

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДОНЕЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ВАСИЛЯ СТУСА

ОПАЛЬНИК ОЛЕКСАНДР АНАТОЛІЙОВИЧ

Допускається до захисту:
в.о. завідувача кафедри
інформаційних технологій
к.т.н, доцент

_____ Зелінська О.В.

«__» _____ 2024 р.

**РОЗРОБКА ОНЛАЙН ІНТЕРНЕТ-МАГАЗИНУ ОДЯГУ З
ПІДСИСТЕМОЮ РЕКОМЕНДАЦІЙ ТОВАРІВ**

Спеціальність 122 Комп'ютерні науки

Кваліфікаційна (бакалаврська) робота

Керівник:

Бабаков Р.М., професор кафедри

інформаційних технологій,

д.т.н, доцент

Оцінка: __/__/____

Голова ЕК: _____

Вінниця – 2024

АНОТАЦІЯ

Опальник О.А. Розробка онлайн-інтернет магазину одягу з підсистемою рекомендацій. Спеціальність 122 «Комп'ютерні науки», освітня програма «Комп'ютерні науки». Донецький національний університет імені Василя Стуса, Вінниця, 2024.

У кваліфікаційній (бакалаврській) роботі проведено дослідження переваг та недоліків існуючих онлайн-магазинів одягу, а також обґрунтовано актуальність створення інноваційного інтернет магазину з підсистемою рекомендацій. Окреслені ключові етапи розробки проекту від вибору технологій та програмних засобів до функціональності готового веб-сайту. Описано необхідність створення цього продукту у контексті сучасних тенденцій електронної комерції. В роботі проаналізовано відповідність вимог інтернет-магазину з підсистемою рекомендацій і висвітлено можливості подальшого розвитку проекту.

Ключові слова: онлайн-магазин, одяг, рекомендації, електронна комерція, розробка, технології, програми.

ABSTRACT

Opalnyk O.A. Development of an online clothing store with a subsystem of recommendations. Specialty 122 "Computer Science", educational program "Computer Science". Vasyl Stus Donetsk National University, Vinnytsia, 2024.

In the qualifying (bachelor's) thesis, the advantages and disadvantages of existing online clothing stores were studied, and the relevance of creating an innovative online store with a subsystem of recommendations was substantiated. The key stages of project development are outlined, from the selection of technologies and software tools to the functionality of the finished website. The need to create this product in the context of modern e-commerce trends is described. The work analyzed the compliance of the requirements of the online store with the subsystem of recommendations and highlighted the possibilities of further development of the project.

Keywords: online store, clothes, recommendations, e-commerce, development, technologies, programs.

ЗМІСТ

ВСТУП.....	4
РОЗДІЛ 1. ТЕОРЕТИЧНА ІНФОРМАЦІЯ ЩОДО РОЗРОБКИ САЙТУ.....	6
1.1. Огляд інтернет-магазинів, та систем рекомендацій.....	6
1.2. Аналіз конкурентних сайтів.....	8
1.3. Постановка задачі.....	12
РОЗДІЛ 2. АНАЛІЗ ТЕХНОЛОГІЙ ДЛЯ НАПИСАННЯ САЙТУ.....	14
2.1. Огляд технологій для розробки сайту.....	14
2.2 PHP - як мова програмування для сайту.....	16
2.3 HTML та CSS як технологія розробки візуальної частини сайту.....	19
2.4 Огляд CSS-бібліотеки Bootstrap 4.....	23
2.5 Огляд скриптової мови програмування JavaScript	26
2.6 Середовище для розробки сайту.....	28
2.7. Розробка backlog-у та структури сайту.....	30
2.8. Середовище для розгортання онлайн-магазину.....	32
РОЗДІЛ 3. РОЗРОБКА ОНЛАЙН ІНТЕРНЕТ-МАГАЗИНУ ОДЯГУ З ПІДСИСТЕМОЮ РЕКОМЕНДАЦІЇ ТОВАРІВ.....	34
3.1. Написання коду інтернет-магазину.....	34
3.2. Можливості розвитку та впровадження додаткового функціоналу інтернет-магазину.....	38
ВИСНОВКИ.....	42
СПИСОК ВИКОРИСТАНИХ ПОСИЛАНЬ.....	44
ДОДАТКИ.....	45

ВСТУП

У сучасному цифровому світі інтернет-магазини стають все більш популярними серед споживачів, які шукають зручність, швидкість та різноманіття у закупівлях. Розробка онлайн інтернет-магазину одягу і системою рекомендації товарів є актуальною та важливою задачею в епоху цифрової трансформації торгівлі.

Підсистема рекомендацій товарів в інтернет-магазині одягу може значно полегшити процес вибору покупцем, забезпечуючи персоналізовані пропозиції на основі його вподобань, історії покупок та трендів моди. Це допомагає покупцям знайти не лише те, що їм потрібно, але й стимулює до додаткових покупок, підвищуючи обсяги продажів для бізнесу.

У даній дипломній роботі ми детально розглянемо процес розробки онлайн інтернет-магазину одягу з підсистемою рекомендації товарів, вивчимо сучасні технології та методи аналізу даних для підтримки рекомендаційної системи, а також проаналізуємо вплив цього інноваційного підходу на ефективність та конкурентоспроможність інтернет-магазину в сучасному електронному бізнесі.

Першим кроком у розробці інтернет-магазину з підсистемою рекомендацій є детальний аналіз вимог та потреб цільової аудиторії. Важливо розуміти, які критерії впливають на вибір товарів користувачами, їхні уподобання, попередні покупки та поведінку на сайті для успішного впровадження системи рекомендацій.

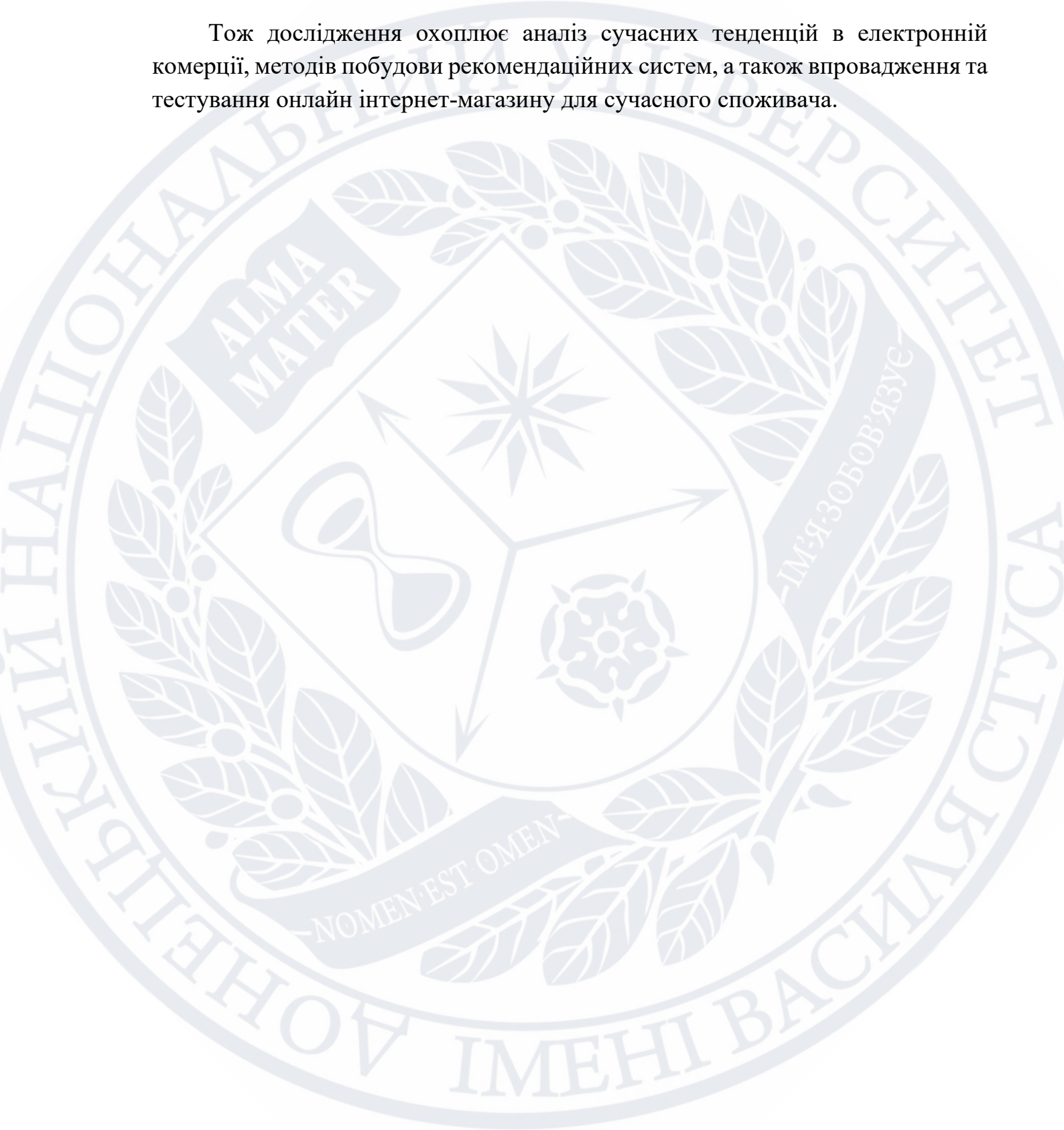
Після збору даних про користувачів та їхніх вподобань необхідно вибрати відповідні технології для реалізації підсистеми рекомендацій. Важливо вибрати ефективний алгоритм рекомендацій, який враховуватиме великий обсяг даних та забезпечує точні та релевантні рекомендації для кожного користувача.

Після запуску інтернет-магазину з підсистемою рекомендацій важливо провести тестування та аналітику використання для перевірки ефективності системи. На основі отриманих даних можна вносити виправлення та покращення до алгоритмів рекомендацій з метою підвищення їхньої точності та корисності для користувачів.

Ця дипломна робота спрямована на вивчення та розробку онлайн інтернет-магазину одягу з підсистемою рекомендацій товарів з метою

покращення користувацького досвіду, збільшення конверсії та підвищення ефективності електронної комерції у сучасному ринку.

Тож дослідження охоплює аналіз сучасних тенденцій в електронній комерції, методів побудови рекомендаційних систем, а також впровадження та тестування онлайн інтернет-магазину для сучасного споживача.



РОЗДІЛ 1

ТЕОРЕТИЧНА ІНФОРМАЦІЯ ЩОДО РОЗРОБКИ САЙТУ

Тут ми розглянемо конкурентів, та можливість реалізації потрібного нам продукту у вигляді інтернет-магазину

1.1. Огляд інтернет-магазинів, та систем рекомендацій

З розвитком цифрових технологій інтернет-магазини стали невід'ємною частиною сучасного бізнесу. Інтернет-магазини одягу, зокрема, займають значну частку ринку електронної комерції та пропонують великий вибір товарів для споживачів. У даному розділі розглянемо інтернет-магазини, створені на основі різних CMS-систем, таких як Woocommerce, PrestaShop, Opencart, а також самописні сайти. Оцінка цих магазинів буде проведена за критеріями безпеки, швидкості, UX (користувацький досвід) та UI (користувацький інтерфейс).

1. Woocommerce

Woocommerce є однією з найпопулярніших платформ для створення інтернет-магазинів, яка базується на **WordPress**. Вона має відкритий код і пропонує широкий спектр можливостей для налаштування.

Woocommerce забезпечує високий рівень безпеки завдяки регулярним оновленням та можливості інтеграції з різними плагінами для безпеки, такими як Wordfence. Застосування SSL сертифікатів та двофакторної автентифікації додає додатковий рівень захисту.

Швидкість роботи інтернет-магазину на Woocommerce значною мірою залежить від хостингу та оптимізації. Використання кешування та CDN (Content Delivery Network) суттєво покращує час завантаження сторінок.

Дана платформа пропонує безліч тем та плагінів, які дозволяють створити привабливий та зручний інтерфейс для користувачів. Різноманітність налаштувань дозволяє адаптувати магазин під специфічні потреби кожного бізнесу.

