

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ
ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ
КАФЕДРА КОМП'ЮТЕРНИХ НАУК**

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «Сайт Інтернет магазину Телемарт на основі JavaScript»

на здобуття освітнього ступеня магістра
зі спеціальності 122 Комп'ютерні науки
(код, найменування спеціальності)
освітньо-професійної програми Комп'ютерні науки
(назва)

*Кваліфікаційна робота містить результати власних досліджень.
Використання ідей, результатів і текстів інших авторів мають посилання
на відповідне джерело*

(підпис)

Дзюба О.В.
(Ім'я, ПРІЗВИЩЕ здобувача)

Виконав:
здобувач вищої освіти
група КНДМ-62

Олексій ДЗЮБА

Керівник:
науковий ступінь,
вчене звання

Олена ШИКУЛА
д.ф.-м.н., професор

Рецензент:
науковий ступінь,
вчене звання

(Ім'я, ПРІЗВИЩЕ)

Київ 2025

**ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ**
Навчально-науковий інститут інформаційних технологій

Кафедра Комп'ютерних наук

Ступінь вищої освіти Магістр

Спеціальність Комп'ютерні науки

Освітньо-професійна програма Комп'ютерні науки

ЗАТВЕРДЖУЮ

Завідувач кафедру Комп'ютерних наук

_____ Віктор ВИШНІВСЬКИЙ
« _____ » _____ 2025 р.

**ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ**

_____ Дзюбі Олексію Валерійовичу

(прізвище, ім'я, по батькові здобувача)

1. Тема кваліфікаційної роботи: Сайт Інтернет магазину Телемарт на основі JavaScript

керівник кваліфікаційної роботи Олена ШИКУЛА д.ф.-м.н., професор,

(Ім'я, ПРИЗВИЩЕ науковий ступінь, вчене звання)

затверджені наказом Державного університету інформаційно-комунікаційних технологій від «15» жовтня 2024р. №320

2. Строк подання кваліфікаційної роботи «15» грудня 2024р.

3. Вихідні дані до кваліфікаційної роботи: науково-технічна література, веб-сайт на основі Java Script. Параметри для створення власного веб-сайту.

4. Зміст розрахунково-пояснювальної записки (перелік питань, які потрібно розробити)

Дослідження та аналіз існуючих веб-сайтів інтернет магазинів.

Дослідження середовищ розробки веб-сайтів.

Розробка власного прототипу веб-сайту інтернет магазину.

5. Перелік графічного матеріалу: *презентація*

1. Огляд технології SDN: архітектура, порівняння, приклади використання.
2. Огляд середовища для моделювання та аналізу Mininet.
3. Реалізована топологія мережі.
4. Фрагменти програмного коду.

6. Дата видачі завдання «16» вересня 2024 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1	Аналіз наявної науково-технічної літератури	16.09-10.10.24	
2	Огляд існуючих веб-сайтів інтернет магазинів	11.10-13.10.24	
3	Дослідження та порівняння середовищ розробки	14.10-16.10.24	
4	Налаштування середовища розробки	17.10-20.10.24	
5	Розробка базового дизайну веб-сайту	11.10-13.10.24	
6	Розробка веб-сайту	14.10-05.12.24	
7	Розробка демонстраційних матеріалів	06.12-13.12.24	

Здобувач вищої освіти

(підпис)

Олексій ДЗЮБА

(Ім'я, ПРІЗВИЩЕ)

Керівник

кваліфікаційної роботи

(підпис)

Олена ШИКУЛА

(Ім'я, ПРІЗВИЩЕ)

РЕФЕРАТ

Текстова частина кваліфікаційної роботи на здобуття освітнього ступеня магістра: 44с., 12 рис., 26 джерел.

Наукове завдання – Дослідити сучасну розробку веб-сайтів інтернет магазинів, розробити власний веб-сайт інтернет магазину електроніки.

Мета роботи – За допомогою JavaScript створити власний веб-сайт інтернет магазину електроніки.

