTP, підтримка механізмів шаблонів, спеціальний виконуваний файл для швидкого створення програм та багато іншого.

2.1.2. Vue.js

Vue.js — це прогресивний фреймворк JavaScript для створення інтерфейсів користувача. Простота інтеграції Vue.js забезпечує ефективну та інтерактивну веб-розробку. Його часто використовують для створення інтерфейсів користувача та односторінкових програм.

Vue.js є ідеальним вибором, якщо потрібен фреймворк, який поєднує простоту з потужними функціями. У Vue.js пропонується легка траєкторія навчання, масштабованість і бездоганна інтеграція, що робить його хорошим вибором для проектів будь-якого розміру.

Впровадження Vue.js забезпечує безперебійну розробку, у процесі створення адаптивних, модульних програм, які можна підтримувати. Система реактивності та компонентна архітектура Vue.js покращують організацію коду та роблять його засобом для масштабованих та динамічних веб-проектів.

Основна бібліотека Vue.js зосереджена на рівні перегляду, що полегшує добір та інтеграцію з іншими бібліотеками чи існуючими проектами.

2.1.3. React JS

React JS — бібліотека JavaScript з відкритим вихідним кодом, презентована Facebook у 2013 році, і з того часу вона стала однією з найпопулярніших бібліотек інтерфейсної розробки у світі. Він був розроблений для вирішення проблем, з якими стикається Facebook під час розробки великих веб-додатків, які повинні обробляти значні обсяги даних і швидкі зміни. Після появи React отримав широке поширення серед провідних компаній, серед яких Netflix, Airbnb, Uber, Instagram і Dropbox, а також тисячі стартапів.

React ідеально підходить для створення користувальницьких інтерфейсів і користується популярністю за свою ефективність і гнучкість у створенні інтерактивних і складних веб-додатків. Компонентна React структура забезпечує швидшу розробку, покращену взаємодію з користувачем і легке обслуговування. React.js підвищує ефективність проекту, забезпечуючи конкурентну перевагу в розробці надійних і зручних програм.

Розроблений Facebook, React зосереджується на створенні повторно використовуваних компонентів, які керують станом, що призводить до більш передбачуваного коду, який легше налагодити.

React також має сильну підтримку спільноти, багату екосистему та широке впровадження, що зміцнює його статус глобального інструменту для веб-розробників.

2.1.4. Angular.js

Angular.js–це потужна веб-програма з відкритим вихідним кодом від Google, яка використовується технологічними гігантами, наприклад, PayPal.

Google представив Angular з початковою версією AngularJS, розробленою Міско Гевері та Адамом Абронсом. Він був офіційно запущений у 2010 році, щоб оптимізувати веб-розробку та тестування, надаючи основу для створення динамічних односторінкових програм. Angular спрощує інтерфейсну розробку завдяки таким функціям, як двостороннє зв’язування даних і підтримка TypeScript.

Angular.js також відзначається багатим набором функцій: двостороннє зв’язування даних, обробка форм, маршрутизація та впровадження залежностей, що спрощує процес розробки та підвищує продуктивність.

На відміну від бібліотек, які зосереджуються на рівні перегляду, Angular надає повну структуру, включаючи інструменти для тестування, розробки та розгортання програм.

2.1.5. Django

Django – це високорівневий веб-фреймворк Python, що приваблює можливосями швидкої розробки та чистим, прагматичним дизайном. Він дотримується архітектурного шаблону Model-View-Controller (MVC) і містить систему об’єктно-реляційного відображення (ORM) для операцій з базою даних. Django орієнтований на максимальну автоматизацію.

Django також високо цінується за його функції безпеки, оскільки він допомагає розробникам уникати поширених помилок безпеки, надаючи структуру, яка була розроблена для автоматичного захисту веб-сайту.

2.1.6. Rails

Ruby on Rails, який часто називають Rails, – це фреймворк із повним набором веб-додатків, написаний на Ruby. В ньому закладені принципи конфігурації і неповторення, що підкреслює простоту та продуктивність в розробці.

Rails також відомий своєю повнотою, яка охоплює як фронтенд, так і бекенд розробку. Це включає в себе все, від рівня презентації (HTML, CSS, JavaScript) до взаємодії з базою даних. Цей інтегрований підхід означає, що ви можете створювати повну програму лише за допомогою Rails без необхідності жонглювати кількома різними фреймворками чи мовами.

2.1.7. Spring Boot

Spring Boot є розширенням Spring framework і спрощує розробку готових до виробництва програм на основі Java. Він оснащений такими вбудованими функціями, як впровадження залежностей, керування конфігурацією та вбудований веб-сервер, що полегшує початок роботи з веб-розробкою на Java.

Однією з його видатних особливостей є впровадження залежностей. Це може здатися складним, але насправді це просто шаблон проектування, де об’єкти не створюють свої залежності самі, а замість цього надають їх із зовнішнього джерела (наприклад, Spring Framework).

Ідея полягає в тому, щоб мати більш модульний код, який можна тестувати та підтримувати.

Іншим ключовим аспектом Spring є підтримка аспектно-орієнтованого програмування (AOP), яка відокремлює наскрізні проблеми (наприклад, журналювання та безпеку) від бізнес-логіки.

Це чудово, оскільки це не тільки робить код чистішим, але й простіше підтримувати та масштабувати.

2.1.8. Laravel

Laravel — це фреймворк веб-додатків PHP, відомий своїм елегантним синтаксисом і виразними функціями. Він дотримується шаблону Model-View-Controller (MVC) і надає інструменти для таких завдань, як маршрутизація, кешування та автентифікація. Laravel популярний для створення сучасних програм PHP.

Щось, що справді виділяє мене, це те, що Laravel дуже люблять за його здатність спрощувати завдання, типові для проектів PHP, як-от автентифікація, маршрутизація, сеанси та кешування.

Іншою вражаючою особливістю Laravel є його елегантний і виразний синтаксис, розроблений, щоб зробити процес розробки не тільки легшим, але й приємнішим.

Laravel також має власний полегшений механізм створення шаблонів під назвою Blade, який інтуїтивно зрозумілий і допомагає створювати дивовижні макети за допомогою своїх структур керування.

Цей фреймворк веб-розробки також містить широкий набір інструментів і функцій, які допомагають із такими завданнями, як черги, події в реальному часі та заплановані завдання, що робить його дуже придатним для створення сучасних веб-додатків у реальному часі.

2.2. Кібербезпека

Однією з тенденцій, на яку можна зробити ставку в 2024 році, є те, що кібербезпека буде критичнішою, ніж будь-коли.

На тлі дедалі складніших кібератак архітектура нульової довіри (ZTA) стає популярною моделлю безпеки, яка передбачає, що жоден користувач або пристрій не є надійними за замовчуванням, незалежно від їхнього розташування чи мережі. Цей підхід ставатиме все більш актуальним у 2024 році, оскільки компанії прагнуть захистити свої веб-додатки від нових кіберзагроз.

Це особливо примітно, оскільки ZTA змінює традиційну мережеву безпеку, припускаючи, що потенційні загрози можуть виникати з будь-якого місця, як всередині, так і поза мережею. Він посилює безпеку за допомогою мікросегментації, доступу з найменшими привілеями та постійного моніторингу, ефективно адаптуючись до сучасних складних ІТ-середовищ, таких як хмарні служби та віддалена робота.

Підвищення рівня кібербезпеки й надалі залишатиметься домінуючою тенденцією у 2024 році, і в результаті прогресивні стратегії, зокрема архітектура нульової довіри, отримають все більше поширення.

Оскільки WordPress продовжує домінувати на ринку CMS, його широке використання робить його особливо вразливим. Нові моделі безпеки, наприклад, ZTA, можуть бути ефективним компонентом стратегії безпеки WordPress, оскільки їх суворий підхід добре відповідає унікальним викликам сайтів WordPress.

Принцип ZTA – ніколи не довіряти жодному користувачу чи пристрою за замовчуванням може зменшити ризики, пов’язані із застарілими плагінами та темами, які становлять основну частину загроз безпеці WordPress. Забезпечуючи постійну автентифікацію та авторизацію для кожного запиту на доступ, ZTA може запобігти неавторизованому доступу, навіть якщо в цих компонентах існує вразливість.

Крім того, завдяки акценту ZTA на мікросегментації та доступі з найменшими привілеями, потенційна шкода від порушення безпеки може бути значно стримана. Навіть якщо зловмисник отримує доступ через плагін або тему з невиправленою вразливістю, його здатність пересуватися по мережі та отримувати доступ до конфіденційних даних значно обмежена.

По суті, інтеграція принципів нульової довіри до стратегії безпеки будь-якого веб-сайту може додати глибини його механізмам захисту. ZTA не тільки доповнює регулярні оновлення та технічне обслуговування, але й забезпечує динамічний, проактивний підхід проти нових кіберзагроз.

2.3. Доповнена та віртуальна реальність у веб-додатках

Щоб забезпечити роботу AR/VR в Інтернеті, багато веб-розробників почали писати плагіни. До альтернатив, які AR/VR може покращити, відносять:

* Створення оригінального онлайн-контенту. Навіть недосвідчені розробники тепер можуть легше створювати оригінальний контент на основі AR/VR для Інтернету завдяки швидкому зростанню доступності бібліотек з відкритим кодом.
* Збільшений вибір вмісту. Поєднуючи ці дві технології, підприємства можуть надавати віртуальні тури, які транспортують клієнтів до місць призначення, які вони інакше не змогли б дослідити.
* Тактика на почуття клієнтів. Ця конкретна маркетингова стратегія набуває популярності, оскільки вона привертає увагу користувачів і спонукає їх взаємодіяти та співпрацювати з продуктом до прийняття рішення про покупку.

Кожен приклад використання технології було підсумовано разом із причинами, чому це корисно для створення передових веб-сайтів і програм.

2.3.1. Доповнена реальність (AR)

За допомогою доповненої реальності (AR) користувачі можуть використовувати смартфон або веб-камеру для перегляду фізичного світу навколо себе, але вони також бачитимуть додаткові виготовлені зображення та функції, накладені на реальні речі. AR служить провідником між користувачами та віртуальною реальністю, полегшуючи взаємодію залежно від місця розташування. Завдяки розвитку технологій «підгонки», включаючи React-360, веб-розробники можуть легко інтегрувати панорамні зображення та відео на свої веб-сайти, створюючи безліч онлайн-додатків, заснованих на доповненій реальності.

Оскільки веб-розробка все більше і більше зміщується в бік мобільних пристроїв, доповнена реальність буде широко використовуватися в мобільних інтерфейсах користувача, оскільки вона стане більш доступною. З технологічного боку варто зазначити, що JavaScript займає провідне місце в цій галузі, враховуючи, що значна частина розробки AR наразі відбувається в онлайн-середовищі. Веб-дизайнери можуть використовувати набір функціональних можливостей, які пропонує JavaScript, щоб створювати нові програми доповненої реальності або додавати перегляди доповненої реальності до вже існуючих.

Крім того, Google використовує технологію доповненої реальності (AR) у своєму браузері Chrome. Як описано в блозі Google Chromium, у Chrome 79 представлений імерсивний API пристрою WebXR, який тепер розширює підтримку доповненої реальності. Крім того, Google інтегрував підтримку для WebXR Hit Test API, що дає можливість демонструвати цифрові елементи в поєднанні з матеріальними об’єктами реального світу.

2.3.2. Віртуальна реальність (VR)

Віртуальна реальність чинить значний вплив на розвиток сектору веб-розробки. На відміну від доповненої реальності (AR), VR часто вимагає спеціального обладнання, наприклад гарнітур, і пропонує повне занурення, яке переносить користувачів подалі від їх фізичного оточення. Точніше кажучи, користувачі повністю занурюються у сфабриковане середовище, взаємодіючи з ним так само, як у реальному світі. VR найкраще працює з веб-розробкою. Клієнти будуть зачаровані можливістю досліджувати продукти, не виходячи з дому, завдяки захоплюючим і приголомшливим візуальним ефектам, які пропонує 3D веб-дизайн. Такий рівень залучення може суттєво впливати на поведінку та рішення клієнтів.

Додаткові функції VR, які підвищують привабливість веб-сайту, охоплюють: створення онлайн-салонів, які висвітлюють конкретні характеристики продукту; значне скорочення часу на покупки, надаючи можливість випробувати продукти; подібно до доповненої реальності (AR), технологію віртуальної реальності (VR) можна використовувати під час руху.