2. PrestaShop

PrestaShop - це ще одна популярна платформа для створення інтернет-магазинів, яка також базується на відкритому коді.

PrestaShop надає вбудовані інструменти для захисту магазину, такі як захист від SQL-ін'єкцій та можливість створення резервних копій. Розробники регулярно випускають оновлення для закриття вразливостей.

PrestaShop відома своєю швидкістю роботи, але для досягнення оптимальних результатів важливо використовувати продуктивний хостинг та оптимізаційні модулі.

Завдяки великій кількості доступних модулів та тем, PrestaShop дозволяє створити інтерфейс, який задовольнить найвибагливіших користувачів. Можливість гнучкого налаштування дозволяє поліпшити користувацький досвід.

3. Opencart

Opencart - це відкрита платформа для електронної комерції, яка пропонує простоту використання та широкий набір функцій.

Opencart забезпечує базові функції безпеки, такі як захист від CSRF та XSS атак. Проте для досягнення високого рівня захисту рекомендується встановлювати додаткові модулі та регулярно оновлювати систему.

Оптимізація продуктивності Opencart включає використання кешування, оптимізацію зображень та використання швидкого хостингу. Це допомагає зменшити час завантаження сторінок.

Opencart надає різноманітність тем та модулів, що дозволяють створити інтуїтивно зрозумілий та привабливий інтерфейс для користувачів. Простота навігації та зручність використання є ключовими перевагами даної платформи.

4. Самописний сайт

Самописний сайт - це рішення, яке розробляється з нуля відповідно до специфічних вимог клієнта та бізнесу.

Рівень безпеки самописного сайту залежить від професіоналізму розробників. Використання сучасних стандартів кодування, регулярні аудит та оновлення забезпечують високий рівень захисту.

Самописний сайт може бути оптимізований для досягнення максимальної швидкості роботи. Використання легковагового коду та оптимізації ресурсу значно покращує продуктивність.

Розробка з нуля дозволяє створити унікальний інтерфейс з урахуванням потреб цільової аудиторії. Це забезпечує високу гнучкість у дизайні та функціоналі, що позитивно впливає на користувацький досвід.

1.2. Аналіз конкурентних сайтів

Сучасні інтернет-магазини одягу не лише забезпечують зручний спосіб придбання речей, але й активно використовують системи рекомендацій для підвищення користувацького досвіду та збільшення продажів. В цьому розділі ми розглянемо три популярні українські інтернет-магазини одягу — Vovk, Lure Store та ISSA PLUS. Я зосередився на аналізі їхніх характеристик, включаючи швидкість завантаження, перевантаженість елементів інтерфейсу, простоту в технічній підтримці, а також користувацький досвід (UX) і користувацький інтерфейс (UI). Крім того, проаналізуємо використання систем рекомендацій в цих магазинах.

1. Інтернет-Магазин Vovk

Інтернет-магазин Vovk (<https://vovk.com/ua/>) демонструє хорошу швидкість завантаження сторінок. За даними інструментів для аналізу швидкості, таких як Google PageSpeed Insights, середній показник швидкості завантаження головної сторінки становить 85/100 для десктопної версії та 75/100 для мобільної.

Інтерфейс сайту Vovk організований досить добре, без надмірної кількості елементів, що можуть відволікати користувача. Категорії товарів, банери акцій та кнопки для навігації чітко розташовані та зрозумілі.

Технічна підтримка в Vovk доступна через онлайн-чат, електронну пошту та телефон. Відповіді на запити надаються оперативно, що значно підвищує рівень довіри та зручності для клієнтів.

З точки зору користувацького досвіду (UX) та користувацького інтерфейсу (UI), сайт Vovk добре спроектований. Простий і мінімалістичний дизайн, зручна навігація та адаптивна верстка забезпечують швидкий доступ до потрібної інформації і товарів.

На сайті Vovk можна помітити елементи системи рекомендацій, яка пропонує схожі товари залежно від переглянутих або обраних користувачем. Однак, інформація про конкретні алгоритми або технології, які використовуються, не надається.

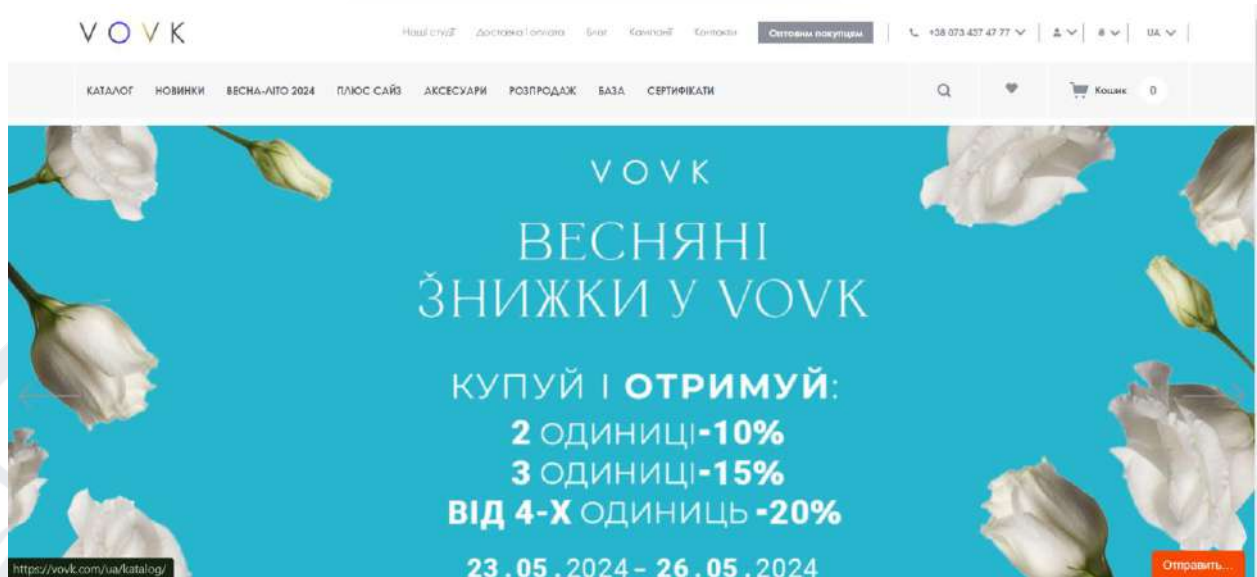


Рисунок 1.1- Інтерфейс інтернет-магазину Vovk

2. Інтернет-Магазин Lure Store

Lure Store (<http://lurestore.ua/>) також має високу швидкість завантаження, з показниками близькими до Vovk. Середній бал за даними PageSpeed Insights складає 80/100 для десктопної версії та 70/100 для мобільної.

Інтерфейс Lure Store більш насичений візуальними елементами та банерами, однак це не сильно впливає на загальне враження. В цілому, структура сайту залишається зрозумілою і логічно організованою.

Lure Store надає кілька каналів для технічної підтримки, включаючи онлайн-чат та контактну форму на сайті. Відповіді, як правило, надходять швидко, що робить взаємодію з магазином приємною та зручною.

З точки зору UX та UI, Lure Store має стильний та сучасний дизайн. Великий акцент робиться на візуальну складову, що сприяє привабливості сайту. Однак, деякі користувачі можуть відчувати перевантаженість інформацією через велику кількість зображень та банерів.

Lure Store використовує систему рекомендацій для пропозиції товарів на основі попередніх переглядів та покупок користувачів. Як і у випадку з Vovk, конкретні деталі алгоритмів не публікуються.

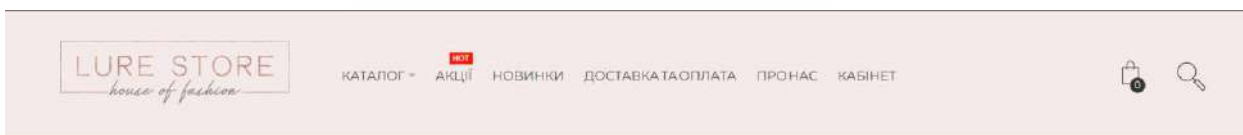


Рисунок 1.2 -Інтерфейс інтернет-магазину Lure Store

3. Інтернет-Магазин ISSA PLUS

ISSA PLUS (<https://issaplus.com/ua>) демонструє найвищу швидкість завантаження серед розглянутих магазинів. Середній бал PageSpeed Insights становить 90/100 для десктопної версії та 85/100 для мобільної.

Інтерфейс ISSA PLUS дуже чистий і мінімалістичний. Всі елементи логічно розміщені та легкі для сприйняття, що робить взаємодію з сайтом простою і приємною.

Інтернет-магазин ISSA PLUS надає користувачам кілька варіантів для зв'язку з технічною підтримкою, включаючи онлайн-чат та електронну пошту. Відповіді на запити надаються швидко та професійно.

З точки зору UX та UI, сайт ISSA PLUS є прикладом відмінного дизайну. Простота використання, швидка навігація та адаптивність під різні пристрої створюють позитивний користувацький досвід.

ISSA PLUS активно використовує системи рекомендацій, що базуються на поведінці користувачів, включаючи переглянуті товари та попередні покупки. Це допомагає пропонувати релевантні товари та підвищує ймовірність здійснення покупок.

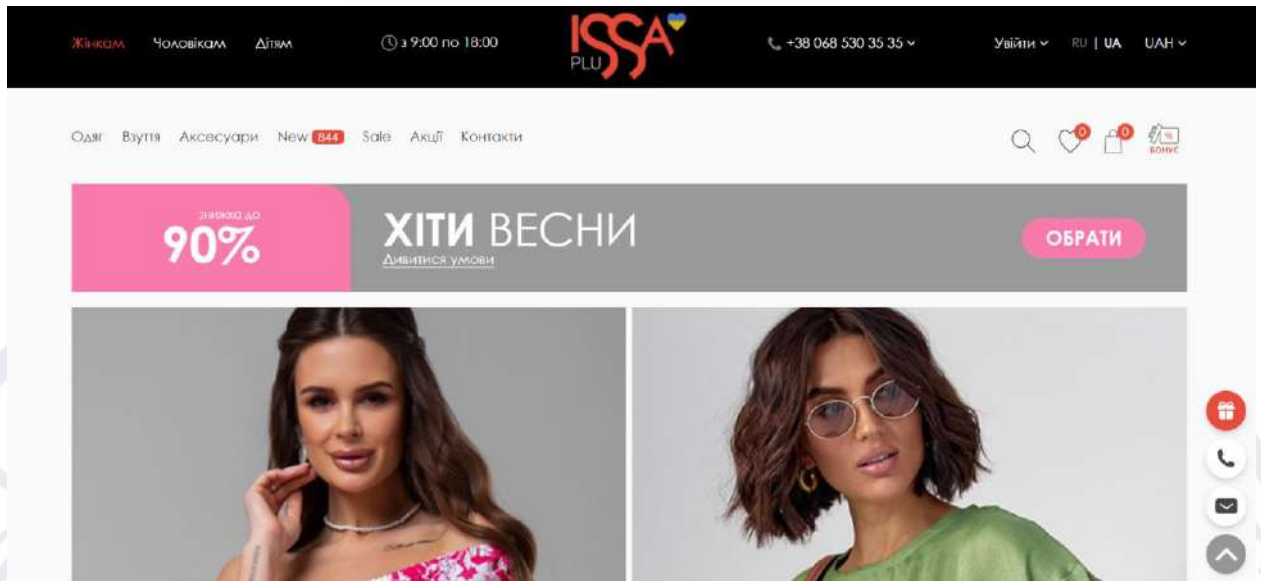


Рисунок 1.3- Інтерфейс інтернет-магазину Issa Plus

З метою порівняння обраних магазинів, було створено таблицю 1.1, що дозволяє, чітко побачити переваги та недоліки кожного конкурента.