Об'єкт дослідження – Розробка веб-сайту інтернет магазину.

Предмет дослідження – середовища розробки, мови програмування, існуючі веб-сайти інтернет магазини.

Короткий зміст роботи:

Здійснено аналіз наповнення веб-сайтів інтернет магазинів, її структури, аналіз поняття E-Commerce (електронної комерції). Досліджено середовища розробки веб-сайтів, мов програмування для написання.

За допомогою JavaScript з використанням бази даних розроблено власний веб-сайт інтернет магазину електроніки.

Веб-сайт допоможе просувати власний магазин електроніки.

Ключові слова: ІНТЕРНЕТ МАГАЗИН, ВЕБ-САЙТ, BRACKETS, HTML, CSS, JAVASCRIPT

ABSTRACT

Text part of the master's qualification work: 44p., 12 img., 26 sources.

Research Task:

Research the modern development of websites for online stores, develop your own website for an online electronics store.

Research Goal:

Use JavaScript to create your own online electronics store website.

Object of Research:

Development of an online store website.

Subject of Research:

Development environments, programming languages, and existing e-commerce websites.

Summary of the Work:

An analysis of the contents of online store websites, their structure, and an analysis of the concept of E-Commerce (electronic commerce) was carried out. Researched web site development environments, programming languages for writing.

An online electronics store's own website was developed using JavaScript using a database.

A website will help promote your own electronics store.

keywords: E-COMMERCE WEBSITE, WEBSITE, BRACKETS, HTML, CSS, JAVASCRIPT

ЗМІСТ

ВСТУП

1 ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Поняття E-Commerce (електронної комерції)

1.2 Поняттята характеристики пов'язанні з інтернет-магазином з точки зору різновидів веб-сайтів.

2 ОПИС ІНСТРУМЕНТАЛЬНОЇ БАЗИ

2.1 Послідовність етапів розробки веб-сайту інтернет-магазину

2.2 Опис інструментів розробки

3 ОПИС РОЗРОБКИ ВЛАСНОГО ПРОДУКТУ

3.1 Структура веб-сайту інтернет-магазину

3.2 Опис необхідних мов програмування та їх використання

3.3 Опис власного веб-сайту інтернет магазину

ВИСНОВКИ

ПЕРЕЛІК ПОСИЛАНЬ

ДОДАТОК А. ФРАГМЕНТИ ПРОГРАМНОГО КОДУ

ДОДАТОК Б. ДЕМОНСТРАЦІЙНИЙ МАТЕРІАЛ

ВСТУП

Це дослідження викликано змінами, що відбуваються у сфері технологій. Технології постійно розвиваються, не стоять на місці, і в сфері продажів Інтернет-магазини стають все популярнішими.

Зараз існує безліч Інтернет-магазинів, і багато компаній прагнуть перенести свій бізнес в Інтернет. Однак серед величезної кількості веб-ресурсів знайти дійсно якісні стає досить складно. Крім того, сьогодні важко знайти спеціаліста, який може підтримувати створені веб-ресурси, якщо вони написані різними мовами програмування.

Сучасні засоби створення сайтів дозволяють розробляти сайти та Інтернет-магазини з використанням систем управління контентом (CMS), що дає можливість змінювати, доповнювати, видаляти інформацію на сайті без безпосередньої участі програмістів. Перевага цього методу полягає в тому, що знання програмування зазвичай не потрібні. Але програмувати веб-сайт вручну знімає всі обмеження у використанні конструкторів веб-сайтів.

Веб-ресурси необхідно розвивати таким чином, щоб їх можна було легко та швидко редагувати. Контент веб-сайту не потребує додаткових знань і досвіду. Втім, створення Інтернет-магазину передбачає вирішення кількох завдань.