Мобільна віртуальна реальність, прикладом якої є використання Google Cardboard, полегшує досвід віртуальної реальності шляхом підключення смартфона до портативного пристрою віртуальної реальності.

WebVR – це широко поширений інтерфейс прикладного програмування (API) на основі JavaScript, який полегшує інтеграцію пристроїв віртуальної реальності з веб-браузерами. Це найкращий вибір для створення захоплюючих веб-сайтів VR. Ключові технології, які використовує WebVR, зокрема React 360, Three.js і A-Frame, дозволяють створювати VR-досвід найвищого рівня. Спеціалізовані команди Mozilla та Chrome активно співпрацюють, щоб розширити підтримку WebVR. Зараз він працює в експериментальних версіях Chrome і нічних випусках Firefox. Після того, як ці вдосконалення перейдуть у стабільні випуски, користувачі без зусиль отримають доступ до вмісту доповненої реальності (AR) і віртуальної реальності (VR), усуваючи потребу в додаткових процедурах. Тим часом такі технічні гіганти, як Microsoft, Facebook та інші, продовжують інвестувати в розвиток технологій VR.

2.3.3. Майбутнє AR/VR для веб-розробки

Наразі веб-розробники продовжують стикатися з труднощами у боротьбі з технологіями AR та VR. Звичайно, у короткостроковій перспективі розробникам потрібно буде використовувати великі дані та аналітику, щоб краще зрозуміти потреби своїх користувачів, щоб отримати вигоду від AR/VR. Але зрештою вважається, що ці дві інновації повністю поглинуть сектор веб-розробки. Оскільки споживачі шукають передові інструменти для уявлення можливих сценаріїв покупки під час руху, виникне підвищений попит на інформацію преміум-якості. Протягом наступних років AR/VR розвиватиметься та пропонуватиме повністю налаштований, розширений і простіший для доступу користувач. Маркетологи можуть отримати доступ до збільшеного потоку даних у реальному часі, які базуються на вподобаннях користувачів. Цей приплив дасть їм змогу гнучко налаштовувати свої маркетингові та рекламні інструменти та стратегії, включаючи обмін повідомленнями, терміни та позиціонування. Зрештою, така еволюція створить широкий простір для виходу на ринок нових пропозицій продуктів. Наприклад окуляри AR стануть реальною заміною камер смартфонів.

2.4. Безсерверна архітектура

Amazon був першим із «Великої трійки» хмарних провайдерів, який публічно запустив безсерверну обчислювальну послугу. Це була AWS Lambda, анонсована в листопаді 2014 року. У 2016 році вийшли на ринок Google Cloud Functions і Microsoft Azure Functions.

Безсерверна архітектура – це тренд у веб-розробці, який продовжить поширюватися у 2024 році. Її часто називають «функцією як послуга» (FaaS), вона позбавляє розробників необхідності керувати серверами.

Натомість ви можете зосередитися на написанні коду та розгортанні функцій, підвищуючи масштабованість і економічну ефективність.

Безсерверний підхід дає змогу програмам працювати на хмарних серверах. Тож не потрібно турбуватися про доступність сервера, його потужність або керування інфраструктурою. AWS, Microsoft Azure Functions, Google Cloud Functions та інші пропонують такі послуги. Крім того, це дуже економічно ефективно, оскільки вартість послуги зазвичай розраховується залежно від реального використання ресурсів.

Цю тенденцію можна застосувати в розвитку більшості компаній для ідентифікації зображень, обробки мультимедіа, чат-ботів і помічників, систем сповіщень, додатків Інтернету речей, збору даних тощо.

Безсерверна архітектура знаходить застосування в різних сферах, таких як обробка зображень, відеоконтент, соціальні мережі та обробка даних IoT. Це особливо корисно для таких завдань, як динамічна зміна розміру зображень, перекодування відео, аналіз зображень, зроблених безпілотником, і обробка даних датчиків IoT. Крім того, безсерверні фреймворки відмінно справляються з віртуальними помічниками, чат-ботами та завданнями, керованими подіями, як-от реагування на зміни в сховищі чи базах даних. Вони також підходять для традиційних робочих навантажень, таких як HTTP REST API, веб-додатків, мобільних серверів, безперервної інтеграції та конвеєрів доставки, що робить їх ідеальним вибором для впровадження практик DevOps.

Висновки до другого розділу

Наразі існує велика безліч інструментів і технологій для веб-розробки. Від постійної інтеграції штучного інтелекту до впровадження нових технологій і систем безпеки, майбутнє веб-розробки та цифрового досвіду, яке воно надасть, виглядає яскравішим, ніж будь-коли.

Фреймворки дедалі більше спрощують роботу розробників, дозволяючи приділити більше уваги і часу бізнес-аналітиці, а не написанню повторюваного коду.

Розвивається кібербезпека. З’являються нові методи захисту. Від двофакторної аутентифікації до архітектури нульової довіри.

Технології доповненої та віртуальної реальності все більше стають частиною нашого життя. В майбутньому це відкриває безліч можливостей, особливо для бізнесу.

В той час, як безсерверний підхід дозволить економити ресурси компаній на купівлю і обслуговування серверів, що позитивно вплине на стійкість системи і дозволить зосередитись на більш важливих задачах.

Розділ 3

ШТУЧНИЙ ІНТЕЛЕКТ У ВЕБ-ПРОГРАМУВАННІ

Штучний інтелект (ШІ) відноситься до комп’ютерних систем, здатних виконувати складні завдання, які історично могла робити лише людина, наприклад міркувати, приймати рішення або вирішувати проблеми.

Сьогодні поняття "штучний інтелект" описує широкий спектр технологій, які забезпечують функціонування багатьох сервісів та продуктів, які ми використовуємо щоденно - від програм, що рекомендують фільми, до віртуальних асистентів, які надають підтримку клієнтам у реальному часі.

3.1. Штучний інтелект

Штучний інтелект, або скорочено ШІ – це галузь комп’ютерних наук, в якій досліджується розробка машин і систем, здатних виконувати завдання, що зазвичай вимагають людського інтелекту, такі як навчання, розв’язання проблем і прийняття рішень. В основі ШІ лежить ідея створення машин, що можуть мислити й міркувати, як люди, і можуть вчитися на власному досвіді, щоб з часом покращувати свою продуктивність. Сфера штучного інтелекту постійно розвивається і має потенціал революціонізувати багато аспектів нашого життя – від охорони здоров’я і фінансів до транспорту і розваг [13].

Штучний інтелект (ШІ) – це процес, коли комп'ютерні системи моделюють різні аспекти людського мислення. Технології ШІ, такі як експертні системи, обробка природної мови та машинне зору, охоплюють широкий ряд інструментів: від машинного навчання до обробки природної мови (NLP). У світі сьогодення ШІ широко застосовується в різних сферах, від хмарних сервісів до споживчих додатків і вбудованого програмного забезпечення. Наприклад, у 2022 році популярність ШІ зросла завдяки відомим додаткам Generative Pre-Training Transformer та OpenAI DALL-E, які можуть перетворювати текст у зображення та створювати текст за допомогою засобів штучного інтелекту. Широке поширення таких інструментів, як ChatGPT (рис. 3.1), стало основою для використання терміну ШІ в свідомості багатьох користувачів. Важливою рисою ШІ є його здатність до раціонального мислення і виконання завдань для досягнення певної мети, створення програм, що навчаються і адаптуються до нових даних, використовуючи методи машинного навчання і глибокого навчання.



Рисунок 3.1. – Приклад використання ChatGPT

Незважаючи на численні філософські розбіжності щодо того, чи існують «справжні» розумні машини, коли сьогодні більшість людей використовують термін ШІ, вони мають на увазі набір технологій машинного навчання, таких як Chat GPT або комп’ютерне бачення. Це приклади машин для виконання завдань, які раніше могли виконувати лише люди, як-от створення письмового вмісту, керування автомобілем або аналіз даних.

На найпростішому рівні машинне навчання використовує алгоритми, навчені наборами даних, для створення моделей машинного навчання. На їх основі комп’ютерні системи виконують такі завдання, як створення рекомендацій пісень, визначення найшвидшого шляху подорожі до місця призначення або переклад тексту з однієї мови на іншу. До найпоширеніших прикладів ШІ, які використовуються сьогодні, відносять:

*ChatGPT*: використовує великі мовні моделі (LLM) для генерування тексту у відповідь на запитання чи коментарі, поставлені до нього.

*Перекладач Google*: використовує алгоритми глибокого навчання для перекладу тексту з однієї мови на іншу.

*Netflix*: використовує алгоритми машинного навчання для створення персоналізованих механізмів рекомендацій для користувачів на основі їх попередньої історії переглядів.

*Tesla:* використовує комп’ютерний зір для забезпечення функцій автономного керування на своїх автомобілях.

Штучний інтелект поширений у багатьох галузях. Автоматизація завдань, які не вимагають людського втручання, економить гроші та час і може зменшити ризик помилок людини. Ось кілька способів застосування ШІ в різних галузях:

* фінансова галузь: виявлення шахрайства є помітним прикладом використання ШІ у фінансовій галузі. Здатність штучного інтелекту аналізувати великі обсяги даних дозволяє йому виявляти аномалії або шаблони, які сигналізують про шахрайство.
* індустрія охорони здоров'я: робототехніка на основі штучного інтелекту може підтримувати операції поблизу дуже делікатних органів або тканин, щоб зменшити крововтрату або ризик інфікування.
* робототехніка: це галузь інженерії, що займається розробкою та створенням роботів. Ці механічні апарати використовуються для виконання завдань, які можуть бути складними або монотонними для людей. Наприклад, вони застосовуються на виробничих лініях автомобільних заводів або в програмах космічних агентств, таких як NASA, для переміщення важких об'єктів у космосі. Деякі дослідники також використовують технології машинного навчання для створення роботів, які можуть взаємодіяти в соціальних мережах.
* безпілотні автомобілі: автономні транспортні засоби використовують поєднання комп’ютерного бачення, розпізнавання зображень і глибокого навчання для формування автоматизованих навичок керування транспортним засобом, залишаючись на певній смузі та уникаючи несподіваних перешкод, таких як пішоходи.

3.2. Переваги та недоліки штучного інтелекту

Штучні нейронні мережі та технології глибокого навчання штучного інтелекту стрімко розвиваються, оскільки вони дозволяють штучному інтелекту ефективно опрацьовувати великі обсяги даних набагато швидше та точніше, ніж це можливо для людини.

У порівнянні з людиною-дослідником, якому б це зайняло надто багато часу, програми штучного інтелекту, які базуються на машинному навчанні, можуть автоматично аналізувати та перетворювати ці масиви даних у корисну інформацію швидко та ефективно.

Оскільки використання штучного інтелекту стає все більш поширеним у різних продуктах і послугах, організації також повинні бути свідомими потенційної можливості створення систем, які можуть бути упередженими або дискримінаційними, як випадково, так і свідомо.

*Переваги ШІ:*

* добре справляється з роботами, де важливі деталі. Штучний інтелект виявив себе досить ефективним, а навіть кращим за лікарів у діагностиці певних видів раку, зокрема раку молочної залози та меланоми.
* зменшення часу, потрібного для виконання завдань, що включають обробку великих обсягів даних. Штучний інтелект широко використовується в галузях, де потрібно працювати з великими обсягами інформації, таких як банківська справа, фармацевтика та страхування, для прискорення аналізу масивних даних. Наприклад, фінансові установи регулярно використовують штучний інтелект для обробки кредитних заявок та виявлення фінансових маніпуляцій.
* зберігає трудові ресурси та підвищує ефективність. Наприклад, використання автоматизації на складах стало більш поширеним під час пандемії, а з інтеграцією штучного інтелекту та машинного навчання ця тенденція очікується зросте.
* забезпечує стабільні результати. Найкращі інструменти машинного перекладу штучного інтелекту гарантують високу узгодженість, дозволяючи навіть невеликим компаніям ефективно спілкуватися з клієнтами на їхній рідній мові.
* може підвищити задоволеність клієнтів за допомогою персоналізації. Штучний інтелект може адаптувати контент, повідомлення, рекламу, рекомендації та веб-сайти для кожного окремого клієнта.
* віртуальні агенти на основі ШІ завжди доступні. Програмам ШІ не потрібно спати чи робити перерви, забезпечуючи цілодобове обслуговування [17].

*Недоліки ШІ:*

* може виявитися дорогою технологією;
* вимагає глибокої технічної експертизи;
* обмежений доступ до кваліфікованих працівників для розробки інструментів штучного інтелекту;
* не може переносити навички, отримані в одному завданні, на інше;
* створює ризики зменшення робочих місць для людей та збільшення рівня безробіття [23].