Таблиця 1.1 Переваги та недоліки кожного конкурента

Показник	Vovk	Lure Store	ISSA PLUS
Швидкість завантаження	Десктоп: 85/100 Мобільна: 75/100	Десктоп: 80/100 Мобільна: 70/100	Десктоп: 90/100 Мобільна: 85/100
Перевантаженість елементів	Мінімальна	Помірна	Мінімальна
Проста технічна підтримка	Онлайн-чат, електронна пошта, телефон	Онлайн-чат, контактна форма	Онлайн-чат, електронна пошта
UX та UI	Добре спроектований, мінімалістичний дизайн, зручна навігація	Стильний та сучасний дизайн, великий акцент на візуальну складову	Приклад відмінного дизайну, простота використання, швидка навігація
Системи рекомендацій	Є, але деталі алгоритмів не надаються	Є, але деталі алгоритмів не надаються	Активно використовуються, базуються на поведінці користувачів

1.3 Постановка задачі

Створити інтернет-магазин одягу з підсистемою рекомендацій товарів, який буде включати наступний функціонал:

Реєстрація та Авторизація: Користувачам буде надана можливість реєстрації на сайті, щоб мати особистий обліковий запис. Вони зможуть авторизуватися в системі, використовуючи свої дані.

Каталог Товарів: На сайті буде розміщений повний каталог одягу з можливістю перегляду різних категорій товарів, таких як верхній одяг, взуття, аксесуари тощо.

Фільтрація та сортування: Користувачі зможуть використовувати різні фільтри та сортування для зручного пошуку товарів за категоріями, розміром, кольором, ціною, брендом.

Детальні описи з фотографіями, відео та відгуками: Кожен товар буде супроводжуватися детальним описом, фотографіями з різних кутів, а також можливістю додати відео та читати відгуки від інших покупців.

Пошук: Користувачі матимуть можливість швидко знаходити потрібні товари за допомогою функції пошуку за ключовими словами або параметрами.

Система Рекомендацій: Підсистема рекомендацій буде базуватися на попередніх переглядах та покупках користувачів, надаючи персоналізовані рекомендації для кожного клієнта.

Корзина та Оформлення Замовлення: Після вибору товарів користувачі зможуть додати їх до корзини та оформити замовлення, вказавши необхідні дані для доставки та оплати.

Особистий Кабінет: Користувачі матимуть доступ до особистого кабінету, де зможуть переглядати та редагувати свою особисту інформацію, а також переглядати історію своїх замовлень.

РОЗДІЛ 2

АНАЛІЗ ТЕХНОЛОГІЙ ДЛЯ НАПИСАННЯ САЙТУ

Тут ми оглянемо технології та оберемо найкращі для швидкої розробки сайту, бо час - це дуже цінний ресурс.

2.1. Огляд технологій для розробки сайту

Вибір технологій для розробки веб сайту є критичним етапом у створенні успішного веб проекту. Три популярні технології, що часто використовуються для розробки веб сайтів, — це Django, Spring та PHP. Кожна з цих технологій має свої особливості, переваги та недоліки. У цьому розділі буде проведено детальний огляд цих технологій з аналізом їхніх характеристик та аргументами на користь PHP як кращого вибору для багатьох відеопроєктів.

Django

Django — це високорівнева веб-фреймворк для мови програмування Python, що сприяє швидкому розробленню та чистому, прагматичному дизайну. Він розроблений для швидкого створення складних, динамічних веб сайтів з мінімальними зусиллями.

Переваги Django:

- Швидкість розробки: Вбудовані інструменти та адміністративна панель дозволяють швидко створювати та розгортати веб сайти.
- Безпека: Вбудовані засоби захисту від багатьох розповсюджених загроз, таких як SQL-ін'єкції, XSS, CSRF.
- Масштабованість: Підходить для створення масштабованих додатків.
- Сильна спільнота: Велика кількість готових бібліотек та розширень.

Недоліки Django:

- Висока кривизна навчання: Для новачків Python або веб розробки Django може здаватися складним.
- Монотонність: Суворі структура може бути обтяжуючою для деяких розробників.

Spring

Spring — це потужний Java фреймворк, який використовується для створення складних корпоративних додатків. Він включає різноманітні проекти, такі як Spring Boot, який спрощує розробку за рахунок надання готових шаблонів та інструментів.

Переваги Spring:

- Гнучкість: Широкий набір інструментів та проектів для різних потреб.
- Модульність: Можливість вибору та використання лише потрібних компонентів.
- Підтримка великих додатків: Ідеальний для створення великих, масштабованих корпоративних додатків.
- Компонентно-орієнтована архітектура: Легкість в управлінні залежностями та підвищена модульність коду.

Недоліки Spring:

- Складність налаштування: Налаштування Spring може бути складним для новачків.
- Кривизна навчання: Java та Spring мають високу кривизну навчання для нових розробників.

PHP

PHP — це популярна мова сценаріїв, що в основному використовується для веб розробки. PHP є серверною мовою програмування, яка є основою багатьох популярних CMS, таких як WordPress, Drupal, Joomla.

Переваги PHP:

- Простота навчання: Легко вивчити і почати використовувати.
- Широке поширення: Використовується на великій кількості веб сайтів та підтримується багатьма хостинг-провайдерами.
- Велика спільнота: Багато ресурсів, документації та готових рішень.
- Інтеграція з базами даних: Добре інтегрується з різними базами даних, такими як MySQL, PostgreSQL.

- Гнучкість: Можливість створення різноманітних веб додатків від простих до складних.

Недоліки PHP:

- Безпека: Історично PHP вважається менш безпечною мовою, хоча багато з цих проблем вирішено у нових версіях.
- Підтримка великих додатків: Хоча PHP підходить для створення великих додатків, він може бути менш ефективним у порівнянні з Java та іншими мовами.

Хоча Django та Spring мають свої сильні сторони та ідеально підходять для певних типів проєктів, PHP залишається кращим вибором для багатьох веб сайтів завдяки своїй простоті, широкому поширенню, гнучкості та високій продуктивності. Ці фактори роблять PHP ідеальним рішенням для швидкого та ефективного створення веб сайтів різного рівня складності.

2.2 PHP - як мова програмування для сайту

PHP – це скрипт-мова (scripting language), що вбудовується в HTML, який інтерпретується та виконується на сервері. Найпростіше це показати на прикладі:

Трохи про PHP

Основна відмінність від CGI-скриптів, написаних іншими мовами, типу Perl або C - це те, що в CGI-програмах ви самі пишете виведений HTML-код, а використовуючи PHP - ви вбудовуєте свою програму в готову HTML-сторінку, використовуючи відкриваючий і закриває теги (у прикладі `<?php ... ?>`).

Відмінність PHP від JavaScript, полягає в тому, що PHP-скрипт виконується на сервері (Apache або Nginx), а клієнту передається результат роботи, тоді як JavaScript-код повністю передається на клієнтську машину і тільки там виконується.

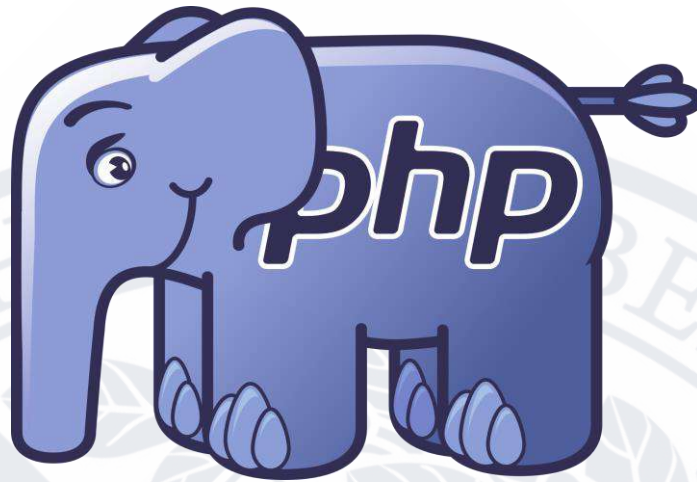


Рисунок 2.1- Логотип мови програмування PHP

Дехто скаже, що PHP дуже схожий на Active Server Pages (ASP), а ентузіасти Java скажуть, що PHP схожий на Java Server Pages (JSP). Всі три мови дозволяють розміщувати код, який виконується на Web-сервері, всередині HTML сторінок.

Можливості PHP

У кількох словах - на PHP можна зробити все, що можна зробити за допомогою CGI-програм. Наприклад: обробляти дані з форм, генерувати динамічні сторінки, отримувати та посилати куки (cookies).

Крім цього PHP включена підтримка багатьох баз даних (databases), що робить написання Web-додатків з використанням БД до неможливості простим.

Ось неповний перелік підтримуваних БД:

MySQL, PostgreSQL, MongoDB, SQLite

На додаток до всього PHP розуміє протоколи IMAP, SNMP, NNTP, POP3 і навіть HTTP, а також має можливість працювати з сокетом (sockets io) та спілкуватися за іншими протоколами.

Коротка історія PHP

Початком PHP можна вважати осінь 1994, коли Рasmus Лердорф (Rasmus Lerdorf) вирішив розширити можливості своєї Home-page (Домашньої сторінки) і написати невеликий движок для виконання найпростіших завдань.

Такий двигун був готовий до початку 1995 року і називався Personal Home Page Tools. Вмів він не дуже багато – розумів найпростішу мову і лише кілька макросів.

До середини 1995 року з'явилася друга версія, яка називалася PHP/FI Version 2. Приставка FI – приєдналася з іншого пакету Расмуса, який умів обробляти форми (Form Interpreter). PHP/FI компілювався всередину Apache та використовував стандартний API Apache. PHP скрипти виявилися швидше аналогічних CGI - скриптів, так як серверу не було необхідності породжувати новий процес. Мова PHP за можливостями наблизилася до Perl, найпопулярнішої мови для написання програм CGI. Була додана підтримка багатьох відомих баз даних (наприклад, MySQL та Oracle). Інтерфейс до GD - бібліотеки, дозволяв генерувати картинку на льоту. З цього моменту почалося широке поширення PHP/FI.

Наприкінці 1997 року Зеев Сураскі (Zeev Suraski) і Енді Гутманс (Andi Gutmans) вирішили переписати внутрішній двигун, з метою виправити помилки інтерпретатора і підвищити швидкість виконання скриптів. Через півроку, 6 червня 1998 вийшла нова версія, яка була названа PHP 3. До літа 1999 PHP 3 був включений в кілька комерційних продуктів. За даними NetCraft на листопад 1999 р. PHP використовувався в більш ніж 1 млн. доменах.

Чому потрібно обрати для цього проекту саме PHP?

Розробникам Web-додатків немає необхідності говорити, що web-сторінки - це текст і картинки. Цікавий сайт повинен підтримувати певний рівень інтерактивності з користувачем: пошук інформації, продаж продуктів, конференції тощо. Зазвичай усе це реалізувалося CGI-скриптами, написаними на Perl. Але CGI-скрипти дуже погано масштабуються. Кожен новий виклик CGI вимагає від ядра породження нового процесу, а це займає процесорний час і витрачає оперативну пам'ять. PHP пропонує інший варіант - він працює як частина Web-сервера, і цим схожий на ASP від Microsoft.

PHP синтаксис дуже схожий на синтаксис C або Perl. Люди, які знайомі з програмуванням, дуже швидко зможуть почати писати програми на PHP. У цій мові немає строгої типізації даних і немає необхідності в діях виділення/звільнення пам'яті.

Програми, написані на PHP, досить легко читанні. Написаний PHP – код легко візуально прочитати та зрозуміти, на відміну від Perl-програм.

Недоліки PHP

PHP є мовою, що інтерпретується, і, внаслідок цього, не може зрівнятися за швидкістю з компілюваним C або навіть Go. Однак при написанні невеликих програм, що, загалом, властиво проектам на PHP, коли весь проект складається з багатьох невеликих сторінок з кодом, вступають через накладні витрати на завантаження в пам'ять і виклик CGI-програми, написаної на C.

2.3 HTML та CSS як технологія розробки візуальної частини сайту

Web-сторінка є елементарною частиною Всесвітньої мережі (World Wide Web, www). WWW, своєю чергою, — одне з можливостей комп'ютерної мережі Internet, що пов'язує комп'ютери у світі.