Потрібно впливати на конкретну цільову аудиторію та залучати постачальників товарів. Структура та напрямок самого сайту можуть бути найрізноманітнішими: від продажу конкретних товарів до надання послуг та навичок. Високий рівень виконання та хороша функціональність сайту визначають його головний критерій — ефективність. Дуже важливо створити Інтернет-магазин з привабливим дизайном, але водночас важливо розробити зручний та ергономічний інтерфейс для користувачів.

Можливість інформувати про майбутні знижки, акції та інші події також не буде зайвою. Необхідно проводити дослідження ринку, стежити за новітніми розробками в галузі електроніки. Актуальна інформація про асортимент продукції

та її наявність надзвичайно важлива. Зайти на сайт набагато простіше, ніж зателефонувати, а оформити замовлення ще простіше. Система електронного маркетингу, швидкість і простота та доступність реєстрації, замовлення через особистий кабінет замовника, можливість створити цілий інформаційний портал розширює бізнес-можливості.

Сайт Інтернет-магазину — це сучасний транзакційний канал. Завдяки Інтернет-магазину кожен може продавати будь-які товари. Ви можете пропонувати товари чи послуги широкій аудиторії за допомогою Інтернету. Інтернет-магазини дають змогу покупцям економити час, гроші та зусилля. Як свідчить статистика, онлайн-покупки зростають все більше не лише в Україні, але й в усьому світі. Все більше людей здійснюють покупки в Інтернет-магазинах.

Метою дипломної роботи є визначення та обґрунтування сучасних методів розробки веб-сайту Інтернет-магазину електроніки. Основні завдання дослідження:

1. Проаналізувати термінологічний апарат дослідження.
2. Визначити специфіку веб-сайтів Інтернет-магазинів.
3. Провести аналіз різновидів сучасних методів розробки веб-сайтів Інтернет-магазинів.
4. Визначити основні етапи створення Інтернет-магазину.
5. Проаналізувати основні вимоги до створення веб-сайту Інтернет-магазину.
6. Оцінити програмне забезпечення для створення веб-сайту.
7. Розробити концепцію, дизайн та програмну реалізацію веб-сайту Інтернет-магазину електроніки.

1 ОПИС ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Поняття E-Commerce (електронної комерції)

Електронна комерція, більш відома як e-commerce, є однією з найбільш динамічних сфер сучасного бізнесу, яка змінила уявлення про купівлю та продаж товарів і послуг. Цей термін охоплює всі форми бізнес-операцій, які здійснюються через Інтернет, зокрема обмін фінансовими ресурсами та даними для здійснення транзакцій. Первісно електронна комерція асоціювалася саме з продажами фізичних товарів онлайн, але наразі спектр її дії значно ширший, включаючи найрізноманітніші форми взаємодії в бізнес-середовищі.

З кожним роком обсяг операцій у сфері електронної комерції невідменно зростає, захоплюючи дедалі більшу частку ринку. Цей ріст не обмежується тільки швидкістю та обсягом продажів, але й різноманітністю послуг, які можуть включати бронювання, уточнення замовлень, управління транзакціями. До сфери e-commerce також належать банківські послуги та системи електронних платежів, що забезпечують велику гнучкість та доступність для кінцевого користувача.

Історія електронної комерції почалася з першої онлайн-транзакції, яку здійснив музикант Стінг через свою платформу NetMarket. Це стала перша відома угода між споживачем і компанією через Інтернет, яка дала початок тому, що ми нині знаємо як e-commerce.

Моделі електронної комерції досить різноманітні, проте можна виділити чотири основні типи:

- Бізнес для бізнесу (B2B): коли одна компанія продає товари чи послуги іншій компанії, забезпечуючи великий обсяг та спеціалізовані рішення.
- Споживач для споживача (C2C): такі платформи дозволяють споживачам продавати свої товари чи послуги іншим споживачам, що сприяє взаємодії та обміну між користувачами.