3.3. Як працює ШІ

У зв'язку зі зростанням інтересу до штучного інтелекту, постачальники продуктів та послуг інтенсивно рекламують використання цієї технології. Часто те, що вони називають штучним інтелектом, фактично є лише однією складовою, наприклад, машинним навчанням. Для роботи з штучним інтелектом необхідне спеціалізоване обладнання та програмне забезпечення для написання та навчання алгоритмів машинного навчання. Ні одна мова програмування не може повністю замінити штучний інтелект, проте Python, R, Java, C++ і Julia є серед популярних серед розробників цієї технології.

Системи штучного інтелекту в основному працюють, оброблюючи великі обсяги маркованих навчальних даних. Вони аналізують ці дані на наявність кореляцій і шаблонів, а потім використовують отриману інформацію для прогнозування майбутніх станів. Наприклад, чат-бот, який отримує текстові зразки, може навчитися створювати реалістичні діалоги з людьми. Також інструменти розпізнавання зображень можуть навчитися ідентифікувати та описувати об'єкти на зображеннях, проаналізувавши мільйони прикладів. Нові методи генерації штучного інтелекту, що швидко розвиваються, можуть створювати реалістичний контент, такий як текст, зображення, музика та інші медіа.

Програмування штучного інтелекту фокусується на когнітивних навичках, що включають таке:

* Навчання: цей аспект орієнтований на збір даних та створення правил для їх перетворення на корисну інформацію. Правила, які відомі як алгоритми, надають обчислювальним пристроям послідовні інструкції для виконання конкретних завдань.
* Міркування: цей аспект програмування штучного інтелекту спрямований на вибір оптимального алгоритму для досягнення потрібного результату.
* Самокорекція: цей аспект створення штучного інтелекту призначений для постійного налаштування алгоритмів та забезпечення максимально точних результатів.
* Творчість: цей аспект штучного інтелекту використовує різноманітні методи, такі як нейронні мережі, системи на основі правил та статистичні методи, для генерації нових зображень, текстів, музики та ідей.
* Автоматизація: у поєднанні з технологіями штучного інтелекту, інструменти автоматизації можуть розширити спектр та обсяг виконуваних завдань. Наприклад, роботизована автоматизація процесів (RPA), яка базується на програмному забезпеченні, автоматизує повторювані завдання обробки даних на основі правил, які традиційно виконувалися людьми. У поєднанні з машинним навчанням і новими інструментами штучного інтелекту RPA може автоматизувати значні частини корпоративних процесів, дозволяючи таким роботам передавати дані від штучного інтелекту та реагувати на зміни в процесі [14].

Машинне навчання. Це наука про те, як навчити комп'ютер працювати без явного програмування. Глибоке навчання, у свою чергу, є підмножиною машинного навчання, яке, в основному, можна описати як автоматизацію прогностичного аналізу. Існують три типи алгоритмів машинного навчання:

* Контрольоване навчання. Набори даних маркуються таким чином, щоб виділити шаблони, які потім можна використовувати для класифікації нових наборів даних.
* Навчання без учителя. Набори даних не маркуються і групуються за схожістю або відмінністю.
* Навчання з підсиленням. Набори даних не маркуються, але після виконання дій система штучного інтелекту отримує зворотний зв'язок, який допомагає покращити її дії.
* Машинний зір. Ця технологія дозволяє машинам бачити. Вона зафіксовує та аналізує візуальну інформацію за допомогою камер, аналого-цифрового перетворення та цифрової обробки сигналів. Хоча машинний зір часто порівнюють з людським зором, він не пов'язаний з біологією і може бути програмований, наприклад, для того, щоб бачити через стіни. Ця технологія використовується в різних сферах, від ідентифікації підписів до аналізу медичних зображень. Часто комп'ютерний зір, який фокусується на обробці зображень, плутають з машинним зором.

Обробка природної мови (NLP): Це обробка людської мови за допомогою комп'ютерних програм. Один із найстаріших і найвідоміших прикладів NLP - це виявлення спаму, коли алгоритм аналізує тему та текст електронних листів і визначає, чи є вони небажаними. Сучасні підходи до NLP базуються на машинному навчанні. Завдання в області NLP включають переклад тексту, аналіз настроїв та розпізнавання мови.

Хоча інструменти штучного інтелекту відкривають нові можливості для бізнесу, їх використання також викликає етичні питання. Системи штучного інтелекту можуть підсилити те, що вони вже вивчили, що може створити проблеми. Алгоритми машинного навчання, які лежать в основі багатьох з таких інструментів, виявляються такими розумними, наскільки розумні дані, які використовуються під час навчання. Оскільки людина визначає, які дані використовуватимуться для навчання програми, упередженість машинного навчання може стати проблемою, яку потрібно ретельно контролювати.

Для тих, хто має намір використовувати машинне навчання в реальних виробничих системах, важливо враховувати етичні аспекти в їх процесах навчання штучного інтелекту та намагатися уникати упередженості. Це особливо суттєво при використанні алгоритмів штучного інтелекту, таких як глибоке навчання та генеративно-суперницькі мережі (GAN), які, за своєю природою, можуть бути надто складними для пояснення в рамках програм.

3.4. Інструменти та сервіси ШІ

Інструменти та сервіси штучного інтелекту швидко розвиваються, і поточні інновації можна прослідкувати до важливих подій, таких як запуск нейромережі AlexNet у 2012 році. Це відкрило нову еру високопродуктивного штучного інтелекту, що базується на графічних процесорах та великих обсягах даних. Однією з ключових змін стало те, що тепер можна навчати нейронні мережі на величезних обсягах даних паралельно на кількох ядрах графічного процесора, що дозволило масштабувати процес тренування більш ефективно.

За останні кілька років співпраця між компаніями Google, Microsoft та OpenAI з апаратними інноваціями від Nvidia дозволила використовувати все більші моделі штучного інтелекту на більшій кількості графічних процесорів, що сприяло суттєвому підвищенню продуктивності та масштабованості.

Співпраця між цими провідними компаніями у сфері штучного інтелекту була критичною для успіху нещодавніх інновацій, таких як ChatGPT та інші проривні сервіси штучного інтелекту. Ось короткий огляд важливих інновацій в інструментах та сервісах штучного інтелекту.

Наприклад, Google провів передові дослідження для знаходження більш ефективного процесу навчання штучного інтелекту на великих кластерах звичайних ПК з графічними процесорами. Це привело до відкриття трансформерів, які автоматизують багато аспектів навчання штучного інтелекту на немаркованих даних.

Оптимізація апаратного забезпечення має велике значення для прогресу штучного інтелекту. Компанії, такі як Nvidia, працюють над вдосконаленням мікрокоду для паралельної обробки на кількох ядрах GPU для найпопулярніших алгоритмів. Це поєднання швидшого апаратного забезпечення, ефективних алгоритмів штучного інтелекту та точного налаштування інструкцій графічного процесора значно підвищує продуктивність штучного інтелекту. Крім того, Nvidia активно співпрацює з провайдерами хмарних сервісів, щоб зробити штучний інтелект доступним як AI-as-a-Service через моделі IaaS, SaaS і PaaS.

Також варто відзначити розвиток генеративних попередньо навчених трансформерів (GPT). Раніше компаніям доводилося навчати свої моделі ШІ з нуля, що вимагало значних витрат часу, грошей і досвіду. Однак тепер більше постачальників, таких як OpenAI, Nvidia, Microsoft, Google та інші, надають генеративні попередньо навчені трансформери, які можна точно налаштувати для конкретних завдань за значно менший час і кошти. Це дозволяє підприємствам прискорити вихід на ринок та знизити ризик вкладень.

Хмарні сервіси штучного інтелекту дійсно можуть допомогти підприємствам використовувати штучний інтелект у своєму бізнесі, але однією з найбільших перешкод є складнощі з обробкою даних і наукою про дані, які потрібні для інтеграції можливостей штучного інтелекту в нові програми або розробки нових. Проте провідні постачальники хмарних технологій пропонують вирішення для цих проблем, розгортаючи власний фірмовий штучний інтелект як пропозиції послуг. Найпопулярніші сервіси включають AWS AI Services, Google Cloud AI, Microsoft Azure AI, IBM AI і Oracle Cloud Infrastructure AI Services.

Крім того, сучасні моделі штучного інтелекту як послуга стають все більш доступними завдяки провідним розробникам. Наприклад, OpenAI пропонує десятки великих мовних моделей, оптимізованих для чатів, обробки природної мови, створення зображень і коду, доступні через Azure. Nvidia, в свою чергу, продає інфраструктуру штучного інтелекту та базові моделі, оптимізовані для тексту, зображень і медичних даних, які доступні у всіх провайдерів хмарних послуг. Багато інших гравців також пропонують моделі, адаптовані для різних галузей та випадків використання. Такі рішення дозволяють підприємствам швидше та ефективніше впроваджувати розробки у сфері штучного інтелекту.

3.5. Сфери застосування ШІ

Штучний інтелект проклав собі дорогу на різноманітні ринки. Є безліч сфер застосування. Приклади наведені нижче.

*ШІ в охороні здоров'я* дійсно створює можливості для значного покращення результатів лікування пацієнтів і оптимізації витрат в цій сфері. Однією з важливих технологій є IBM Watson, який використовує машинне навчання для кращого та швидшого ставлення медичних діагнозів. Ця система може аналізувати дані пацієнтів та інші медичні джерела, щоб формувати гіпотези та оцінювати їх достовірність.

У додаток до цього, в охороні здоров'я широко використовуються віртуальні онлайн-асистенти і чат-боти, які допомагають пацієнтам і клієнтам медичних закладів отримувати доступ до потрібної медичної інформації, записуватися на прийоми, розуміти процес виставлення рахунків та виконувати інші адміністративні процедури.

Аналітичні моделі та алгоритми машинного навчання допомагають управлінцям та медичним фахівцям приймати інформовані рішення та реагувати на ситуацію ефективніше.

*ШІ в освіті.* Використання штучного інтелекту в освіті відкриває безліч можливостей для покращення навчального процесу. Однією з основних переваг є автоматизація виставлення оцінок, що дозволяє викладачам сконцентруватися на інших аспектах своєї роботи. ШІ може також індивідуалізувати навчання, адаптуючись до потреб кожного учня і допомагаючи їм просуватися у навчанні у власному темпі.

Репетитори штучного інтелекту можуть забезпечити додаткову підтримку для студентів, допомагаючи їм зберігати правильний шлях навчання. Також важливо відзначити, що технологія може змінити умови навчання, відкриваючи нові можливості для студентів та викладачів.

Генеративні моделі штучного інтелекту, такі як ChatGPT і Google Bard, можуть допомогти викладачам створювати різноманітні навчальні матеріали та зацікавлювати студентів новими методами. Поява таких інструментів також змушує викладачів переглянути домашні завдання та тести, а також політику щодо плагіату, що сприяє покращенню якості освіти.

*ШІ у фінансах.* Використання штучного інтелекту в додатках для персональних фінансів дійсно змінює фінансову індустрію, прискорюючи і полегшуючи процеси для користувачів. Додатки, як Intuit Mint та TurboTax, допомагають людям керувати своїми фінансами, збираючи і аналізуючи особисті дані та надаючи фінансові консультації. Це робить управління фінансами більш ефективним та зручним для користувачів, але може також створювати конкуренцію для традиційних фінансових установ.

Наприклад, деякі програми, такі як IBM Watson, застосовуються до процесу купівлі житла, допомагаючи користувачам приймати кращі фінансові рішення. Використання штучного інтелекту у таких сферах може покращити якість життя користувачів, допомагаючи їм зробити більш обґрунтовані інвестиції та фінансові рішення.

Щодо торгівлі на Уолл-стріт, штучний інтелект дійсно став важливим інструментом для аналізу даних та прийняття фінансових рішень. Використання алгоритмів штучного інтелекту дозволяє здійснювати швидкі та точні торгові операції на фондовому ринку. Однак це також вимагає уважності та регуляції, оскільки велика кількість автоматизованих торгівельних алгоритмів може призвести до нестабільності на ринку.