Вся інформація www зберігається у вигляді окремих документів. В основі кожного такого документа лежить текст, який може бути оформлений різними шрифтами, стилями тощо. Основна властивість документів у Всесвітньому Павутинні — наявність посилань на інші документи, що знаходяться у Мережі. Такі посилання називаються гіперпосиланнями (hyperlink), вони дозволяють при перегляді швидко переходити від одного документа до іншого, і цим пов'язують всі документи в єдине ціле. Документи, про які йдеться, називаються web-сторінками. Вміст web-сторінок часто називають гіпертекстом.

Набір сторінок, що об'єднані загальною тематикою, називається сайтом (web site). Зазвичай весь сайт повністю зберігається на одному комп'ютері мережі, званому web-сервером (Apache або Nginx). Досить часто сервери містять декілька сайтів.

У кожного об'єкта, до якого можна отримати доступ через Internet (зокрема, у сайтів та окремих web-сторінок) є своя адреса в мережі, яка називається інакше уніфікованою адресою ресурсу (Uniform Resource Locator, URL).

Розглянемо приклад адреси для сайту: <http://www.donnu.edu.ua>. Перша частина адреси (“http://”) — назва протоколу, яким потрібно проводити передачу даних Мережею. Протокол — набір правил, які визначають, як дані повинні передаватися через мережу. Для різних видів даних використовують різні протоколи. У разі веб-сторінок застосовується протокол передачі гіпертексту (HyperText Transfer Protocol, HTTP). Для передачі довільних файлів

по мережі використовується протокол передачі файлів (File Transfer Protocol, FTP). Існують інші протоколи (Gopher, Telnet тощо. буд.).

За ім'ям протоколу на адресу слід власне адресу, яку краще читати праворуч наліво. Частина “ua” означає, що сайт знаходиться в доменній зоні України, та називається доменом (група сайтів) першого рівня. Домен 1-го рівня може бути географічним (ua - Україна, us - США, uk - Великобританія, de - Німеччина тощо) або тематичними (com - комерційний сайт, edu - освіта, gov - уряд, net - мережеві служби, ml - військова справа та оборона тощо). Усередині домену першого рівня знаходиться домен другого рівня, у нашому випадку - "edu" це означає, що сайт має відношення до освіти та науки. У принципі можливі домени третього та вищих рівнів. Зліва на адресу міститься назва самого сайту (“donnu”). Іноді, щоб підкреслити те, що адреса відноситься до Всесвітньої Павутини, додають частину "www".



Рисунок 2.2 - Логотип мови розмітки HTML

Для перегляду веб-сторінок використовуються спеціальні програми, які називаються браузерами (web browser). Вони дозволяють завантажувати з Мережі для перегляду сторінки із заданою адресою та переходити від документа до іншого за гіперпосиланнями. Найбільш відомими є браузери Netscape Navigator фірми Netscape та Internet Explorer фірми Microsoft.

Але для того, щоб ми змогли зверстати гарний сайт - нам потрібно використовувати CSS.

Під терміном CSS ми розуміємо фразу "Каскадні таблиці стилів". Вони впроваджуються в HTML-код і не вимагають жодних спеціальних редакторів і компіляторів, адже інтерпретуються звичайним браузером.

Каскадні таблиці стилів є описами різних HTML-елементів і створені вони для розширення властивостей останніх. Вперше стилі було запропоновано WWW Consortium у рамках розробки специфікації HTML 3.0, проте реально цю ініціативу почали підтримувати лише 1997 року. Насолодитися CSS мали можливість лише щасливі власники таких браузерів, як Netscape Navigator 4.0 та Internet Explorer 4.0.

На цьому з історичними фактами дозвольте закінчити та розпочати безпосередньо про CSS.

За допомогою CSS ми отримуємо можливість більш вишукано оформити свій Web-сайт, надати йому нові риси та симпатичну особу.



Рисунок 2.3 - Логотип Каскадної таблиці стилів

Синтаксис CSS досить простий у вивченні, тому освоєння каскадних стилів є досить легкою, але дуже корисною. Я хочу навести невеликий приклад. Візьмемо типову таблицю HTML та поставимо атрибут `border="1"`. Після інтерпретації цього коду браузером ми отримаємо звичайнісінький результат, до якого звик наше досвідчене око. Інша справа, що таблиці подібного роду не сумісні з жодним більш-менш хорошим дизайном. Ну куди

годиться ця псевдо-тривимірنا лінія, що є кордоном? Ось тут і приходять на допомогу CSS, за допомогою яких можна без зайвого гемору оформити будь-яку таблицю на власний смак.

Розглянемо невеликий приклад:

```
<STYLE TYPE="text/css">
```

Опис стилів форматування

```
</STYLE>
```

У контейнер тегів <STYLE> укладено опис різних елементів. Атрибут TYPE вказує тип стилів, але не є обов'язковим. Тим не менш, щоб уникнути будь-яких казусів, рекомендується ставити цей, зважаючи на те, що старі браузері можуть неправильно відобразити CSS.

Найчастіше каскадні таблиці розташовуються у шапці документа, але є винятки. Можливо, деякі з вас бачили такі файли, як rurok.css. Служать вони для того, щоб не перенасичувати сторінку CSS-кодом, завантажуючи цей із цих файлів.

Але може виникнути питання: Як описувати властивості деяких елементів сторінки? Насправді, в цьому немає нічого складного, і найпримітивніший випадок присвоєння стилів виглядає так:

```
ім'я елемента {властивість: значення;}
```

наприклад

```
boby {color: red;}
```

Тут "ім'я елемента" - це назва будь-якого HTML-тегу, чий властивості ми бажаємо задати. Наприклад ми взяли елемент body та виставили йому колір усього тексту червоний

Працюючи зі стилями необхідно пам'ятати, що не всі користувачі Мережі мають сучасні браузери, та деякі функції можуть не підтримуватися.

Такі динозаври не зможуть правильно інтерпретувати CSS-код, і обов'язково зіпсують тим самим ваше творіння.

2.4 Огляд CSS-бібліотеки Bootstrap 4

У цьому розділі я хочу роздивитися функціонал Bootstrap 4

Bootstrap - це потужний, багатофункціональний інтерфейсний інструментарій для швидкої розробки та прототипування сайту.



Рисунок 2.4 - Офіційний логотип CSS-бібліотеки Bootstrap

Щоб почати, потрібно увімкнути готові CSS і JavaScript Bootstrap стилі через CDN без будь-яких кроків по збиранню. Подивіться це практично за допомогою цієї демонстрації Bootstrap CodePen.

Також, щоб почати - потрібно створити новий файл index.html у корені проекту. Увімкнути також тег `<meta name="viewport">` для правильної чуйної поведінки на мобільних пристроях.

Ось приклад того, як повинна виглядати сторінка до підключення

```
<!doctype html>
```

```
<html lang="ru">
```

```
<head>
```

```
<meta charset="utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```
<title>Демо Bootstrap</title>

</head>

<body>

<h1>Hello World!</h1>

</body>

</html>
```

Наступним кроком буде увімкнення JS та CSS від Bootstrap. Треба помістити тег `<link>` у `<head>` для CSS і тег `<script>` для пакету JavaScript (включаючи Popper для позиціонування розкритих списків, спливаючі повідомлення та підказки) перед закриваючим `</body>`.

Ось так це повинно виглядати:

```
<!doctype html>

<html lang="en">

<head>

<meta charset="utf-8">

<meta name="viewport" content="width=device-width, initial-scale=1">

<title>Bootstrap demo</title>

<link
href="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/css/bootstrap.min.css"
rel="stylesheet" integrity="..." crossorigin="anonymous">

</head>

<body>

<h1>Hello, world!</h1>
```



```
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@4.3.1/dist/js/bootstrap.bundle.min.js"
integrity="..." sorigin="anonymous"></script>

</body>

</html>
```

Ви також можете включити Popper та JS окремо. Якщо ви не плануєте використовувати списки, спливаючі вікна або спливаючі підказки, заощаджуйте кілька кілобайт, не включаючи Popper.

```
<script
src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.11.8/dist/umd/popper.min.js"
integrity="..." crossorigin="anonymous"></script>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.3/dist/js/bootstrap.min.js"
integrity="..." crossorigin="anonymous"></script>
```

Що цікаво, Bootstrap вже має JS компоненти

Тож, які компоненти явно вимагають обов'язкового використання JavaScript або Popper від Bootstrap?

- Повідомлення (Alerts) для відхилень
- Кнопки (Buttons) для перемикання станів та функціональності прапорця/радіо
- Карусель (Carousel) для всіх режимів слайдів, елементів керування та індикаторів
- Колапс (Collapse) для перемикання видимості вмісту
- Випадаючі списки (Dropdowns) для відображення та позиціонування (також потрібно Popper)
- Модальні вікна (Modals) для відображення, позиціонування та поведінки прокручування
- Панелі навігації (Navbar) для розширення наших плагінів Collapse та Offcanvas для реалізації адаптивної поведінки
- Навігація (Navs) з плагіном Tab для перемикання панелей вмісту
- Offcanvas для відображення, позиціонування та поведінки прокручування
- Відстеження прокручування (Scrollspy) для поведінки прокручування та оновлень навігації
- Тости (Toasts) для відображення повідомлень та відхилення

- Підказки (Tooltips) і Спливаючі вікна (Popovers) для відображення та позиціонування (також потрібно Popper)

2.5 Огляд скриптової мови програмування JavaScript

Гіпертекстова інформаційна система складається з безлічі інформаційних вузлів, безлічі гіпертекстових зв'язків, визначених на цих вузлах та інструментах маніпулювання вузлами та зв'язками. Технологія World Wide Web - це технологія ведення гіпертекстових розподілених систем в Інтернеті, і, отже, вона повинна відповідати загальному визначенню таких систем. Це означає, що це перелічені вище компоненти гіпертекстової системи повинні бути і в Web.

Web як гіпертекстову систему можна розглядати з двох точок зору. По-перше, як сукупність сторінок, що відображаються, пов'язаних гіпертекстовими переходами (посиланнями - контейнер ANCHOR). По-друге, як безліч елементарних інформаційних об'єктів, що складають сторінки, що відображаються (текст, графіка, мобільний код тощо). В останньому випадку безліч гіпертекстових переходів сторінки - це такий самий інформаційний фрагмент, як і вбудована в текст картинка.

При другому підході гіпертекстова мережа визначається на безлічі елементарних інформаційних об'єктів самими HTML-сторінками, які грають роль гіпертекстових зв'язків. Цей підхід більш продуктивний з точки зору побудови сторінок, що відображаються "на льоту" з готових компонентів.

При генерації сторінок у Web виникає проблема, пов'язана з архітектурою "клієнт-сервер". Сторінки можна генерувати як за клієнта, і за сервера. У 1995 році фахівці компанії Netscape створили механізм управління сторінками на стороні клієнта, розробивши мову програмування JavaScript.

Таким чином, JavaScript – це мова керування сценаріями перегляду гіпертекстових сторінок Web на стороні клієнта. Якщо бути точнішим, то JavaScript – це не лише мова програмування на стороні клієнта. Liveware, прабатько JavaScript, є засобом підстановок на стороні сервера Netscape. Проте найбільшу популярність JavaScript забезпечило програмування за клієнта.

JavaScript



Рисунок 2.5 - Логотип JavaScript у сучасному стеці технологій

Основна ідея JavaScript полягає у можливості зміни значень атрибутів HTML-контейнерів та властивостей середовища відображення у процесі перегляду HTML-сторінки користувачем. При цьому перезавантаження сторінки не відбувається.

Насправді це виявляється у тому, що можна, наприклад, змінити колір фону сторінки чи інтегровану у документ картинку, відкрити нове вікно чи попередження.

Назва "JavaScript" є власністю Netscape. Реалізація мови, здійснена розробниками Microsoft, офіційно називається Jscript. Версії JScript сумісні (якщо бути точним, то не до кінця) з відповідними версіями JavaScript, тобто. JavaScript є підмножиною мови JScript.

JavaScript це стандартизований ECMA (European Computer Manufacturers Association - Асоціація європейських виробників комп'ютерів). Відповідні стандарти мають назви ECMA-262 та ISO-16262. Цими стандартами визначається мова ECMAScript, яка приблизно еквівалентна JavaScript 1.1. Зазначимо, що не всі продажі JavaScript на сьогодні повністю відповідають стандарту ECMA. В рамках цього курсу ми завжди будемо використовувати назву JavaScript.