- Бізнес для споживача (B2C): у цій моделі підприємства продають товари чи надають послуги безпосередньо кінцевим споживачам, створюючи базис для широкомасштабного продажу онлайн.
- Споживач для бізнесу (C2B): цей тип вказує на ситуації, коли споживачі пропонують свої власні продукти чи послуги бізнесу, зворотно впливаючи на ринок.

Для успішного функціонування в сегменті електронної комерції онлайн-бізнеси мають включати різні компоненти, такі як сучасний веб-сайт або інтернет-магазин, ефективні канали для залучення трафіку, включаючи SEO та SMM стратегії, а також інструменти реклами. Продажі та підтримка клієнтів є основою, що включає служби закупівлі, логістики і доставки, а також системи обробки замовлень і CRM для управління взаємодією з клієнтами.

- Електронна комерція може реалізовуватися у різних формах, таких як:
- Роздрібна торгівля: підприємства продають товари напряму клієнтам, усуваючи посередників і зменшуючи витрати.
- Оптова торгівля: продаж товарів великими партіями іншим роздрібним продавцям для подальшого продажу кінцевим споживачам.
- Дропшипінг: модель бізнесу, в якій підприємство продає товари, що виготовлені іншими постачальниками, безпосередньо споживачам, зменшуючи ризики та інвентаризаційні затрати.
- Краудфандинг: стратегія залучення фінансів від споживачів для попереднього фінансування розробки товару.
- Підписки: запропонування товарів чи послуг за умовами автоматичного регулярного оновлення та оплати, що гарантує стабільні доходи та постійний ринок збуту.
- Цифрові продукти: товари, які завантажуються, як-от електронні книги чи програмне забезпечення,

Типи платформ Інтернет-магазинів

В залежності від моделі електронної комерції, можна виділити кілька основних типів платформ для Інтернет-магазинів:

Онлайн-магазин — один із найпоширеніших і найвідоміших форм електронної комерції. Комерційні компанії або окремі підприємці мають можливість пропонувати широкий асортимент різноманітних товарів, послуг чи сервісів. Для зручності покупців платформи зазвичай надають можливість фільтрації за базовими критеріями, такими як категорія продукції, ціна та інші характеристики.

Платформа для розміщення оголошень — цей формат має модель C2C, що стимулює взаємодію між звичайними споживачами. На такій платформі користувачі можуть створювати власні оголошення про продаж товарів або надання послуг. Продавець вказує категорію товару, короткий опис та виставляє свою ціну. Це дозволяє потенційним покупцям швидко знаходити відповідні пропозиції.

Дисконтні сайти. Ці платформи агрегують та організують пропозиції від різних компаній і приватних осіб, аби надати покупцям можливість отримувати знижки. Зазвичай вони діють через надання промокодів або спеціальних пропозицій, які заохочують користувачів до покупок.

Інтернет-аукціони. На таких платформах товари виставляються на продаж у вигляді лотів без заздалегідь фіксованої ціни, але із визначеним часом завершення аукціону. У цей період потенційні покупці беруть участь у торгах, що створює конкурентне середовище.

Інтернет-магазини є втіленням концепції електронної комерції. Давайте детальніше розглянемо концепцію та ключові особливості Інтернет-магазинів.

1.2 Поняття та характеристики пов'язані з інтернет-магазином з точки зору різновидів веб-сайтів

Один із ключових способів комерційного використання веб-сайтів полягає в продажу товарів і послуг через Інтернет. Цей підхід до ведення бізнесу набуває все більшого поширення і став невід'ємною частиною сучасного комерційного середовища.

Компанії та індивідуальні підприємці, які розробляють свої веб-сайти, частіше всього використовують їх для забезпечення доступу до інформації про свою діяльність, асортимент товарів, пропоновані послуги та інші необхідні для споживачів відомості.