*ШІ в законодавстві.* Використання штучного інтелекту в юридичній галузі дійсно революціонізує процеси та підвищує ефективність обслуговування клієнтів. Ось кілька способів, якими це відбувається:

Машинне навчання для аналізу даних і прогнозування результатів:

* юридичні фірми можуть використовувати алгоритми машинного навчання для аналізу великих обсягів даних, щоб зрозуміти тенденції, прогнозувати результати справ і робити рекомендації клієнтам;
* комп'ютерний зір для класифікації інформації з документів: Системи комп'ютерного зору можуть автоматично аналізувати та класифікувати документи, швидко знаходячи потрібну інформацію і виконуючи завдання, які раніше потребували б великої кількості людських ресурсів;
* обробка природної мови для інтерпретації запитів інформації: Технології обробки природної мови дозволяють системам розуміти та відповідати на запити клієнтів, а також автоматично аналізувати текст юридичних документів, що дозволяє швидше знаходити потрібну інформацію.

Ці інструменти значно спрощують та прискорюють рутинні завдання юридичних фахівців, дозволяючи їм сконцентруватися на більш складних аспектах своєї роботи та надавати кращий сервіс клієнтам.

*ШІ в розвагах і ЗМІ.* Використання штучного інтелекту, зокрема генеративних моделей, в розважальній індустрії може дійсно забезпечити значний вплив і покращення в різних аспектах. Ось деякі можливості та обмеження використання ChatGPT та інших генеративних моделей в розважальному бізнесі:

* цільова реклама та рекомендації контенту: ШІ може аналізувати дані користувачів та їхній вміст, щоб створювати більш персоналізовані рекламні пропозиції та рекомендації контенту;
* розповсюдження та виявлення шахрайства: Алгоритми ШІ можуть допомагати автоматично виявляти підроблені або шахрайські вміст та діяти в якості фільтрації перед розповсюдженням;
* створення сценаріїв і фільмів: ШІ може бути використаний для генерації ідей, сценаріїв або навіть виробництва фільмів на основі певних параметрів або шаблонів;
* автоматизація журналістики: Моделі ШІ можуть допомагати у створенні новинних матеріалів, відгуків або статей, а також у виборі заголовків і вмісту на основі поданих даних або параметрів.

Насправді, потенціал використання генеративного штучного інтелекту у розважальній індустрії досить великий. Однак важливо враховувати етичні питання, точність і достовірність генерованого контенту, а також збереження людського творчого внеску та контролю.

*ШІ в банківській справі.*Використання штучного інтелекту в банківській сфері дійсно відкриває широкі можливості для покращення ефективності та забезпечення кращого обслуговування клієнтів. Ось деякі способи, які банки використовують ШІ:

* чат-боти: ШІ дозволяє банкам створювати чат-ботів для автоматизованої комунікації з клієнтами, що спрощує процес обслуговування, надає швидку підтримку та допомагає з різноманітними фінансовими питаннями;
* віртуальні помічники: ШІ може бути використаний для створення віртуальних помічників, які надають персоналізовану консультацію клієнтам, допомагають у виборі продуктів та послуг, а також у вирішенні питань щодо банківських операцій;
* покращений аналіз ризиків: ШІ допомагає банкам аналізувати ризики та оцінювати кредитний потенціал клієнтів за допомогою алгоритмів машинного навчання, що дозволяє зробити більш точні й швидкі рішення щодо кредитування;
* автоматизація процесів: ШІ допомагає автоматизувати багато рутинних процесів в банківській сфері, таких як обробка транзакцій, визначення кредитних лімітів, виявлення шахрайства та інші операції, що допомагає зменшити витрати та покращити ефективність.

Загалом, використання штучного інтелекту дозволяє банкам підвищити рівень обслуговування клієнтів, зменшити витрати та покращити ризиковий менеджмент.

*ШІ в транспорті.* Штучний інтелект дійсно відіграє ключову роль у транспортній і логістичній галузях, сприяючи покращенню безпеки, ефективності та надійності систем та процесів. Ось деякі способи, які ШІ використовується в транспорті:

* управління трафіком: ШІ використовується для оптимізації руху транспортних засобів на дорогах та в повітряному просторі, зменшуючи затори і покращуючи потоки транспорту;
* прогнозування затримок рейсів: ШІ аналізує великі обсяги даних, щоб прогнозувати можливі затримки рейсів, що дозволяє авіакомпаніям та іншим перевізникам планувати розклади та вживати заходів для мінімізації негативних впливів;
* безпека морського судноплавства: ШІ допомагає виявляти потенційні загрози безпеці та уникати аварій шляхом аналізу даних про рух суден, погодних умов та інших факторів;
* оптимізація ланцюга постачання: ШІ допомагає компаніям прогнозувати попит на товари, оптимізувати запаси та маршрути доставки, що дозволяє підприємствам знижувати витрати та підвищувати ефективність управління логістикою.

Загалом, використання штучного інтелекту дозволяє транспортним компаніям та логістичним операторам підвищити продуктивність, покращити якість обслуговування та забезпечити безпеку та надійність у транспортних операціях.

*ШІ на виробництві.*Виробничі підприємства активно використовують штучний інтелект і роботизовані системи для автоматизації та оптимізації виробничих процесів. Ось кілька способів, якими це відбувається:

* коботи (коллаборативні роботи) співпрацюють з людьми у виробничих середовищах, допомагаючи виконувати завдання, які вимагають спільної роботи людей та роботів. Вони можуть бути програмовані на виконання різних завдань, включаючи збірку, пакування та обробку матеріалів;
* прогнозування та планування виробництва: Штучний інтелект використовується для аналізу великих обсягів даних та прогнозування попиту на продукцію, що дозволяє виробникам оптимізувати процеси виробництва та планувати виробничі потоки;
* моніторинг та обслуговування обладнання: ШІ допомагає відслідковувати стан обладнання та прогнозувати можливі поломки, що дозволяє підприємствам зменшувати час простою обладнання та знижувати витрати на обслуговування:
* якість контролю: Системи штучного інтелекту можуть виявляти дефекти на продукції шляхом аналізу зображень або даних датчиків, що дозволяє вчасно вживати заходів для виправлення проблем та забезпечення якості продукції.

Загалом, штучний інтелект дозволяє виробничим підприємствам підвищувати продуктивність, знижувати витрати та покращувати якість продукції, роблячи виробничі процеси більш ефективними та конкурентоспроможними.

*ШІ в бізнесі.* Штучний інтелект та машинне навчання дійсно трансформують способи, якими компанії взаємодіють зі своїми клієнтами і ведуть бізнес. Ось деякі зміни, які вони приносять:

* персоналізовані послуги. Алгоритми машинного навчання допомагають компаніям аналізувати великі обсяги даних про клієнтів та їхню поведінку, щоб зрозуміти їхні потреби та вподобання. Це дозволяє створювати персоналізовані пропозиції та послуги, які краще відповідають потребам кожного клієнта;
* автоматизована підтримка клієнтів. Чат-боти, які використовують штучний інтелект, можуть надавати миттєві відповіді на запитання клієнтів та вирішувати їхні проблеми, зменшуючи навантаження на людський персонал та підвищуючи швидкість обслуговування;
* прогнозування та аналіз. Алгоритми машинного навчання допомагають прогнозувати та аналізувати покупкову поведінку клієнтів, що дозволяє компаніям адаптувати свої стратегії маркетингу та продажів для досягнення кращих результатів;
* зміна бізнес-моделей. ШІ може стати каталізатором для перетворення бізнес-моделей, створюючи нові можливості та конкурентні переваги. Наприклад, компанії можуть переходити до моделей підписки, використовуючи аналітику для прогнозування попиту та пропозиції;
* революція в дизайні продуктів. Застосування генеративних технологій штучного інтелекту може допомогти в розробці нових продуктів та послуг, забезпечуючи більшу інноваційність та відповідність потребам клієнтів.
* Відповіді на запитання та розгляд скарг відразу після їх надходження (Підтримка клієнтів). За словами Gartner, до 2021 року 25% компаній у всьому світі матимуть віртуального помічника для вирішення питань підтримки [12].

*Безпека.* ШІ і машинне навчання мають великий потенціал у сфері кібербезпеки. Деякі з головних переваг цих технологій у цій галузі включають:

* виявлення аномалій. Алгоритми машинного навчання можуть аналізувати великі обсяги даних і виявляти аномальні зразки, що можуть вказувати на потенційні загрози або атаки;
* швидке реагування. Штучний інтелект може автоматично реагувати на виявлені загрози, надаючи швидку відповідь і запобігаючи подальшому поширенню атак;
* автоматизоване виявлення нових загроз. Штучний інтелект може навчитися виявляти нові типи загроз та атак, навіть якщо вони раніше були невідомі, шляхом аналізу та виявлення загальних патернів у поведінці зловмисників;
* мінімізація помилок. Автоматизовані системи, що використовують штучний інтелект, можуть значно знизити кількість помилкових спрацьовувань, що дозволяє ефективніше виявляти загрози та зменшувати кількість хибнопозитивних сповіщень;
* постійне навчання і вдосконалення. Машинне навчання може неперервно вдосконалювати свої алгоритми на основі нових даних та досвіду, що допомагає виявляти й аналізувати нові загрози з часом.

*ШІ в програмуванні та ІТ-процесах.* Нові генеративні інструменти штучного інтелекту відкривають нові можливості для автоматизації процесу розробки програмного забезпечення. Вони дозволяють розробникам створювати код за допомогою природної мови або навіть лише шляхом надання підказок. Однак наразі ці інструменти все ще перебувають на етапі розвитку, і їхнє використання ще не поширюється масово.

Нові генеративні інструменти можуть полегшити та прискорити процес розробки, але вони навряд чи замінять розробників програмного забезпечення повністю. Розробники все ще потрібні для створення та налагодження складних програмних продуктів, а також для вирішення непередбачених проблем і впровадження нових ідей.

Щодо інших ІТ-процесів, штучний інтелект і машинне навчання вже успішно використовуються для автоматизації багатьох завдань, включаючи введення даних, виявлення шахрайства, обслуговування клієнтів та прогнозне обслуговування. Вони допомагають підвищити ефективність та точність цих процесів, а також зменшити витрати і ризики.

3.6. Засоби штучного інтелекту для веб-розробки

У застосуванні до веб-розробки ШІ може аналізувати дані, робити прогнози та ефективно автоматизувати різні процеси. Це може значно покращити продуктивність веб-сайту, залучення користувачів і загальні результати бізнесу.

Штучний інтелект у веб-розробці виходить за межі простої автоматизації. Це передбачає інтеграцію машинного навчання, обробки природної мови та інших технологій штучного інтелекту для створення розумніших, інтуїтивно зрозуміліших веб-сайтів і веб-додатків. Ці алгоритми штучного інтелекту можуть обробляти величезні обсяги даних, щоб генерувати інформацію, приймати рішення в реальному часі та адаптуватися до вподобань користувачів.

Однією з ключових ролей штучного інтелекту в сучасній веб-розробці є забезпечення персоналізованого досвіду. Аналізуючи поведінку користувачів, алгоритми ШІ можуть рекомендувати відповідний контент, продукти та послуги. Це не тільки покращує залучення користувачів, але й підвищує рівень конверсії та задоволеність клієнтів. ШІ робить можливим те, що під час відвідування інтернет-магазину користувач отримуватиме персоналізовані рекомендації щодо продуктів на основі історії веб-перегляду та вподобань.

ШІ відіграє ключову роль у стимулюванні інновацій у веб-розробці. Завдяки чат-ботам і віртуальним помічникам на основі штучного інтелекту веб-сайти можуть надавати підтримку клієнтів 24/7, відповідаючи на запити та оперативно розв’язуючи проблеми. Ці чат-боти можуть розуміти природну мову та надавати інтелектуальні відповіді, створюючи безперебійне та ефективне обслуговування клієнтів. Вони можуть опрацьовувати кілька взаємодій з клієнтами одночасно, заощаджуючи час і ресурси для компаній.

ШІ забезпечує розширений аналіз даних у веб-розробці. Використовуючи алгоритми машинного навчання, веб-сайти можуть обробляти й аналізувати величезні обсяги даних, щоб отримати цінну інформацію. Наприклад, веб-сайт електронної комерції може використовувати штучний інтелект, щоб аналізувати моделі поведінки клієнтів і робити точні прогнози щодо майбутніх покупок. Потім цю інформацію можна використовувати для оптимізації маркетингових стратегій, покращення рекомендацій щодо продукту та підвищення загальної ефективності бізнесу.

Крім того, штучний інтелект у веб-розробці може автоматизувати різні процеси, зменшуючи потребу в ручному втручанні. Наприклад, алгоритми штучного інтелекту можуть автоматично класифікувати вміст і позначати його тегами, полегшуючи організацію та пошук інформації на веб-сайті. Вони також можуть автоматизувати повторювані завдання, такі як введення даних, звільняючи час для розробників, щоб зосередитися на більш складних і творчих аспектах веб-розробки.