Для створення механізму керування сторінками на стороні клієнта було запропоновано використовувати об'єктну модель (DOM) документа. Суть

моделі в тому, що кожен HTML-контейнер – це об'єкт, який характеризується трійкою:

- Властивості (properties)
- Методи (methods)
- Події (events)

Об'єктну модель можна назвати як спосіб зв'язку між сторінками і браузером. Об'єктна модель - це представлення об'єктів, методів, властивостей та подій, які присутні та відбуваються у програмному забезпеченні браузера, у вигляді зручному для роботи з ними коду HTML та вихідного тексту сценарію на сторінці. Ми можемо з її допомогою повідомляти наші побажання браузеру і далі відвідувачу сторінки. Браузер виконає наші команди та відповідно змінить сторінку на екрані.

Об'єкти з однаковим набором властивостей, методів та подій поєднуються у класи однотипних об'єктів. Класи – це описи можливих об'єктів. Самі об'єкти з'являються лише після завантаження документа браузером або як результат програми. Про це слід завжди пам'ятати, щоб не звернутися до об'єкта, якого немає.

2.6 Середовище для розробки сайту

Розробка сайту - це непростий та творчий процес, який потребує від використання певного середовища для зручності та ефективності. У цьому розділі ми розглянемо ключові компоненти середовища для розробки веб-сайтів, та оберемо найзручніше для нашого проекту.

Але перед початком роботи потрібно обрати найбільш підходяще середовище, тож для цього ми порівняємо декілька найпопулярніших рішень

Порівняння різних інтегрованих середовищ розробки (IDE) може допомогти визначити, яке з них найкраще підходить для ваших потреб. Розглянемо Visual Studio Code, JetBrains WebStorm, Sublime Text і PHP Storm з точки зору їх позитивних і негативних сторін:

Таблиця 2.6 Порівняння різних інтегрованих середовищ розробки (IDE)

IDE	Позитивні сторони	Негативні сторони
Visual Studio Code	- Безкоштовний і з відкритим вихідним кодом.	- Не так потужний, як інші IDE.
	- Велике спільнота користувачів та розширень.	- Менше функціональних можливостей.
	- Легкий та швидкий.	- Деякі розширення можуть бути менш стабільними.
JetBrains WebStorm	- Повноцінна підтримка JavaScript та інших мов.	- Платний продукт, що потребує ліцензії.
	- Потужні інструменти для розробки веб-додатків.	- Може бути важчим для запуску на слабких комп'ютерах.
Sublime Text	- Швидкий та реагує на взаємодію.	- Відсутність вбудованої підтримки для деяких мов та технологій.
	- Велика кількість плагінів для розширення функціональності.	- Не оновлюється так часто, як інші IDE.
PHP Storm	- Спеціалізується на розробці на PHP.	- Може бути зайвим для проектів, не пов'язаних з PHP.
	- Інтеграція з іншими інструментами розробки.	- Вимагає певного часу для вивчення та освоєння.

Після аналізу плюсів і мінусів кожної IDE можна зробити висновок, що Visual Studio Code виділяється найзручніший та легший продукт серед наведених. Його безкоштовність, широкі можливості розширень, легкість

використання та простота налаштування роблять його привабливим вибором розробки цього проекту. Тому і зупинимося на Visual Studio Code.

2.7. Розробка backlog-у та структури сайту

Розробка веб-сайтів є складним процесом, який вимагає чіткого планування та організації завдань. Одним з ефективних інструментів для керування проектами є створення backlog-у та структури сайту. У цьому підрозділі ми розглянемо як можна використовувати додаток Notion для проектування та трекінгу процесу розробки.



Рисунок 2.6 - Логотип Notion

Backlog у Notion

Backlog - це список завдань або функціоналу, які потрібно виконати або реалізувати в процесі створення сайту. У Notion можна створити backlog за допомогою чекбоксів та випадаючих списків, які ще включають в себе такий функціонал:

- Кожне завдання або функціонал повинні бути чітко та детально описані.
- Можна використовувати теги або мітки для визначення пріоритету кожного елементу backlog-у.
- Відслідковування поточного статусу завдань - "в розробці", "очікує перевірки", "завершено" і т. д.

- Планування та призначення термінів: важливо встановлювати реалістичні терміни виконання завдань і слідкувати за їх виконанням.

Ось так виглядає мій трекінг прогресу виконання робіт з розробки сайту. Але це не повний функціонал, цю платформу можна запрограмувати на більш складніші системи відслідковування, але зараз нам це не потрібно.

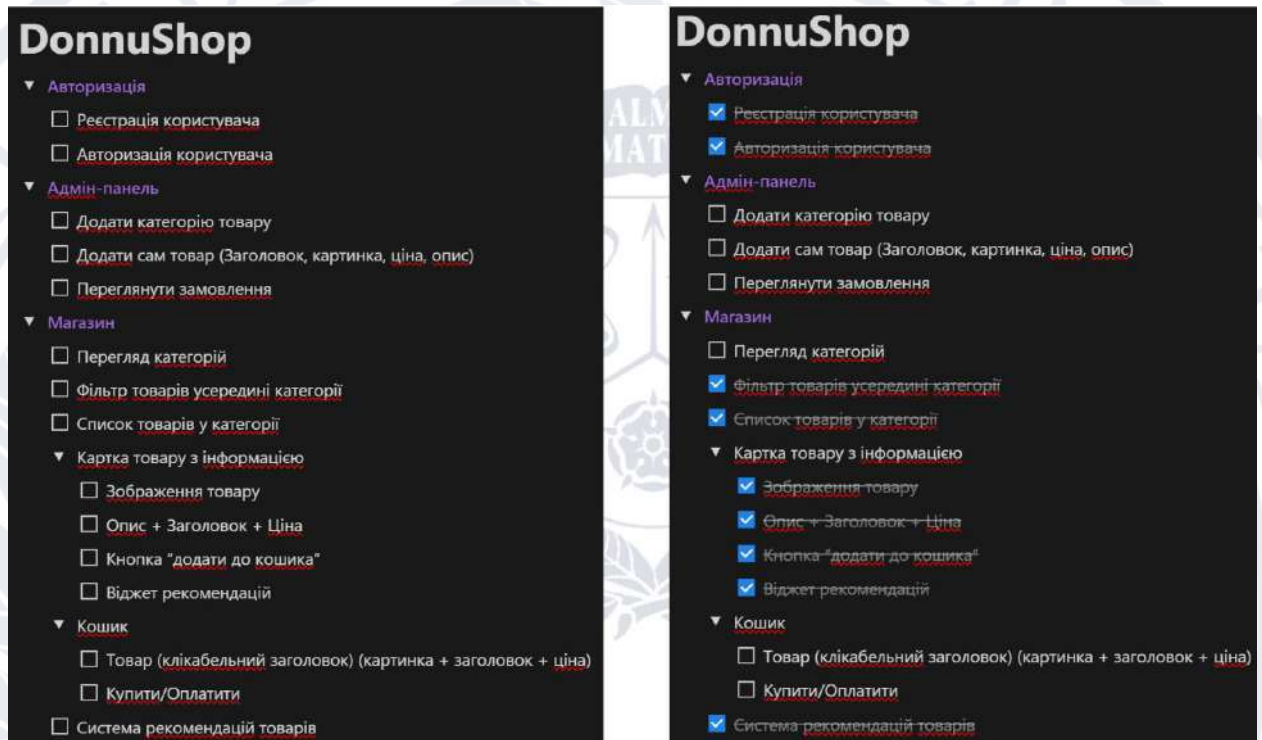


Рисунок 2.7- Використання Notion до та після відслідковування прогресу

Структура сайту в Notion

Структура сайту допомагає визначити ієрархію сторінок, контенту та навігації на веб-сайті. Notion може бути корисним інструментом для створення такої структури:

- **Головна сторінка:** початкова точка для користувачів, де можна розмістити загальну інформацію про сайт.
- **Навігаційне меню:** список посилань на основні розділи або сторінки сайту.
- **Розділи та підсторінки (категорії):** структурування контенту за категоріями та підкатегоріями для зручної навігації.

- **Картка товарів:** опис, ключова інформація, посилання на інші сторінки.

Тож, розробка backlog-у та структури сайту у додатку Notion значно спростила мені процес планування та розробки сайту. Зручний інтерфейс Notion дозволив створити систему для оновлювати та відстежувати етапів розробки.

2.8. Середовище для розгортання онлайн-магазину

У світі веб-розробки два імена стоять окремо, коли йдеться про сервери: Apache і Nginx. Вони обидва відіграють ключову роль у розгортанні веб-додатків та обслуговуванні мільйонів веб-сайтів по всьому світу.

Apache HTTP Server - мабуть, найвідоміший і найстаріший веб-сервер. Його історія розпочалася у 1995 році, коли група розробників, зібравшись під егідою Apache Group, створила його на основі коду NCSA HTTPd. Apache був і залишається вільним та відкритим проектом, що сприяло його широкому поширенню та впровадженню.



Рисунок 2.8 - Логотип Веб-серверу Apache

Nginx – відносно новий гравець, але його вплив у веб-серверному світі значний. Розроблений Ігорем Сисоевим і вперше випущений в 2004 році, Nginx був задуманий як легковажний, швидкий і сервер, що масштабується. Він був спроектований на вирішення проблеми C10k (одночасно обслуговувати 10 000 з'єднань), що було складним завданням для традиційних серверів того часу.

Обидва сервери, Apache та Nginx, надають широкий спектр функціональних можливостей, але їх архітектури та підходи трохи різняться. Apache відомий своєю гнучкістю та розширюваністю. Він підтримує багато модулів, які дозволяють адаптувати його під різні потреби веб-розробників. Apache зазвичай використовується як «всеосяжне» рішення, що підходить для різних сценаріїв розгортання.



Рисунок 2.9-Логотип Веб-сервера nginx

Nginx, з іншого боку, спочатку розроблявся для забезпечення високої продуктивності та ефективного використання ресурсів. Його асинхронна архітектура робить його ідеальним для обробки великої кількості одночасних з'єднань. Він часто використовується як проксі-сервер або зворотний проксі, а також для статичного вмісту і кешування.

Використання для PHP

Обидва сервери популярні як середовище виконання для PHP-програм, але їх конфігурації та підходи до цього відрізняються.

Apache зазвичай використовується з модулем **mod_php**, який інтегрує інтерпретатор PHP у процес сервера. Це простий спосіб запуску PHP-програм, але може бути не так ефективним для обробки великих навантажень.

Nginx, навпаки, найчастіше використовується з **FastCGI Process Manager** (PHP-FPM). PHP-FPM є окремим процесом, який обробляє запити PHP та взаємодіє з Nginx через FastCGI. Це дозволяє краще масштабувати PHP-додатки та керувати ресурсами сервера.

Apache та Nginx, незважаючи на свої відмінності, обидва є потужними інструментами для розгортання веб-додатків, а інколи вони використовуються навіть одночасно!