На веб-сайтах зазвичай розміщується широка інформація про товари та послуги, що включає деталі характеристик, ціни, гарантійні зобов'язання, а також способи доставки. Це робить дані легкодоступними для потенційних клієнтів, які можуть швидко зорієнтуватись у виборі серед численних пропозицій. Інтернет-магазини виконують важливу роль у сучасному економічному середовищі, оскільки вони надають компаніям ефективні інструменти для збільшення обсягів продажів, зростання товарообігу та підвищення рівня задоволеності покупців. Існує декілька визначень Інтернет-магазину, які допомагають зрозуміти цю концепцію більш детально:

Інтернет-магазин — це специфічний формат бізнесу, що ґрунтується на можливостях дистанційної роботи з клієнтами через мережу Інтернет. Цей формат забезпечує широку доступність товарів та послуг для споживачів, оскільки вони можуть здійснювати покупки з будь-якої точки світу, за умови наявності доступу до Інтернету.

Інтернет-магазини — це інтерактивні веб-ресурси, які крім реклами товарів чи послуг, також приймають замовлення, надають послуги доставки та реалізують різноманітні способи оплати. Це чудова можливість для підприємств взаємодіяти зі своїми клієнтами на новому рівні та надавати їм зручності, яких вони потребують.

Інтернет-магазин — це система, що складається з безлічі програм, які працюють на веб-сайті і забезпечують можливість покупцям дистанційно обирати товари зі списку та розміщувати свої замовлення. Ця технологія суттєво спрощує та пришвидшує процес покупок, роблячи його більш зручним і доступним.

Інтернет-магазини виступають посередниками між кінцевими споживачами та складами виробників, що означає, що вони з'єднують виробників товарів безпосередньо з покупцями, що оптимізує ланцюг постачання.

Щоб скористатись усіма перевагами, які надає Інтернет-магазин, користувачам необхідно мати лише доступ до Інтернету та мобільний пристрій.

Логіка використання інтернет-магазинів досить проста і зручна. Користувач заходить на веб-сайт і має можливість переглядати каталог товарів, в якому представлені різноманітні категорії та моделі. Коли користувач обирає певний товар або послугу, він переходить на сторінку оформлення замовлення, де може вказати необхідну кількість, бажаний колір або інші параметри. Якщо покупець вже замовляв товари раніше, йому достатньо увійти на сайт під своїм обліковим записом.

Усі важливі дані, такі як адреса доставки, спосіб оплати та історія попередніх замовлень, будуть автоматично підставлені, що значно полегшує процес оформлення нових замовлень. Це надає покупцеві зручність і економить час, що також сприяє підвищенню загального рівня задоволеності клієнтів.

Інтернет-магазин і його переваги та недоліки

У сучасному світі інтернет-магазини стають важливою складовою комерційної діяльності, пропонуючи безліч переваг, що істотно відрізняють їх від традиційних роздрібних точок продажу. Ось декілька ключових аспектів, які визначають їх привабливість для споживачів:

Цілодобове обслуговування: Інтернет-магазини функціонують 24 години на добу, 7 днів на тиждень, що надає клієнтам можливість здійснювати покупки у будь-який час. Користувачі можуть шопінгом займатися і вранці, і вночі, що є великою перевагою для тих, хто має напружений графік роботи або зобов'язання в інших сферах життя. Це дозволяє покупцям зручно організовувати свій час і робити покупки в зручний для них момент.

Відсутність фізичних приміщень: Інтернет-магазини не потребують наявності фізичних торгових точок, що значно знижує витрати на оренду приміщень, комунальні послуги та їх обслуговування. Це створює можливості для зниження цін на товари та послуги, що в свою чергу робить їх більш доступними для споживачів.

Замовлення товарів під замовлення: У багатьох випадках користувачі

можуть замовляти товари, які тимчасово відсутні на складі, та отримувати їх, як тільки вони з'являться у наявності. Це дозволяє покупцям отримати доступ до ширшого асортименту товарів, не переживаючи про те, що потрібний товар може бути недоступним.