Аналізуючи дані користувачів і моделі поведінки, ШI може надавати персоналізовані рекомендації, персоналізований контент і інтуїтивно зрозумілу навігацію. Це забезпечує більш привабливий і релевантний досвід користувача. Алгоритми штучного інтелекту також можуть оптимізувати час завантаження сторінки, підвищуючи ефективність веб-сайту та знижуючи показники відмов.

Наприклад, уявіть, що користувач відвідує веб-сайт електронної комерції та шукає певний продукт. Завдяки інтеграції штучного інтелекту веб-сайт може аналізувати історію пошуку, історію покупок і налаштування користувача, щоб надавати персоналізовані рекомендації щодо продуктів. Це не тільки покращує досвід користувача, заощаджуючи його час і зусилля на пошук відповідних продуктів, але також збільшує шанси конверсії для власника веб-сайту.

Крім того, штучний інтелект може аналізувати поведінку користувачів на веб-сайті, наприклад шаблони кліків і час, витрачений на кожну сторінку, щоб визначати сфери вдосконалення. Ці цінні дані можуть бути використані для оптимізації макета веб-сайту, розміщення вмісту та кнопок із закликом до дії, що призведе до більш інтуїтивно зрозумілого та зручного інтерфейсу.

3.6.1. Процеси веб-розробки за допомогою ШІ

Автоматизація є ключовим аспектом ШІ у веб-розробці. ШІ може автоматизувати повторювані завдання: керування вмістом, тестування та виправлення помилок. Це дає можливість розробникам зосередитися на більш складних і творчих аспектах веб-розробки, збільшуючи продуктивність і прискорюючи час виходу на ринок.

Розглянемо процес управління контентом. Завдяки інтеграції AI веб-розробники можуть використовувати алгоритми обробки природної мови для автоматичного створення метатегів, заголовків і описів для веб-сторінок. Це не тільки економить час, але й забезпечує послідовний і зручний для SEO вміст на веб-сайті.

Крім того, AI може допомогти в тестуванні та виправленні помилок. Замість того, щоб вручну перевіряти кожну функцію та виправляти помилки, алгоритми штучного інтелекту можуть аналізувати код і виявляти потенційні проблеми, скорочуючи час і зусилля, необхідні для забезпечення якості.

Також AI може сприяти співпраці між командами веб-розробників. Аналізуючи сильні та слабкі сторони окремих членів команди, алгоритми ШІ можуть призначати завдання та розподіляти ресурси більш ефективно. Це дає можливість покращити командну роботу та управління проектами, що зрештою приведе до кращих результатів веб-розробки.

3.6.2. Приклади інтеграції технологій ШІ з веб-розробкою

Кілька технологій ШІ є ключовими у веб-розробці. Розглянемо дві основні: чат-боти та віртуальні помічники зі штучним інтелектом, а також штучний інтелект для веб-дизайну та верстки.

Чат-боти та віртуальні помічники на основі штучного інтелекту забезпечують підтримку клієнтів у режимі реального часу, відповідають на поширені запитання та допомагають користувачам у навігації веб-сайтами чи програмами. Ці інтелектуальні агенти розроблені, щоб розуміти та імітувати людську мову, що дозволяє їм брати участь у природних і змістовних розмовах з користувачами.

Завдяки розширеним можливостям обробки природної мови чат-боти штучного інтелекту можуть точно інтерпретувати запити користувачів і надавати відповідні відповіді. Вони можуть обробляти широкий спектр запитів клієнтів, від базової інформації до складних сценаріїв усунення несправностей. Автоматизуючи завдання підтримки клієнтів, чат-боти можуть значно скоротити час відповіді та підвищити рівень задоволеності клієнтів.

З іншого боку, віртуальні помічники виходять за рамки підтримки клієнтів і можуть виконувати різноманітні завдання. Вони можуть призначати зустрічі, давати рекомендації та навіть виконувати онлайн-транзакції від імені користувачів. Ці віртуальні помічники використовують такі технології штучного інтелекту, як машинне навчання та розпізнавання мовлення, щоб розуміти команди користувача та відповідним чином виконувати дії.

Завдяки інтеграції чат-ботів зі штучним інтелектом і віртуальних помічників у веб-сайти або додатки компанії можуть забезпечити цілодобову підтримку, покращити взаємодію з користувачами та зменшити витрати на підтримку. Ці інтелектуальні агенти стали безцінними інструментами для посилення взаємодії з клієнтами та оптимізації онлайн-взаємодії.



Рисунок – 3.2 Приклад чат-бота на офіційному веб-сайті Київстар

Отже, для додавання чат-бота на веб-сайт можна скористатися різними сервісами та інструментами, такими як Dialogflow, Chatfuel, ManyChat або власноруч написати свого чат-бота з використанням JavaScript та WebSocket або HTTP-запитів. Приклад коду для додавання простого чат-бота на веб-сайт за допомогою JavaScript та WebSocket наведений у додатку  А.

Цей код –простий приклад інтеграції чат-бота на веб-сайт. Залежно від потреб та вимог, можна розширити його функціонал та додати більше можливостей.

Алгоритми штучного інтелекту змінюють спосіб розробки та оформлення веб-сайтів. Аналізуючи вподобання користувачів, візуальні стилі та поточні тенденції дизайну, ШІ допомагає створювати естетично привабливі та зручні веб-дизайни.

Традиційно веб-дизайнери витрачають значну кількість часу та зусиль на створення візуально приголомшливих веб-сайтів. Їм потрібно враховувати такі фактори, як колірні схеми, типографіка та макет, щоб забезпечити привабливість і залучення користувачів. Однак за допомогою штучного інтелекту для веб-дизайну цей процес можна автоматизувати, заощаджуючи дизайнерам дорогоцінний час і ресурси.

Алгоритми штучного інтелекту можуть аналізувати величезні обсяги даних, до яких належать шаблони поведінки користувачів і тенденції дизайну, щоб генерувати рекомендації щодо дизайну. Ці рекомендації можуть варіюватися від кольорових палітр і вибору шрифтів до розташування елементів на веб-сторінці. Використовуючи штучний інтелект, дизайнери можуть швидко досліджувати кілька варіантів дизайну та повторювати їх, зрештою досягаючи найефективнішого та візуально привабливого дизайну.

Крім того, AI також може допомогти оптимізувати веб-дизайн для різних пристроїв і розмірів екрану. З розповсюдженням мобільних пристроїв дуже важливо, щоб веб-сайти були адаптивними та адаптованими. Алгоритми штучного інтелекту можуть аналізувати вміст і структуру веб-сайту та автоматично коригувати макет для забезпечення оптимального перегляду на різних пристроях.

Загалом штучний інтелект для веб-дизайну та верстки пропонує величезний потенціал для оптимізації процесу дизайну, покращення взаємодії з користувачем і створення візуально приголомшливих веб-сайтів. Використовуючи можливості штучного інтелекту, компанії можуть залишатися попереду в конкурентному цифровому середовищі та надавати своїм користувачам винятковий веб-досвід.

3.6.3. Компоненти упровадження ШІ у веб-розробку

Впровадження ШІ у веб-розробку вимагає ретельного планування та точного виконання. Ось основні кроки для успішної інтеграції ШІ у проекти веб-розробки:

1. Планування та розробка стратегії інтеграції ШІ. Перш ніж занурюватися в впровадження штучного інтелекту, дуже важливо визначити конкретні випадки використання та цілі штучного інтелекту у проектах веб-розробки. Необхідно визначити, як штучний інтелект може покращити взаємодію з користувачем, покращити бізнес-операції або вирішити конкретні завдання. Створити комплексну стратегію інтеграції ШІ, яка відповідає загальній дорожній карті веб-розробки.

На етапі планування важливо враховувати потенційний вплив ШІ на проекти веб-розробки. Необхідно провести ретельне дослідження ринку, щоб зрозуміти поточні тенденції та досягнення в технології ШІ. Це допоможе вам залишатися попереду конкурентів і переконатися, що інтеграція ШІ буде інноваційною та передовою.

Крім того, важливо оцінити доцільність впровадження ШІ у проекти веб-розробки. Треба врахувати такі фактори, як бюджет, ресурси та технічний досвід.

2. Вибір правильних інструментів і платформ ШІ. Від готових платформ штучного інтелекту до спеціальної розробки штучного інтелекту – доступні різні варіанти, коли справа доходить до вибору інструментів штучного інтелекту для веб-розробки. Необхідно дослідити та оцінити різні інструменти та платформи ШІ на основі вимог проекту, масштабованості та простоти інтеграції.

Вибираючи інструменти та платформи штучного інтелекту, необхідно врахувати конкретні функції та функції, які є важливими для проектів веб-розробки. Наприклад, якщо створювати веб-сайт електронної комерції, можуть знадобитися інструменти штучного інтелекту, які можуть надавати персоналізовані рекомендації щодо продуктів на основі вподобань користувача та історії веб-перегляду.

Крім того, треба врахувати сумісність інструментів і платформ ШІ з існуючою інфраструктурою веб-розробки. Інтеграція має бути безперебійною та ефективною, зводячи до мінімуму будь-які збої в робочому процесі веб-розробки.

Крім того, важливо оцінити масштабованість і гнучкість інструментів і платформ ШІ. У міру того, як проекти веб-розробки зростають, інтеграція штучного інтелекту повинна мати можливість пристосуватися до збільшення обробки даних і взаємодії з користувачем.

В решті решт треба розглянути  довгострокове технічне обслуговування та підтримку інструментів і платформ ШІ. Переконайтеся, що є спеціальна команда підтримки або спільнота, яка може допомогти з будь-якими технічними проблемами або оновленнями.

3.6.4. Конфіденційність даних

ШІ значною мірою використовує дані. Тому забезпечення конфіденційності та безпеки даних є надзвичайно важливим. Важливо застосовувати надійні заходи захисту даних, дотримуватись правил конфіденційності та отримувати згоду користувачів на збір і використання їхніх даних. Прозорість у використанні даних створює довіру та підвищує впевненість користувачів.

Однією з головних проблем веб-розробки штучного інтелекту є потенційне зловживання або неправильне поводження з даними користувачів. Оскільки алгоритми штучного інтелекту постійно навчаються та адаптуються, стає надзвичайно важливим захистити конфіденційність людей, чиї дані використовуються. Розробники повинні застосовувати надійні методи шифрування, безпечні системи зберігання та контроль доступу, щоб захистити конфіденційну інформацію.

Крім того, дотримання правил конфіденційності, таких як Загальний регламент захисту даних (GDPR), є важливим. Це означає, що користувачі мають контроль над своїми даними та знають, як вони використовуватимуться. Отримання чіткої згоди від користувачів перед збором і використанням їхніх даних є важливим кроком у зміцненні довіри та підтримці прозорості.

3.6.5. Прогнозовані тенденції та інновації застосування ШІ у веб-розробці

Майбутнє ШІ у веб-розробці виглядає багатообіцяючим. Швидкий розвиток технологій AI та їх поширення в галузях продовжуватимуть формувати практику веб-розробки. Штучний інтелект вже досяг значних успіхів у різних сферах, і веб-розробка не є винятком. Завдяки здатності аналізувати величезні масиви даних, вчитися на шаблонах і приймати розумні рішення, штучний інтелект революціонізує спосіб створення та використання веб-сайтів і програм.

Алгоритми на основі штучного інтелекту стають дедалі складнішими, дозволяючи веб-розробникам створювати інтелектуальні системи рекомендацій, які персоналізують вміст на основі вподобань і поведінки користувачів. Ці механізми рекомендацій не лише покращують взаємодію з користувачем, але й стимулюють залучення та збільшують кількість конверсій.

Ще одна захоплююча тенденція у веб-розробці, керованій штучним інтелектом, — це зростання голосових інтерфейсів. З популярністю таких віртуальних помічників, як Siri, Alexa та Google Assistant, голосові команди стають кращим методом взаємодії. Технології штучного інтелекту дозволяють веб-сайтам і програмам розуміти голосові команди та реагувати на них, забезпечуючи користувачам інтуїтивно зрозумілу роботу без рук.

Гіперперсоналізація – ще одна прогнозована тенденція веб-розробки, керованої ШІ. Використовуючи алгоритми штучного інтелекту, веб-сайти можуть адаптувати вміст, продукти та рекомендації для окремих користувачів, створюючи персоналізований і релевантний досвід. Цей рівень персоналізації не тільки покращує задоволеність користувачів, але й підвищує лояльність клієнтів і стимулює повторні операції.