Вибір між ними залежить від конкретних потреб, наразі наш проект невеликий, тому ми обираємо Apache для запуску PHP

РОЗДІЛ 3

РОЗРОБКА ОНЛАЙН ІНТЕРНЕТ-МАГАЗИНУ ОДЯГУ З ПІДСИСТЕМОЮ РЕКОМЕНДАЦІЙ ТОВАРІВ

3.1. Написання коду інтернет-магазину

Головний банер сайту, який користувач бачить, коли заходить на сайт

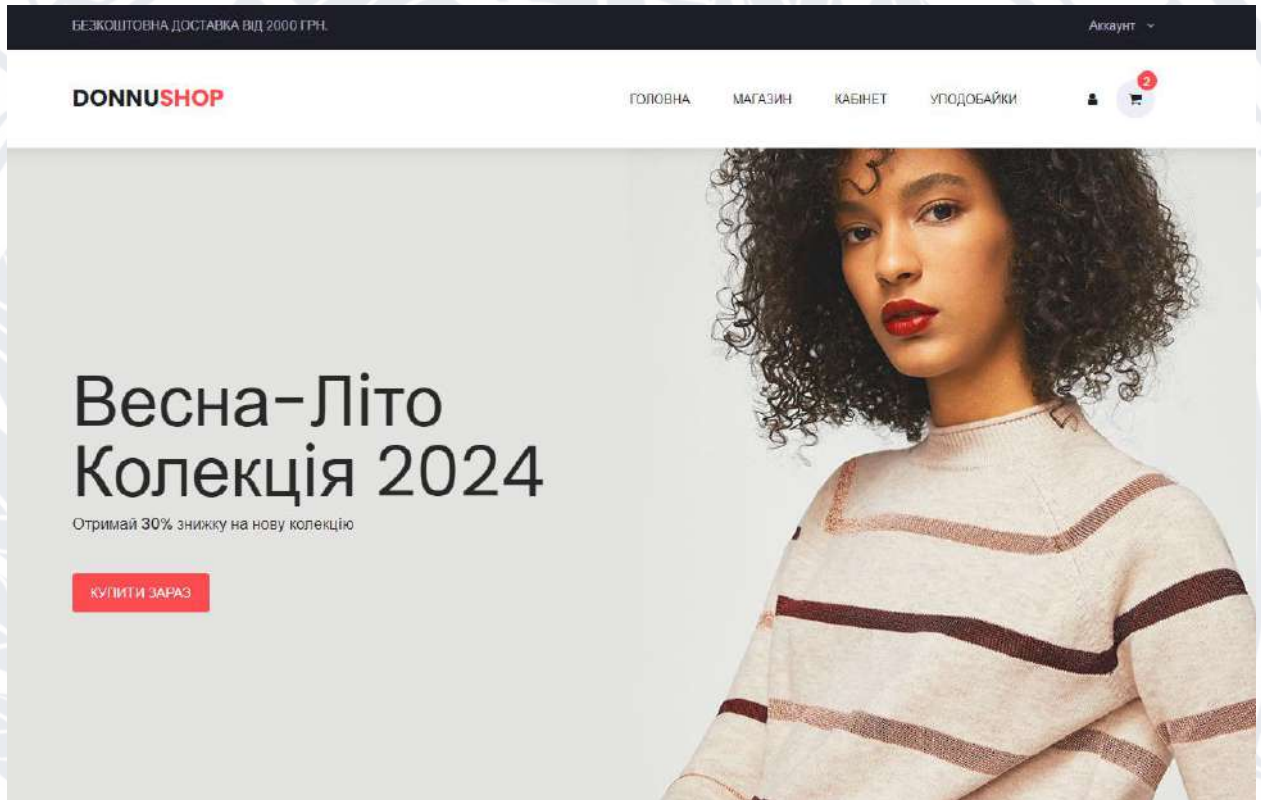


Рисунок 3.1- Головний банер сайту

Це так званий хедер сайту

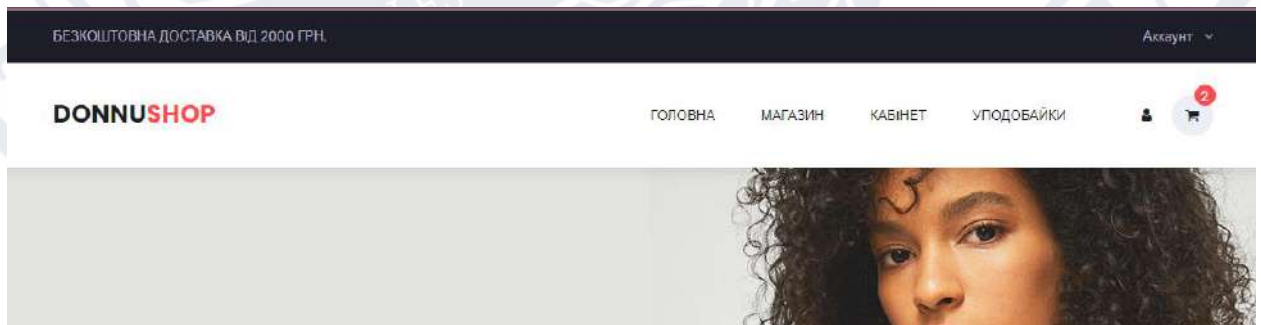


Рисунок 3.2 -Хедер сайту

Навігація по категоріям товарів

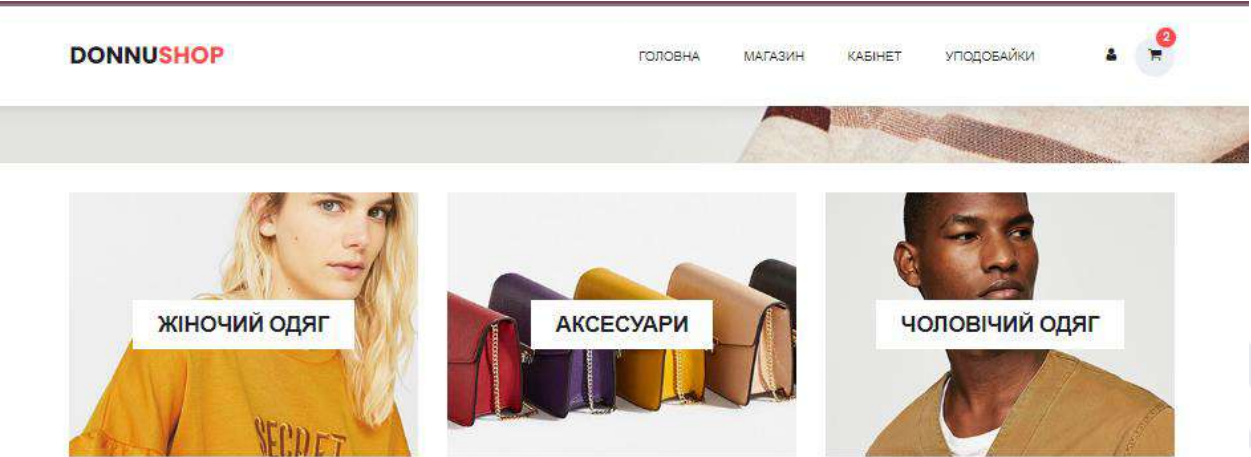


Рисунок 3.3- Навігація по категоріям

Автоматичні рекомендації товарів із нової колекції

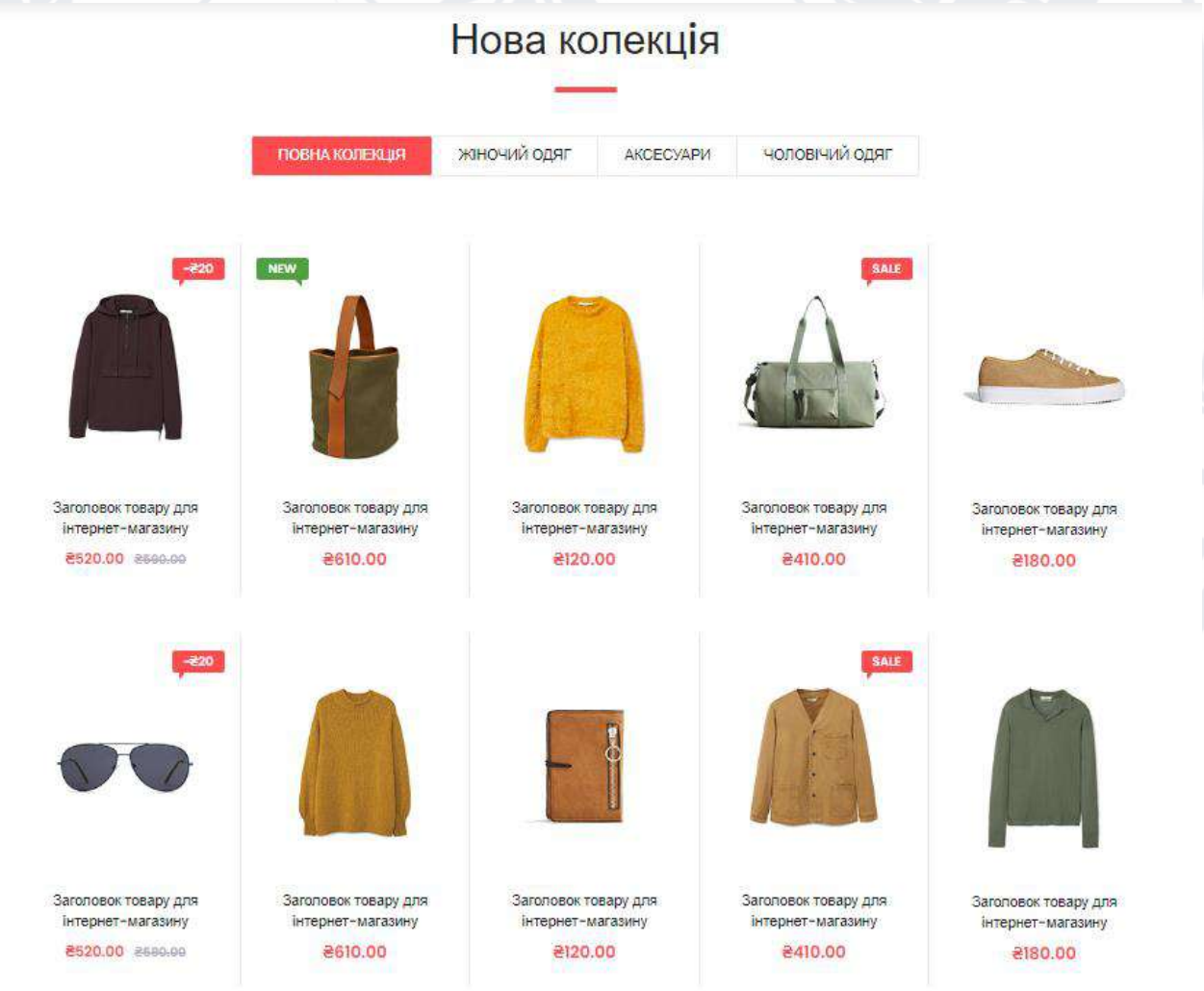


Рисунок 3.4- Рекомендації товарів

Маркетинговий банер для покупця з можливістю скидок

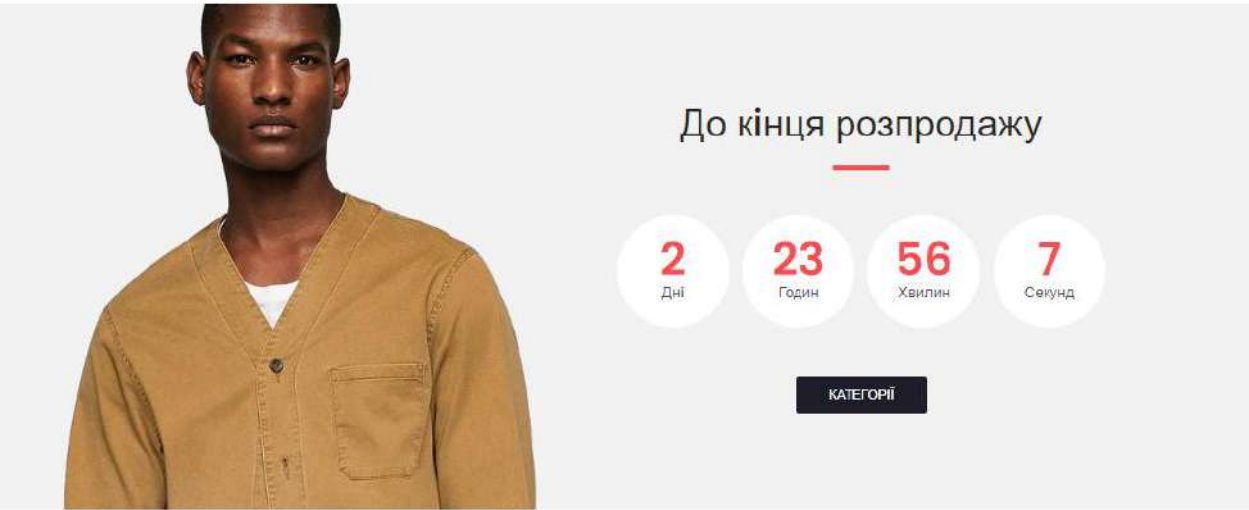


Рисунок 3.5 -Рекомендації товарів

Так званий підвал сайту (footer)

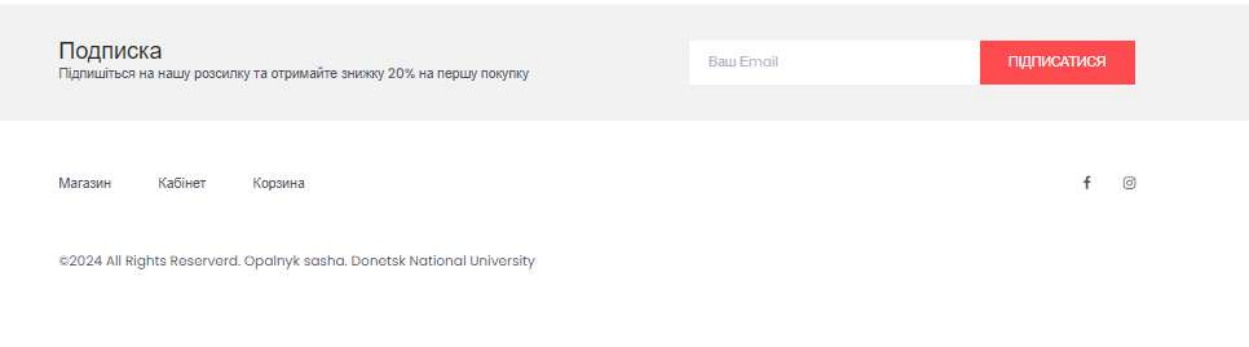


Рисунок 3.6- Підвал сайту

Сторінка “Категорії” з налаштуванням фільтрів

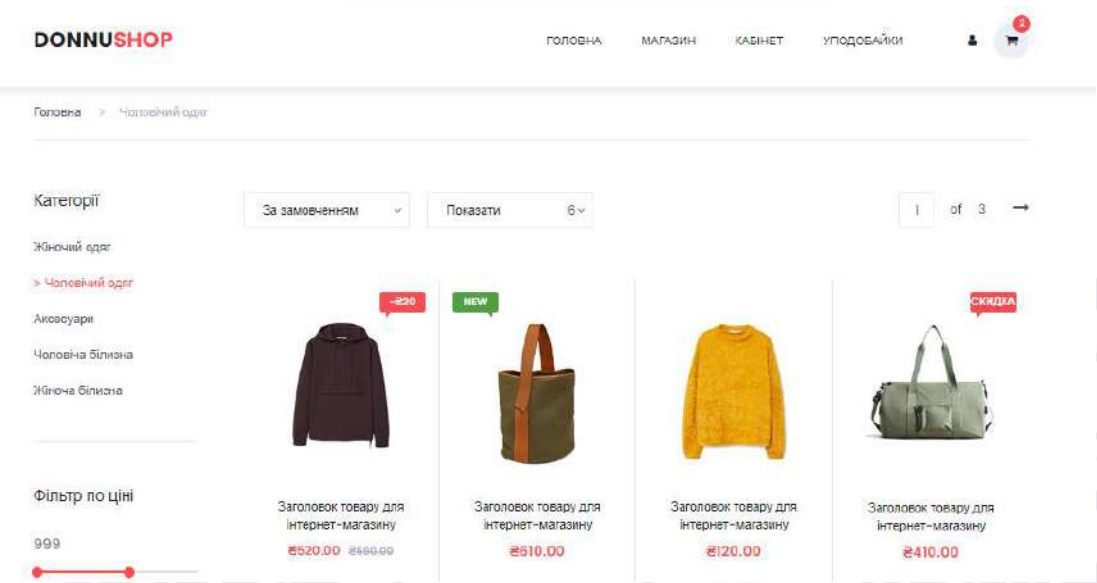


Рисунок 3.7- Категорія з налаштуванням фільтрів

Інформаційна картка товару

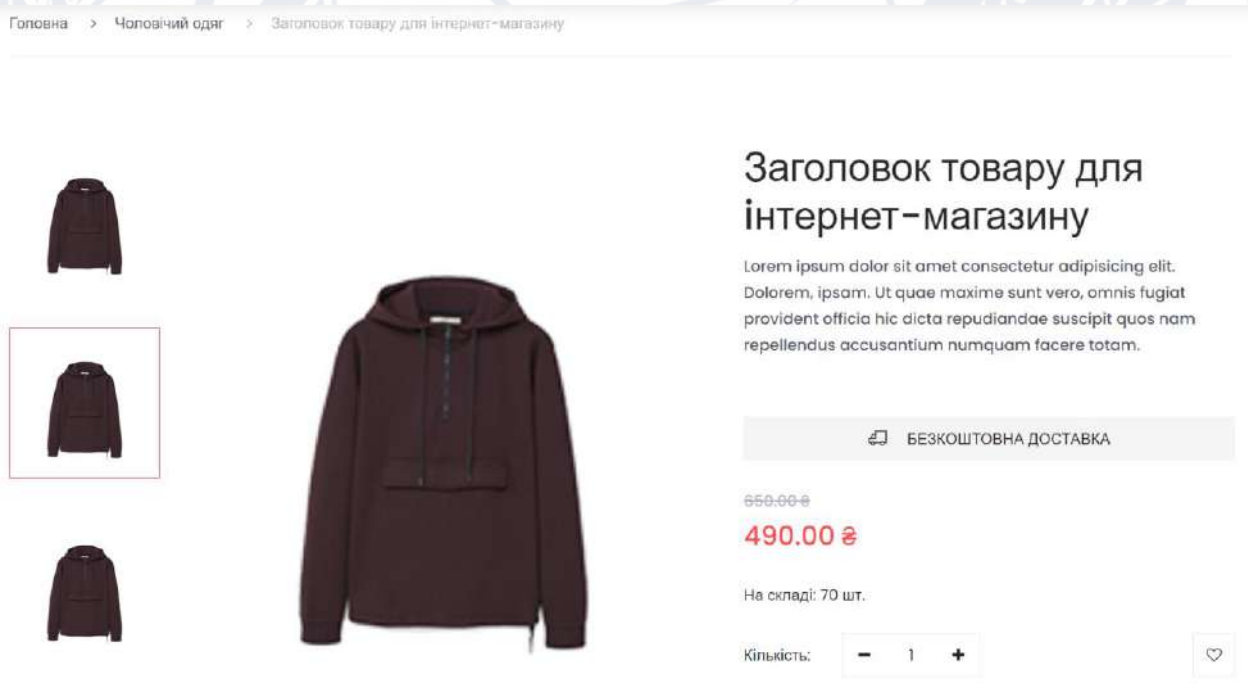


Рисунок 3.8- Інформаційна картка товару

Віджет підсистеми рекомендацій товарів

Рекомендовані товари



Рисунок 3.9- Інформаційна картка товару

3.2. Можливості розвитку та впровадження додаткового функціоналу інтернет-магазину

Інтернет-магазини постійно еволюціонують, щоб задовольняти зростаючі потреби споживачів та забезпечувати їм максимально зручний і приємний досвід покупок. Додавання додаткового функціоналу є важливою частиною цього процесу. У цьому розділі розглянемо можливості розвитку та впровадження таких функцій, як особистий кабінет, уподобайки, порівняння товарів, корзина товарів, знижкові купони та онлайн оплата.

1. Особистий кабінет

Особистий кабінет — це індивідуальний простір користувача на сайті, який дозволяє керувати своїм профілем, замовленнями та налаштуваннями.

Можливості:

- Реєстрація та вхід: Користувачі можуть створити акаунт, зберігаючи свої контактні дані, історію замовлень та переваги.
- Керування замовленнями: Перегляд історії замовлень, відстеження статусу доставки, повторне замовлення.
- Особисті дані: Можливість редагувати контактну інформацію, змінювати пароль, налаштувати повідомлення.
- Збережені адреси: Збереження декількох адрес для швидкого оформлення замовлень.

Впровадження:

- Використання фреймворків для автентифікації (наприклад, Django Allauth для Django, Spring Security для Spring, або відповідні бібліотеки для PHP).
- Розробка інтерфейсу користувача для зручної навігації та керування профілем.

2. Уподобайки

Уподобайки дозволяють користувачам зберігати товари, які їм сподобалися, для подальшого перегляду або покупки.

Можливості:

- Додавання до списку бажань: Користувачі можуть зберігати товари, які вони хотіли б придбати у майбутньому.
- Сповіщення про акції: Повідомлення користувачів про знижки або спеціальні пропозиції на товари зі списку бажань.

Впровадження:

- Створення бази даних для збереження списку бажань користувачів.
- Інтеграція функціоналу додавання/видалення товарів до списку бажань у користувацький інтерфейс.

3. Порівняння товарів

Порівняння товарів дозволяє користувачам одночасно оцінювати декілька продуктів за різними параметрами.

Можливості:

- Додавання товарів для порівняння: Користувачі можуть обирати декілька товарів та порівнювати їх характеристики.
- Візуальне порівняння: Таблиця порівняння з відображенням основних характеристик товарів.

Впровадження:

- Створення функціоналу додавання товарів до списку порівняння.

- Розробка інтерфейсу для візуального порівняння товарів.

4. Корзина товарів

Корзина товарів — ключовий компонент будь-якого інтернет-магазину, що дозволяє користувачам збирати товари для подальшого оформлення замовлення.

Можливості:

- Додавання та видалення товарів: Користувачі можуть додавати товари до корзини, змінювати їх кількість або видаляти.
- Перегляд вартості замовлення: Автоматичний підрахунок загальної вартості замовлення, включаючи податки та доставку.

Впровадження:

- Створення сесійних даних або використання бази даних для зберігання інформації про корзину.
- Інтеграція функціоналу додавання/видалення товарів у користувацький інтерфейс.

5. Знижкові купони

Знижкові купони дозволяють користувачам отримувати знижки на покупки, що стимулює продажі та підвищує лояльність клієнтів.

Можливості:

- Введення коду купона: Користувачі можуть вводити коди купонів при оформленні замовлення для отримання знижки.
- Автоматичні знижки: Відображення доступних знижок без необхідності введення коду.

Впровадження:

- Створення системи генерації та валідації купонів.
- Інтеграція функціоналу використання купонів при оформленні замовлення.

6. Онлайн оплата

Онлайн оплата забезпечує зручність для користувачів, дозволяючи їм оплачувати замовлення безпосередньо на сайті.

Можливості:

- Підтримка різних платіжних систем: Інтеграція з популярними платіжними системами, такими як PayPal, Stripe, банківські картки.
- Безпека платежів: Використання протоколів безпеки, таких як SSL, для захисту платіжної інформації.

Впровадження:

- Інтеграція API платіжних систем у сайт.
- Забезпечення відповідності стандартам безпеки (PCI DSS) для обробки платежів.

Розвиток та впровадження додаткового функціоналу, такого як особистий кабінет, уподобайки, порівняння товарів, корзина товарів, знижкові купони та онлайн оплата, значно покращують користувацький досвід та підвищують ефективність інтернет-магазину. Ці функції забезпечують зручність, гнучкість та безпеку для користувачів, стимулюючи їх до більш частих покупок та підвищуючи лояльність до бренду. Правильне впровадження цих функцій допоможе створити конкурентоспроможний та успішний інтернет-магазин.

ВИСНОВКИ

У даній дипломній роботі ми успішно реалізували онлайн інтернет-магазин одягу з підсистемою рекомендацій товарів, що є важливим кроком у розвитку сучасної електронної комерції. Було виконано низку завдань, спрямованих на створення функціонального та ефективного онлайн-магазину, який задовольняє потреби сучасних споживачів.

Досягнуті результати:

1. Аналіз вимог та потреб цільової аудиторії:

- Проведено детальний аналіз критеріїв, які впливають на вибір товарів користувачами, включаючи їхні уподобання, попередні покупки та поведінку на сайті.
- Зібрано та проаналізовано дані про користувачів, що дало змогу зрозуміти основні фактори, що впливають на їхні покупки.

2. Вибір технологій для реалізації підсистеми рекомендацій:

- Проведено огляд сучасних технологій та методів аналізу даних, що використовуються для побудови рекомендаційних систем.
- Обрано ефективні алгоритми рекомендацій, які враховують великий обсяг даних та забезпечують точні й релевантні рекомендації для кожного користувача.