Оскільки штучний інтелект продовжує розвиватися, ми можемо очікувати більш складні програми ШІ у веб-розробці. Інтелектуальні механізми рекомендацій, голосові інтерфейси та гіперперсоналізація – це деякі з прогнозованих тенденцій. Штучний інтелект забезпечить ще більший інтуїтивний і захоплюючий веб-досвід, кардинально змінивши спосіб взаємодії користувачів із веб-сайтами та програмами.

Також очікується, що ШІ відіграватиме значну роль в автоматизації повторюваних і трудомістких завдань у веб-розробці. Від автоматизації веб-дизайну до генерації фрагментів коду технології ШІ можуть оптимізувати процес розробки, дозволяючи розробникам зосередитися на більш складних і творчих аспектах своєї роботи.

Крім того, чат-боти на основі штучного інтелекту стають все більш популярними у веб-розробці. Ці інтелектуальні віртуальні помічники можуть надавати миттєву підтримку клієнтів, відповідати на поширені запитання та навіть обробляти транзакції. Завдяки інтеграції чат-ботів штучного інтелекту у веб-сайти компанії можуть покращити обслуговування клієнтів, покращити час відповіді та зменшити операційні витрати.

Висновки до третього розділу

ШІ став невід’ємною частиною веб-розробки, революціонізувавши спосіб створення та керування веб-сайтами та веб-додатками. Штучний інтелект приносить численні переваги веб-розробці: від персоналізованого досвіду до передового аналізу даних і автоматизації. Оскільки технології продовжують розвиватися, ми можемо очікувати, що ШІ відіграватиме ще більшу роль у формуванні майбутнього веб-розробки.

Щоб залишатися попереду в майбутньому веб-розробки на основі штучного інтелекту, компаніям слід почати вивчати стратегії інтеграції штучного інтелекту та надати пріоритет ухваленню рішень на основі даних. Впровадження штучного інтелекту в веб-розробку вимагає глибокого розуміння базових технологій та їх потенційного застосування. Важливо збирати й аналізувати відповідні дані для ефективного навчання моделей ШІ.

Інтеграція штучного інтелекту у веб-розробку пропонує численні переваги. Штучний інтелект може революціонізувати спосіб проектування та розробки веб-сайтів: від покращення взаємодії з користувачами за допомогою персоналізованих рекомендацій і оптимізації продуктивності до оптимізації процесів веб-розробки за допомогою автоматизації та покращеної співпраці. Застосування технології AI може дати веб-сайту конкурентну перевагу в сучасному цифровому середовищі.

Використовуючи аналітику даних і машинне навчання, компанії можуть отримати цінну інформацію про поведінку користувачів, уподобання та тенденції, дозволяючи їм приймати обґрунтовані рішення та надавати персоналізований досвід.

Крім того, компанії повинні інвестувати в безперервне навчання та підвищення кваліфікації, щоб не відставати від розвитку ШІ. Веб-розробники та дизайнери повинні бути в курсі найновіших інструментів ШІ, фреймворків і передових практик, щоб використовувати весь потенціал ШІ у веб-розробці.

Розділ 4

ТРЕНДИ UI/UX ДИЗАЙНУ

В сучасному веб-дизайні велика увага приділяється використанню веб-анімації, яка визначається як використання рухливих зображень та елементів для покращення візуальної привабливості веб-сайтів. Веб-анімація не лише стала невід'ємною частиною сучасного веб-дизайну, але й здатна відігравати ключову роль у створенні успішного та інтерактивного інтернет-сайту. Є декілька тенденцій що зроблять сайт успішним в 2024 році [2].

4.1. Динамічні мікроанімації

Мікроанімація – це короткі рухи, або зміни, які тривають від мілісекунд до секунд. Вони додають до елементів інтерфейсу користувача тонкий рух і візуальний зворотний зв’язок, покращуючи взаємодію та підвищуючи залучення користувачів. На відміну від своїх яскравих аналогів, вони прагнуть до безперебійної інтеграції, безшумно працюючи над покращенням шляху користувача.

Мікроанімацією є використання рухливих зображень, переходів та інших анімаційних ефектів для збільшення візуальної привабливості веб-сайтів. Мікроанімація має значний вплив на перше враження від веб-сайту [16]. Завдяки елементам мікроанімації користувачі отримують більше можливостей для взаємодії, що поліпшує їхні враження та забезпечує запам'ятовуваність веб-сайту. Це можуть бути анімовані кнопки, реакції на введення чи виразні переходи між сторінками. Мікроанімації створюють враження живої та дбайливо продуманої інтерфейсної взаємодії [2].

Хоча мікроанімація може бути потужним інструментом, її надмірне використання може мати негативні наслідки.

Використовувати мікроанімацію краще помірно. Для цього є кілька правил:

* кожна мікроанімація має служити чіткій меті. Треба уникати безпричинної анімації, яка відволікає або дратує користувачів;
* послідовність: необхідно установити узгоджений стиль анімації та час на веб-сайті. Різноманітні анімації можуть створити дискомфорт користувачу;
* пріоритет продуктивності: анімація повинна бути легкою і оптимізованою для всіх пристроїв, щоб уникнути сповільнення часу завантаження сторінки;
* доступність має значення: необхідно реалізувати анімацію, яка враховує користувачів з обмеженими можливостями, наприклад надаючи альтернативні текстові описи для тих, хто використовує програми зчитування з екрана.

Для оживлення мікроанімації використовують декілька підходів:

* CSS-анімація: параметри  для базової анімації повинні бути легкі та універсальні. Вони ідеально підходять для базових мікровзаємодій, як-от наведення кнопки та переходи;
* бібліотеки JavaScript: забезпечують розширений контроль і складні ефекти анімації. Можете використовувати такі бібліотеки, як GSAP або anime.js, для більш складних і гнучких анімацій;
* SVG-анімація: можна увімкнути плавну векторну анімацію;
* спеціальні інструменти анімації: для спрощення процесу анімації використовують зручні інтерфейси;
* API веб-анімації: для передових ефектів і глибшого контролю над властивостями анімації.

Вибір правильного підходу залежить від потреб і ресурсів проекту. Враховуються такі фактори, як складність анімації, вимоги до продуктивності та досвід команди.

Окрім технічних аспектів, розглядається стратегічне розгортання мікроанімації для досягнення конкретних цілей:

* виділення ключових областей: мікроанімація використовується, щоб виділити важливу інформацію, як-от заклики до дії, функції продукту або важливі повідомлення.
* персоналізація досвіду: реалізовуються динамічні мікроанімації, які реагують на взаємодії користувача, створюючи більш привабливий і персоналізований досвід.
* розповідь історії: мікроанімацію можна використовувати, щоб тонко розповісти історію бренду, додавши емоційної глибини та запам’ятовуваності.
* підкреслення ієрархії: проведення користувачів через інформаційну ієрархію веб-сайту, візуально виділяючи розділи або елементи за допомогою мікроанімації.

Мікроанімації – це не просто візуальні прикраси; вони є стратегічними інструментами впливу на поведінку користувачів і досягнення конкретних цілей. Вони можуть направляти, інформувати, залучати та, зрештою, створювати чудові та незабутні враження від користувачів.

В додатку Б наведений приклад коду на CSS та JavaScript, який додаватиме мікроанімації для кнопки на веб-сайті при наведенні курсора миші. Цей приклад використовує CSS для визначення анімаційних ефектів, таких як зміна кольору тла та розміру, а також JavaScript для додавання анімації при наведенні миші.

Завдяки цьому коду кнопка матиме мікроанімації при наведенні курсора миші: змінить колір фону та збільшиться трохи у розмірі.

4.2. Технологія 3D

У сучасному веб-дизайні 3D-анімація стала не лише технологічним прогресом, але й потужним інструментом для створення переконливих і реалістичних зображень. Додаткова глибина і просторість 3D-анімації відкриває перед дизайнерами нові можливості для ефективної взаємодії з аудиторією.

Використання 3D-анімації у веб-дизайні розширює можливості візуального враження, надаючи сторінці глибину і об'єм. Об'єкти можуть виходити за межі екрану, а анімовані елементи виглядають як тривимірні моделі, що збільшує реалістичність та привабливість веб-сайту.

Використання 3D-анімації привносить новий рівень реалізму у веб-дизайн. Ця техніка дозволяє створювати анімовані переходи, які підсилюють реальний досвід. Це може використовуватися для підсилення бренду, створення враження інтерактивності та збільшення уваги користувачів до деталей.

Отже, щоб додати 3D-анімацію на веб-сайт, треба скористатися бібліотеками та фреймворками, такими як Three.js або WebGL, які надають можливості для реалізації 3D-графіки у веб-браузері. В додатку В наведено приклад коду для додавання 3D-об'єкта на веб-сайт за допомогою бібліотеки Three.js. Цей код створює 3D куб, який обертається на сцені.

4.3. Оригінальна типографіка

Оригінальна типографіка веб-дизайну розвивається завдяки новаторським технологіям та концепціям. Ось кілька ключових напрямів:

* використання нових веб-шрифтів та їх технологій розширює можливості дизайнерів у створенні різноманітних типографічних рішень;.
* CSS Grid та Flexbox відкривають ширші можливості для створення складних та адаптивних макетів, полегшуючи керування типографікою на різних пристроях;
* використання анімації покращує вигляд тексту, роблячи його більш привабливим та динамічним, що привертає увагу користувачів;
* Variable Fonts – це шрифти з різноманітними параметрами, які можуть змінюватися динамічно, що відкриває нові можливості для контролю типографії;
* темний режим та висококонтрастний дизайн стають все більш популярними, спрямовуючи увагу на зручність читання тексту для всіх користувачів;
* оптимізація веб-шрифтів допомагає покращити швидкість завантаження сторінок, забезпечуючи швидке відображення тексту;
* респонсивний дизайн гарантує легкість читання тексту на різних пристроях та екранах.

Всі ці новації спрямовані на покращення якості та зручності веб-типографіки для кращого користувацького досвіду.

4.4. Мобільна адаптація

Мобільна адаптація веб-сайтів полягає у вдосконаленні їхнього вигляду та функціональності для користувачів мобільних пристроїв, таких як смартфони та планшети. Основні аспекти цього підходу:

* респонсивний дизайн: це спосіб розробки веб-сайтів, який автоматично адаптує їхній макет до різних розмірів екранів, включаючи мобільні пристрої. Шляхом використання CSS та медіа-запитів, сайт може реагувати на розмір екрану та змінювати свій вигляд для оптимального відображення;
* мобільні версії сайтів: деякі веб-сайти створюють окремі версії спеціально для мобільних пристроїв, які можуть мати спрощений макет та обмежений функціонал порівняно з десктопною версією. Це сприяє забезпеченню зручного користувацького досвіду для мобільних користувачів;
* швидкість завантаження: оскільки мобільні пристрої мають обмежені ресурси, важливо, щоб веб-сторінки завантажувалися швидко. Це може бути досягнуто за допомогою оптимізації зображень, мінімізації файлів CSS та JavaScript, а також використання кешування;
* забезпечення користувацької зручності: важливо, щоб мобільні користувачі мали зручний та ефективний досвід використання веб-сайту. Це може включати просту навігацію, великі елементи керування, підготовлені для дотику, та оптимізовані форми для введення даних;
* SEO-оптимізація для мобільних пристроїв: пошукові системи, такі як Google, враховують мобільну дружність веб-сайтів при ранжуванні результатів пошуку для мобільних користувачів. Тому важливо оптимізувати мобільну версію сайту для покращення його видимості.

З ростом числа користувачів мобільних пристроїв для доступу до Інтернету, мобільна адаптація стає все більш важливою для забезпечення ефективного користувацького досвіду.

Інновації у мобільній адаптації веб-дизайну постійно еволюціонують, оскільки технології та потреби користувачів розвиваються. Ось деякі основні напрямки:

* прискорена мобільна адаптація: технології AMP (Accelerated Mobile Pages) сприяють швидкому завантаженню сторінок на мобільних пристроях, поліпшуючи користувацький досвід та SEO-показники;
* покращений респонсивний дизайн: нові методи та інструменти використовують для створення більш гнучких та ефективних респонсивних макетів, які автоматично адаптуються до різних розмірів екранів мобільних пристроїв;
* мобільні анімації та інтерактивність: використання анімаційних та інтерактивних ефектів збагачує користувацький досвід на мобільних пристроях, робить його більш захоплюючим та ефективним;
* інтеграція з мобільними функціями пристроїв: використання нових API та можливостей мобільних пристроїв, геолокації, акселерометра, використовується для створення більш інтерактивних та персоналізованих веб-додатків;
* оптимізація для голосового пошуку: зростаюча популярність голосових асистентів вимагає оптимізації веб-сайтів для голосового пошуку, щоб забезпечити кращу доступність та видимість;

Ці інновації допомагають поліпшити користувацький досвід та ефективність веб-сайтів на мобільних пристроях, що стає все важливішим у світі, де мобільний інтернет стає все доступнішим та зручнішим.

4.5. Голосовий інтерфейс користувача

За допомогою розвитку голосових технологій та віртуальних асистентів голосовий інтерфейс користувача (Voice User Interface, VUI) в веб-дизайні стає все більш популярним. Ось кілька основних аспектів цієї тенденції:

* Голосовий пошук та навігація: користувачі можуть здійснювати пошук та навігацію на веб-сайтах за допомогою голосових команд, що полегшує їх взаємодію, особливо на мобільних пристроях.
* Персоналізований досвід: голосові асистенти можуть надавати індивідуально налаштовані рекомендації та інформацію, враховуючи історію користувача, що забезпечує більш персоналізований підхід.
* Інтерактивність і контекстуальність: голосові інтерфейси можуть взаємодіяти з користувачем у реальному часі, задавати додаткові питання та адаптувати відповіді відповідно до контексту.
* Оптимізація для пошукових систем: веб-сайти можуть бути оптимізовані для відповіді на голосові запити, що забезпечує підвищену видимість у пошукових системах.
* Адаптація до мобільних пристроїв: голосові інтерфейси особливо зручні на мобільних пристроях, де введення тексту є важким.
* Безпека і конфіденційність: розробники повинні бути обережними щодо безпеки та конфіденційності даних користувачів голосових інтерфейсів.
* Голосовий інтерфейс користувача в веб-дизайні сприяє поліпшенню доступності та зручності взаємодії з веб-сайтами, відповідаючи сучасним технологічним тенденціям.

Додавання голосового інтерфейсу користувача на веб-сайт може бути реалізоване за допомогою використання різних API голосового розпізнавання та синтезу мови. Один з найпоширеніших сервісів – це Web Speech API, який надає можливість розпізнавання мови та синтезу голосу у браузері. Приклад, як використати Web Speech API для додавання голосового інтерфейсу на веб-сайт наведений в додатку Г.

Цей код реалізує кнопку «Start Dictation», яку користувач може натиснути, щоб активувати розпізнавання мови. Коли користувач говорить щось, текст його мовлення відображається у блоку з id «output».

Важливо зазначити, що Web Speech API не підтримується всіма браузерами, і може бути обмежена на деяких пристроях. Також, може знадобитися SSL-сертифікат для використання цієї функції на веб-сайті, оскільки браузери вимагають безпечного з'єднання для доступу до мікрофону користувача.

4.6. Персоналізація

Інновації в персоналізації веб-дизайну набувають популярності, оскільки вони сприяють створенню унікального та ефективного користувацького досвіду. Виокремлюють кілька ключових напрямків у процесі реалізації персоналізації.

1. Використання машинного навчання та аналізу даних. Алгоритми машинного навчання використовуються для аналізу поведінки користувачів для розуміння їхніх потреб та виборів. Це дає змогу створювати персоналізовані пропозиції та рекомендації.

2. Динамічний контент. Веб-сайти можуть автоматично змінювати свій зміст відповідно до профілю користувача, його інтересів та місця знаходження. Це може включати індивідуальні привітання, рекомендації товарів або статей.

3. Адаптивний дизайн і UX. Персоналізація може включати адаптивний дизайн, який змінюється відповідно до потреб користувача. Наприклад, веб-сайт може міняти макет чи розмір шрифту залежно від налаштувань користувача чи пристрою.

4. Персоналізовані пропозиції та акції. Веб-сайти можуть надавати користувачам індивідуальні знижки, акції або рекламу, враховуючи їхні попередні дії.

5. Аналіз поведінки користувачів в реальному часі. Важливо враховувати не лише історію користувача, але й його поточні дії на веб-сайті для надання більш якісного досвіду.

6. Сегментація аудиторії та персоналізовані сторінки. Розробники можуть створювати різні версії сторінок для різних сегментів аудиторії, враховуючи їхні інтереси та демографічні дані.

Ці інновації сприяють підвищенню залученості користувачів, покращенню конверсії та забезпеченню ефективнішої взаємодії з веб-сайтом.

Висновки до четвертого розділу

У сучасному веб-дизайні велика увага зосереджується на використанні веб-анімації, яка не лише стала необхідною складовою сучасного інтернет-сайту, але й виконує ключову роль у забезпеченні успішного та інтерактивного користувацького досвіду. Новаторські тенденції в цьому напрямку, такі як динамічні мікроанімації, визначають подальший розвиток веб-дизайну у 2024 році.

Мікроанімація, яка включає короткі рухи та зміни, відіграє ключову роль у підвищенні візуальної привабливості веб-сайтів та поліпшенні їхньої інтерфейсної взаємодії. Ця техніка дозволяє створювати тонкий рух і візуальний зворотний зв’язок, що поліпшує враження та ефективність взаємодії з веб-сайтом. Однак, для досягнення успіху, необхідно враховувати деякі правила, такі як чітка мета анімації, узгоджений стиль та пріоритет продуктивності.

Використання різних технік, таких як CSS-анімація, бібліотеки JavaScript, SVG-анімація та спеціальні інструменти, залежно від потреб проекту та досвіду команди, дозволяє створювати індивідуалізовані та ефективні мікроанімації для покращення користувацького досвіду.

Варто зазначити, що мікроанімація виступає не лише як візуальне прикраса, але й як стратегічний інструмент для впливу на поведінку користувачів та досягнення конкретних цілей веб-сайту. Вона допомагає залучати увагу, створювати персоналізований досвід та розповідати історію бренду, що робить користування сайтом більш цікавим та ефективним для відвідувачів.

Використання технології 3D-анімації та просувної типографіки є ключовими напрямками, що революціонізують спосіб, яким веб-сайти сприймаються користувачами. Використання 3D-анімації дозволяє створювати реалістичні та переконливі зображення, надаючи їм глибину і об'єм, що робить взаємодію з аудиторією більш захоплюючою та вражаючою. Технологія 3D привносить новий рівень реалізму у веб-дизайн, підсилюючи враження від інтерактивності та підкреслюючи бренд.

У той же час, оригінальна типографіка веб-дизайну переживає значний розвиток завдяки новаторським технологіям та підходам. Використання нових веб-шрифтів, CSS Grid та Flexbox, анімації тексту та Variable Fonts розширює можливості дизайнерів у створенні різноманітних та ефективних типографічних рішень. Ці новації спрямовані на поліпшення зручності та доступності веб-типографіки для користувачів, забезпечуючи при цьому якість та естетику веб-дизайну.

Мобільна адаптація веб-сайтів є критично важливим аспектом сучасного веб-дизайну, оскільки користувачі все більше використовують мобільні пристрої для доступу до Інтернету. Респонсивний дизайн, мобільні версії сайтів, швидкість завантаження, користувацька зручність та SEO-оптимізація для мобільних пристроїв - всі ці аспекти спрямовані на забезпечення ефективного користувацького досвіду на мобільних платформах.

У мобільній адаптації веб-дизайну спостерігаються постійні інновації. Прискорена мобільна адаптація, покращений респонсивний дизайн, мобільні анімації та інтерактивність, інтеграція з мобільними функціями пристроїв, оптимізація для голосового пошуку та фокус на мобільному дизайні - всі ці напрямки покликані покращити користувацький досвід на мобільних пристроях.

Поруч з цим, голосовий інтерфейс користувача в веб-дизайні набуває все більшої популярності, завдяки розвитку голосових технологій та віртуальних асистентів. Голосовий пошук та навігація, персоналізований досвід, інтерактивність, оптимізація для пошукових систем, адаптація до мобільних пристроїв, безпека і конфіденційність – ці аспекти спрямовані на підвищення доступності та зручності взаємодії з веб-сайтами через голосовий інтерфейс користувача.

Інновації в персоналізації веб-дизайну визначають нові стандарти ефективності та взаємодії з користувачами. Використання машинного навчання та аналізу даних дозволяє створювати персоналізовані пропозиції та рекомендації, забезпечуючи унікальний досвід для кожного користувача. Динамічний контент та адаптивний дизайн реагують на потреби та інтереси користувачів, забезпечуючи їм зміст, який найбільше відповідає їхнім очікуванням. Персоналізовані пропозиції та акції роблять взаємодію з веб-сайтом більш привабливою та індивідуалізованою.

Аналіз поведінки користувачів в реальному часі дозволяє швидко адаптувати зміст та функціональність сайту, забезпечуючи найбільш оптимальний досвід для кожного відвідувача. Сегментація аудиторії та створення персоналізованих сторінок розширює можливості взаємодії з користувачами, враховуючи їхні інтереси та потреби.

Ці інновації в персоналізації веб-дизайну ведуть до підвищення залученості користувачів, зростання конверсії та покращення загального користувацького досвіду. Вони стають не лише ключовими стратегіями для розвитку успішного веб-проекту, але й важливим елементом визначення конкурентних переваг у світі онлайн-бізнесу.

ВИСНОВКИ

У сфері веб-розробки існує розмаїття мов програмування, таких як JavaScript, Python, PHP, Java, Ruby і багато інших. Кожна мова має свої переваги і недоліки, що дозволяє розробникам обирати той інструмент, який найкраще відповідає їхнім потребам. JavaScript і Python володіють потужними засобами і водночас є досить простими для вивчення і використання. Це сприяє полегшенню розробки веб-додатків для новачків і дозволяє швидко розгортати проекти. Крім того, ці мови користуються великою популярністю і мають великі спільноти розробників, що полегшує доступ до документації, бібліотек, фреймворків і ресурсів. Деякі мови, такі як Java і C#, відрізняються великою продуктивністю і ефективністю, що дозволяє створювати великі та масштабовані веб-додатки. Вони особливо підходять для великих команд розробників і підприємств.

Часто веб-розробка включає використання комплексних технологічних стеків, які складаються з різних мов програмування, фреймворків, бібліотек і інструментів. Вибір правильних компонентів для технологічного стеку є важливою складовою успішного проекту.

Існує широкий вибір інструментів і технологій для веб-розробки, які прогнозують яскраве майбутнє для цієї галузі. Від інтеграції штучного інтелекту до впровадження новітніх систем безпеки, перспективи веб-розробки та цифрового досвіду надзвичайно захоплюють.

Фреймворки надають зручні інструменти, які спрощують роботу розробників, дозволяючи їм більше часу приділити бізнес-аналізу, а не рутинному написанню коду.

Кібербезпека продовжує розвиватися, з'являються нові методи захисту, від двофакторної аутентифікації до архітектури нульової довіри.

Технології доповненої та віртуальної реальності стають все більш важливими, відкриваючи безліч можливостей для бізнесу у майбутньому.

Безсерверний підхід ефективно використовує ресурси, економлячи компаніям витрати на купівлю та обслуговування серверів, що сприятиме стійкості системи і дозволить зосередитись на більш важливих завданнях.

ШІ став необхідною складовою веб-розробки, змінюючи підхід до створення та управління веб-сайтами та додатками. Він привносить численні переваги, від персоналізованого досвіду до передового аналізу даних і автоматизації. Оскільки технології розвиваються, ШІ стане ще більш важливим у майбутньому веб-розробки.

Щоб залишатися попереду, компаніям потрібно досліджувати стратегії інтеграції ШІ та пріоритизувати прийняття рішень на основі даних. Інтеграція ШІ вимагає глибокого розуміння технологій та їх можливого застосування, а також збір та аналіз відповідних даних для навчання моделей ШІ.

Це використовується для модернізації процесів розробки та проектування веб-сайтів, від покращення взаємодії з користувачами до оптимізації продуктивності та автоматизації процесів. Використання аналітики даних і машинного навчання дає змогу компаніям отримувати цінну інформацію про користувачів та надавати персоналізований досвід.

У сучасному веб-дизайні велика увага приділяється веб-анімації, яка стала не лише необхідною складовою сайтів, але й ключовою для створення успішного та інтерактивного користувацького досвіду. Новаторські тенденції, такі як динамічні мікроанімації, визначають майбутнє веб-дизайну у 2024 році.