3. Розробка та впровадження інтернет-магазину:

- Розроблено функціональний інтернет-магазин одягу з інтегрованою підсистемою рекомендацій.
- Здійснено впровадження ключових функціональних можливостей, таких як особистий кабінет, уподобайки, порівняння товарів, корзина товарів, знижкові купони та онлайн оплата.

4. Тестування та аналітика використання:

- Проведено комплексне тестування системи, що дозволило перевірити ефективність та точність роботи рекомендаційної підсистеми.
- Здійснено аналіз отриманих даних для внесення виправлень та покращень до алгоритмів рекомендацій, з метою підвищення їхньої точності та корисності для користувачів.

5. Покращення користувацького досвіду:

- Система рекомендацій значно полегшила процес вибору товарів для користувачів, надаючи персоналізовані пропозиції на основі їхніх уподобань та історії покупок.

- Покращення користувацького досвіду сприяло збільшенню конверсії та підвищенню обсягів продажів для бізнесу.

6. Аналіз впливу інноваційного підходу на ефективність електронної комерції:

- Досліджено сучасні тенденції в електронній комерції та методи побудови рекомендаційних систем.
- Виявлено позитивний вплив впровадження рекомендаційної підсистеми на конкурентоспроможність інтернет-магазину в умовах сучасного ринку.

Таким чином, розробка онлайн інтернет-магазину одягу з підсистемою рекомендацій товарів виявилася успішною та ефективною. Впроваджений функціонал забезпечив покращення користувацького досвіду, збільшення конверсії та підвищення ефективності електронної комерції. Результати дослідження можуть бути використані для подальшого розвитку та оптимізації інтернет-магазинів, що сприятиме їхній конкурентоспроможності та успішному функціонуванню в умовах цифрової трансформації торгівлі.

СПИСОК ВИКОРИСТАНИХ ПОСИЛАНЬ

1. Б. Лоусон, Р. Шарп, Вивчаємо HTML 5, 2020, 272 с.
2. Лабберс П. HTML 5 для професіоналів: потужні інструменти для розробки сучасних веб-додатків, 2011, 272 с.
3. BuiltSait. HTML 5: [Електронний ресурс]. – Режим доступу: <http://buildsait.pp.ua/index.php/news-web-/127-html5/>
4. Макфарланд Д, Велика книга CSS3, 3-тє видання, 2014, 608 с.
5. Метт Зандстра, PHP. Об'єкти, шаблони та методики програмування, 2022, 866 с.
6. Дмитро Котеров, PHP. Створення інтернет-магазинів, Андрій, 2018, 1088 с.
7. Річ Боуен, "Apache. Повне керівництво", 2002, 350 с.
8. Шаландер Чристіан, Адміністрування Apache. Самовчитель, 2021, 250 с.
9. Девід Фланаган, JavaScript. Докладний посібник 5-е вид., 2020, 992 с.
10. Марейн Хавербеке, Експресивний JavaScript, 2021, 280 с.
11. Стоян Стефанов, JavaScript. Шаблони, 2018, 272 с.
12. Джон Дакетт, HTML та CSS. Дизайн і побудова веб-сайтів, 2020, 740 с.
13. Дженнифер Нідерст Роббінс, HTML та CSS. Вебдизайн для початківців, 2016, 302 с.
14. Джон Дакетт, HTML та CSS. Веб-дизайн і розробка, 2021, 480 с.
15. Метт Зандстра, PHP в дії. Об'єктно-орієнтоване програмування, 2015, 624 с.
16. Робін Ніксон, PHP, MySQL, JavaScript і HTML5. Все в одному 4-е вид., 2018, 768 с.
17. Елізабет Зобейд, Apache. Адміністрування і налаштування
18. Джон Пол Мюррей, JavaScript для початківців, 2022, 320 с.
19. Етан Браун, JavaScript. Підручник з програмування, 2022, 368 с.



ДОДАТКИ

ДОДАТОК А

Глосарій термінів, які використовувались у роботі

Маршрутизація – це процес визначення оптимального шляху передачі даних у комп'ютерній мережі. Маршрутизатори приймають рішення про передачу даних на основі інформації про мережеві протоколи та поточний стан мережі.

Веб-сервер - це програмне забезпечення, яке обробляє запити від клієнтських пристроїв (наприклад, веб-браузерів) та надає веб-сторінки та інші ресурси у відповідь.

HTML (HyperText Markup Language) - це мова розмітки, яка використовується для створення веб-сторінок. Він визначає структуру та вміст сторінки за допомогою тегів та атрибутів.

JavaScript - це мова програмування, яка дозволяє додавати інтерактивність та динамічну поведінку на веб-сторінках. Він часто використовується для створення клієнтської частини веб-застосунків.

CSS (Cascading Style Sheets) - це мова стилів, що визначає зовнішній вигляд та форматування веб-сторінок. Він використовується для завдання кольорів, шрифтів, розташування елементів та інших аспектів дизайну.

Apache – це найпопулярніший веб-сервер у світі. Він надає надійне та масштабоване оточення для розміщення веб-сайтів та додатків.

Nginx - це високопродуктивний веб-сервер і проксі-сервер. Він відомий своєю ефективністю та здатністю обробляти велику кількість одночасних підключень.

Bootstrap – це популярний фреймворк для розробки веб-інтерфейсів. Він надає набір готових компонентів та стилів, які спрощують створення чуйних та стильних веб-сайтів.

Хостинг - це послуга надання простору на сервері для розміщення веб-сайтів та файлів. Хостинг-провайдери забезпечують доступність веб-сайту в Інтернеті.

Домен - це унікальне ім'я, яке ідентифікує веб-сайт в Інтернеті. Він складається з імені та доменного розширення (наприклад, .com, .org, .ru) та використовується для доступу до веб-сайту.

PHP - це скриптова мова програмування, яка часто використовується для розробки веб-додатків. Він може бути вбудований у HTML-код та виконуватися на сервері.

Backlog - це список завдань або вимог, які очікують на виконання. Він використовується для відстеження та управління роботою в проекті чи команді.

Notion – це платформа для організації та спільної роботи над проектами. Вона надає інструменти для створення нотаток, завдань, баз даних та інших типів контенту.

IDE (Integrated Development Environment) – це інтегроване середовище розробки, яке надає розробникам інструменти для написання, налагодження та тестування програмного коду.

Плагін - це програмне розширення, яке додає додаткові функції чи можливості до іншого програмного забезпечення. Вони часто використовуються у веб-браузерах або текстових редакторах.

E-commerce (електронна комерція) – це процес купівлі та продажу товарів та послуг через інтернет. Він включає онлайн-магазини, електронні платіжні системи та інші онлайн-торговельні платформи.

UpSale – це стратегія маркетингу, яка спрямована на збільшення середнього чека покупця шляхом пропозиції додаткових товарів чи послуг при оформленні покупки.

HTTP (Hypertext Transfer Protocol) – це протокол передачі даних у мережі, який використовується для обміну інформацією між веб-серверами та клієнтськими пристроями.

WWW (World Wide Web) – це система пов'язаних між собою документів та інших ресурсів, доступних через Інтернет. Вона є глобальною мережею веб-сайтів.

SSL (Secure Sockets Layer) – це протокол шифрування, який забезпечує безпечну передачу даних між веб-сервером та клієнтським пристроєм. Він використовується для захисту конфіденційності та цілісності інформації.

HTTPS (Hypertext Transfer Protocol Secure) – це захищена версія протоколу HTTP, яка використовує SSL або його послідовника TLS для забезпечення безпечної передачі даних.

Favicon - це невелика іконка, яка відображається поряд із заголовком веб-сторінки у вкладці браузера або у списку закладок. Він допомагає ідентифікувати веб-сайт візуально.

Конверсія - це процес перетворення відвідувачів веб-сайту на діючих клієнтів або виконання бажаної дії, такої як купівля товару або підписки на розсилку.

Споживач – це людина чи організація, яка використовує товари чи послуги. У контексті бізнесу споживач є цільовою аудиторією та основним джерелом доходу.

Інтернет-магазин – це веб-сайт, який пропонує товари або послуги для покупки через Інтернет. Користувачі можуть вибирати та замовляти товари онлайн, а потім отримувати їхню доставку.

Електронний бізнес - це форма бізнесу, яка ґрунтується на використанні інформаційних технологій та інтернету. Він включає в себе онлайн-торгівлю, електронні платіжні системи, цифровий маркетинг та інші аспекти електронної комерції.

CGI-програма (Common Gateway Interface) – це програма, яка виконується на веб-сервері та обробляє запити від клієнтських пристроїв. Вона дозволяє взаємодіяти із зовнішніми програмами та базами даних.

SQL (Structured Query Language) – це мова програмування, яка використовується для роботи з базами даних. Він дозволяє виконувати запити, додавати, змінювати та видаляти дані в базі даних.

База даних: База даних - це організована колекція даних, яка зберігається та керується за допомогою спеціального програмного забезпечення. Вона використовується для зберігання та обробки великих обсягів інформації.

IMAP (Internet Message Access Protocol): протокол доступу до поштової скриньки, який дозволяє користувачам переглядати повідомлення, що зберігаються на поштовому сервері. IMAP забезпечує можливість керування повідомленнями на сервері без завантаження на пристрій користувача.

POP3 (Post Office Protocol version 3): протокол поштового відділення, який використовується для завантаження електронної пошти з сервера на пристрій кінцевого користувача. За замовчуванням повідомлення видаляються з сервера після завантаження, але цю поведінку можна налаштувати.

SMTP (Simple Mail Transfer Protocol): протокол передачі пошти, який використовується для надсилання електронних листів між серверами електронної пошти. SMTP забезпечує механізми для відправки, пересилання та доставки листів.

URL (Uniform Resource Locator): Однозначна адреса, яка вказує на ресурс у мережі Інтернет. URL зазвичай включає протокол передачі даних (наприклад, HTTP), доменне ім'я та шлях до конкретного ресурсу.

URI (Uniform Resource Identifier): Загальний ідентифікатор ресурсу, який використовується для ідентифікації будь-якого ресурсу в мережі, включаючи URL-адресу. URI включає специфікацію протоколу, доменне ім'я та шлях до ресурсу.

URN (Uniform Resource Name): Уніфіковане ім'я ресурсу, яке є унікальним та постійним ідентифікатором для ресурсу в мережі. URN забезпечує унікальне ім'я для ресурсу, незалежно від його розташування або розташування.

ДЕКЛАРАЦІЯ

про дотримання академічної доброчесності

Я, _____

_____ Повністю вказується ПІБ та статус (посада для працівників, освітня (освітньо-наукова) програма – для здобувачів вищої освіти) що нижче підписалась/підписався, розуміючи та підтримуючи загальновизнані засади справедливості, доброчесності та законності,

ЗОБОВ'ЯЗУЮСЬ:

дотримуватися принципів та правил академічної доброчесності, що визначені законодавством України, локальними нормативними актами Донецького національного університету імені Василя Стуса, положеннями, правилами, умовами, визначеними іншими суб'єктами, та не допускати їх порушення.

ПІДТВЕРДЖУЮ:

що мені відомі положення статті 42 Закону України «Про освіту»; що у даній роботі не представляла/представляв чийсь роботи повністю або частково як свої власні. Там, де я скористалася/скористався працею інших, я зробила/зробив відповідні посилання на джерела інформації; що дана робота не передавалась іншим особам і подається вперше, не порушує авторських та суміжних прав закріплених статтями 21-25 Закону України «Про авторське право та суміжні права», а дані та інформація не отримувались в недозволений спосіб. УСВІДОМЛЮЮ: що ця робота може бути перевірена університетом на плагіат або інші порушення академічної доброчесності, в тому числі з використанням спеціалізованих сервісів; що у разі порушення академічної доброчесності, до мене можуть бути застосовані процедури, передбачені законодавством України та Кодексом академічної доброчесності та корпоративної етики Донецького національного університету імені Василя Стуса, іншими локальними нормативними актами університету, та я можу бути притягнута/притягнутий до академічної відповідальності.

(дата)

(підпис)