Мікроанімація грає ключову роль у підвищенні візуальної привабливості сайтів та поліпшенні їхньої інтерфейсної взаємодії, створюючи тонкий рух та візуальний зворотний зв'язок, поліпшуючи враження та ефективність взаємодії з сайтом. Використання різних ефектів CSS, анімації, бібліотек JavaScript дає змогу створювати індивідуалізовані та ефективні мікроанімації для поліпшення користувацького досвіду. Мікроанімація є не тільки візуальною прикрасою, але й стратегічним засобом для впливу на поведінку користувачів та досягнення конкретних цілей сайту. Вона допомагає залучати увагу, створювати персоналізований досвід та розповідати історію бренду, зробивши взаємодію з сайтом більш цікавою та ефективною для відвідувачів.

Технологія 3D-анімації та оригінальна типографіка є ключовими напрямками, що революціонізують спосіб, яким сайти сприймаються користувачами. Використання 3D-анімації дозволяє створювати реалістичні та переконливі зображення, а оригінальна типографіка розширює можливості дизайнерів у створенні різноманітних та ефективних типографічних рішень.

Мобільна адаптація є критично важливим аспектом сучасного веб-дизайну, оскільки користувачі все більше використовують мобільні пристрої для доступу до Інтернету. Постійні інновації у мобільній адаптації, такі як прискорена мобільна адаптація та мобільні анімації, спрямовані на покращення користувацького досвіду на мобільних пристроях.

Голосовий інтерфейс користувача набуває все більшої популярності, завдяки розвитку голосових технологій та віртуальних асистентів. Його використання спрямоване на підвищення доступності та зручності взаємодії з веб-сайтами.

Персоналізація веб-дизайну, що базується на машинному навчанні та аналізі даних, дозволяє створювати унікальні та персоналізовані пропозиції для кожного користувача, покращуючи їхній досвід взаємодії з сайтом.

Дослідження та аналіз інновацій у веб-розробці, використання нових технологій, дизайну та інтерфейсів підтверджують, що ця галузь залишається динамічною та постійно розвивається. Застосування новаторських підходів до веб-дизайну та розробки дозволяє створювати більш ефективні та привабливі веб-сайти, які відповідають потребам сучасного користувача. Нові технології, зокрема ШІ, веб-анімація, машинне навчання, голосовий інтерфейс користувача, відкривають безліч можливостей для покращення якості та функціональності веб-продуктів. Інтеграція цих технологій у процес розробки сприяє підвищенню залучення користувачів, покращенню взаємодії з сайтом та підвищенню конверсії.

Важливо продовжувати досліджувати та використовувати нові інструменти, технології та підходи у веб-розробці, щоб залишатися конкурентоспроможними та задовольняти зростаючі потреби користувачів. Постійне оновлення навичок, вивчення нових технологій та впровадження кращих практик необхідне для створення веб-продуктів, що не тільки задовольняють сучасні стандарти, а й визначають нові тенденції у цій динамічній галузі.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бугай А.А., Олійник В.В. Коцептуальна модель адаптивного веб-інтерфейсу користувача з використанням інтелектуальних технологій. Міжвідомчий науково-технічний збірник «Адаптивні системи автоматичного управління».2018. № 1 (32). С. 15-22.
2. Веб-анімація: підвищення взаємодії та покращення досвіду користувача. URL: <https://coi.ua/blog/DesignCo/web-animation-enhancing-interaction-and-improving-user-experience/> (Дата звернення 19.05.2024).
3. Веб-фреймворк. URL: <https://ewikiuk.top/wiki/Web_framework> (Дата звернення 26.04.2024).
4. Гнатишин М.А. Аналіз стану прогресивних технологій для створення веб-застосунків: кваліфікаційна робота бакалавра за спеціальністю «122 – комп’ютерні науки» / М.А. Гнатишин. — Тернопіль : ТНТУ, 2023. — 61 с. URL: <https://elartu.tntu.edu.ua/handle/lib/41679>
5. Душейко А.О. Аналіз та використання 3D моделювання у веб-розробці для покращення користувацького інтерфейсу. 2024. Master's Thesis. КПІ ім. Ігоря Сікорського. URL: <https://ela.kpi.ua/server/api/core/bitstreams/5711c86b-406b-41fe-b257-4ef3a721a79b/content>
6. Дущенко О.С. Огляд історії розвитку Web. Науковий журнал "Фізико Математична освіта. 2018. Випуск 2 (16). С.46-50.
7. Козюля А. 5 найпопулярніших фреймворків для веброзробки. URL: <https://dan-it.com.ua/uk/blog/5-najpopuljarnishih-frejmvorkiv-dlja-vebrozrobki/> (Дата звернення 26.04.2024).
8. Погореленко А.К. Штучний інтелект: сутність, аналіз застосування, перспективи розвитку. Економічні науки. 2018. Випуск 32. С. 22–27.
9. Програма конференції молодих науковців. Ніжинський державний університет імені Миколи Гоголя, 16-24 травня 2024 р., м. Ніжин.
10. Тарасова М.В. Порівняння фреймворків для веб-розробки на прикладі AngularJS та React. Міжнародний науковий журнал "Інтернаука". Серія: "Інформатика". 2017. №11. С. 14-18.
11. Фреймворк – це ключ до майстерної веб-розробки: дізнайтеся, як обрати найкращий для вашого проєкту. URL: <https://bdut.co.ua/pro-nas/freymvork/> (Дата звернення 26.04.2024).
12. Що таке розмовний ШІ: принципи та приклади використання. URL: <https://www.chatcompose.com/uk/conversational.html> (Дата звернення 13.05.2024).
13. Що таке штучний інтелект: характеристики, типи, сфери використання. URL: <https://futurenow.com.ua/shho-take-shtuchnyj-intelekt/> (Дата звернення 08.05.2024).
14. Штучний інтелект. URL: <https://termin.in.ua/shtuchnyy-intelekt/> (Дата звернення 11.05.2024).
15. **27 Eye-Opening Website Statistics: Is Your Website Costing You Clients?** <https://www.sweor.com/firstimpressions> (Дата звернення 05.04.2024).
16. AI in Web Development: Breaking Down the Future. URL: <https://medium.com/theymakedesign/ai-in-web-development-9a1b5f04eee> (Дата звернення 19.05.2024).
17. Breaking down the advantages and disadvantages of artificial intelligence. URL: <https://www.ibm.com/blog/breaking-down-the-advantages-and-disadvantages-of-artificial-intelligence/> (Дата звернення 10.05.2024).
18. Goh, H. A., Ho, C. K., Abas, F. S. (2022). Front-end deep learning web apps development and deployment: a review. Applied Intelligence, 1-23. https://doi.org/10.1007/ s10489-022-04278-65.
19. Huang J., Saleh S. & Liu Y. (2021). A review on artificial intelligence in education. Academic Journal of Interdisciplinary Studies. Vol. 10 (3). 206–217. https://doi.org/10.36941/ajis-2021-0077.
20. Petruk M. How to Use AI in Web Development?URL: [https://wesoftyou.com/ai/how-to-use-ai-in-web-development/#:~:text=AI%20also%20enables%20advanced%20data,accurate%20predictions%20about%20future%20purchases](https://wesoftyou.com/ai/how-to-use-ai-in-web-development/#:~:text=AI also enables advanced data,accurate predictions about future purchases) (Дата звернення 01.05.2024).
21. Thompson, E., Brown, L. (2019). Assessing the Effectiveness of Responsive Web Design in Enhancing Mobile User Experience. Journal of Mobile Technology, 8(4), 56-68.
22. Tom Dieck, M. Claudia & Cranmer, Eleanor & Prim, Alexandre & Bamford, David. (2023). The effects of augmented reality shopping experiences: immersion, presence and satisfaction. Journal of Research in Interactive Marketing.March 2023. № 17 (10). 940-958
23. What are the advantages and disadvantages of artificial intelligence (AI)? URL: <https://www.tableau.com/data-insights/ai/advantages-disadvantages> (Дата звернення 10.05.2024).

ДОДАТКИ

Додаток А

**Додавання чат-бота на веб-сайт за допомогою JavaScript та WebSocket**

1. Створюємо сервер, який буде опрацьовувати повідомлення від чат-бота та відправляти відповіді.

const WebSocket = require('ws');

const wss = new WebSocket.Server({ port: 8080 });

wss.on('connection', function connection(ws) {

  ws.on('message', function incoming(message) {

    console.log('received: %s', message);

// Тут можна обробляти отримані повідомлення та відправляти відповіді

    ws.send('Hello, I am a chatbot!');

  });

});

2. Записуємо код JavaScript у HTML-файл:

<script>

  const ws = new WebSocket('ws://localhost:8080');

  ws.onopen = function () {

    console.log('Connected to the chatbot server!');

  };

  ws.onmessage = function (event) {

    console.log('Received message: ', event.data);

// Тут можна відобразити повідомлення в чат-вікні на вашому веб-сайті

  };

  function sendMessage(message) {

    ws.send(message);

    console.log('Sent message: ', message);

// Тут також можна відобразити відправлене повідомлення в чат-вікні

  }

</script>

3. Наступним етапом може бути стилізація чату за допомогою CSS.

Додаток Б

**Мікроанімації для кнопки на веб-сайті за допомогою CSS та JavaScript**

// HTML:

<!DOCTYPE html>

<html lang="en">

<head>

 <meta charset="UTF-8">

 <meta name="viewport" content="width=device-width, initial-scale=1.0">

 <title>Microanimation Example</title>

 <link rel="stylesheet" href="styles.css">

</head>

<body>

 <button class="animated-button">Hover Me</button>

 <script src="script.js"></script>

</body>

</html>

// CSS (styles.css):

.animated-button {

 padding: 10px 20px;

 background-color: #007bff;

 color: #fff;

 border: none;

 border-radius: 5px;

 cursor: pointer;

 transition: background-color 0.3s, transform 0.3s;

}

.animated-button:hover {

 background-color: #0056b3;

 transform: scale(1.05);

}

Додаток В

**Додавання 3D-об'єкта на веб-сайт за допомогою бібліотеки Three.js**

1. Підключимо бібліотеку Three.js, використаємо CDN:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>3D Animation Example</title>

<script src="https://cdnjs.cloudflare.com/ajax/libs/three.js/r128/three.min.js"></script>

</head>

<body>

<script src="script.js"></script>

</body>

</html>

2. Створюємо JavaScript файл для налаштування та створення 3D сцени:

// script.js Створюємо сцену

var scene = new THREE.Scene();

// Створюємо камеру

var camera = new THREE.PerspectiveCamera(75, window.innerWidth / window.innerHeight, 0.1, 1000);

camera.position.z = 5;

// Створюємо візуалізатор

var renderer = new THREE.WebGLRenderer();

renderer.setSize(window.innerWidth, window.innerHeight);

document.body.appendChild(renderer.domElement);

// Створюємо геометрію

var geometry = new THREE.BoxGeometry();

// Створюємо матеріал

var material = new THREE.MeshBasicMaterial({ color: 0x00ff00 });

// Створюємо куб

var cube = new THREE.Mesh(geometry, material);

scene.add(cube);

// Анімація

function animate() {

 requestAnimationFrame(animate);

 cube.rotation.x += 0.01;

 cube.rotation.y += 0.01;

 renderer.render(scene, camera);

}

animate();

3. Зберігаємо цей файл як script.js в тій же папці, де знаходиться основний HTML-файл.

Додаток Г

**Голосовий інтерфейс з Web Speech API**

<!DOCTYPE html>

<html lang="en">

<head>

 <meta charset="UTF-8">

 <meta name="viewport" content="width=device-width, initial-scale=1.0">

 <title>Voice Interface Example</title>

</head>

<body>

 <h1>Voice Interface Example</h1>

 <button onclick="startDictation()">Start Dictation</button>

 <div id="output"></div>

 <script>

 function startDictation() {

 if (window.hasOwnProperty('webkitSpeechRecognition')) {

 var recognition = new webkitSpeechRecognition();

 recognition.continuous = false;

 recognition.interimResults = false;

 recognition.lang = 'en-US';

 recognition.start();

 recognition.onresult = function(e) {

 document.getElementById('output').innerHTML = '';

 for (var i = e.resultIndex; i < e.results.length; ++i) {

 document.getElementById('output').innerHTML += e.results[i][0].transcript;

 } };

 recognition.onend = function() {

 recognition.stop();

 };

 }

 }

 </script>

</body>

</html>