Доступність: Інтернет-магазини доступні з будь-якого місця, де є підключення до Інтернету. Це означає, що покупці можуть здійснювати покупки зі зручного місця — з дому, з роботи або навіть під час подорожі. У світі, де мобільний Інтернет стає дедалі популярнішим, ця перевага стає особливо актуальною.

Глобальний продаж: Інтернет-магазини відкривають можливості не лише для локального, а й для міжнародного бізнесу. Вони можуть пропонувати свої товари та послуги споживачам у різних країнах, де є доступ до Інтернету. Це означає, що компанії можуть розширювати свої ринки, підвищуючи впізнаваність бренду на міжнародній арені.

Автономність: Хорошо розроблені веб-сайти вимагають мінімального обслуговування. Це дозволяє власникам бізнесу зосередитися більше на стратегіях продажу та маркетингу, ніж на технічних аспектах. Також це зменшує витрати на підтримку та експлуатацію платформ.

Попри численні переваги, інтернет-магазини стикаються з деякими недоліками, які можуть мати серйозні наслідки для бізнесу:

Проблеми безпеки: Зі зростанням популярності онлайн-покупок, проблеми безпеки стали актуальнішими. Хакерські атаки на веб-сайти можуть призводити до компрометації баз даних, персональних даних клієнтів та самого сайту. Це не лише шкодить бізнесу, але й ставить під загрозу довіру споживачів.

Технічні помилки: Як і будь-яка інша технологія, інтернет-магазини можуть стикатися з критичними помилками, що призводять до "багів" у роботі сайту. Такі проблеми потребують термінового вирішення, і їх усунення лягає на плечі розробників, які мають бути готові оперативно реагувати на будь-які збої.

Непостійна клієнтська база: Залучити постійних клієнтів у інтернет-магазини може бути складніше, ніж у традиційних магазинах. Доступ до веб-

сайту може бути легким, але так само просто споживачі можуть його залишити, знаходячи альтернативи.

2 ОПИС ІНСТРУМЕНТАЛЬНОЇ БАЗИ

2.1 Послідовність етапів розробки веб-сайту інтернет-магазину

Створення веб-ресурсу, зокрема інтернет-магазину, є багатоетапним процесом, який зазвичай охоплює як основні, так і другорядні етапи. Ці етапи можна умовно розділити на кілька ключових процесів:

- Підготовка
- Планування
- Розробка
- Тестування
- Публікація

Підготовка

Перший етап, що фокусується на підготовці, є надзвичайно важливим, оскільки він закладає основу для всього подальшого процесу. Успішна підготовка передбачає врахування ряду критично важливих аспектів, які вплинуть на ефективність вашого веб-ресурсу в майбутньому.

Ось кілька ключових нюансів, які потрібно розглянути на цьому етапі:

Визначення потреб аудиторії: Основним завданням є з'ясування, які саме потреби має ваша цільова аудиторія. Це може включати вивчення товарів або послуг, які користувачі бажають придбати, а також розуміння їхніх проблем і запитів, які ваш ресурс може вирішити. Проводячи опитування та фокус-групи, ви зможете отримати цінну інформацію про споживчі уподобання та тренди.

Аналіз цільової аудиторії: Визначення характеристик цільової аудиторії є важливим для успіху будь-якого бізнесу. Необхідно сегментувати цю аудиторію на різні групи за критеріями, такими як вік, стать, інтереси та купівельна поведінка. Це дозволить вам налаштувати ваш контент та

маркетингові кампанії так, щоб вони були максимально ефективними.

Вивчення конкуренції: Один із найважливіших аспектів підготовки — це аналіз конкурентів. Це дає змогу зрозуміти, що пропонують інші компанії у вашій ніші, з якими сильними і слабкими сторонами вони стикаються. Конкурентний аналіз також допомагає ідентифікувати можливості для диференціації вашого продукту або послуги.

Створення унікального дизайну: Дизайн вашого веб-ресурсу є одним із найважливіших факторів, які впливають на сприйняття користувачами вашого бренду. Створення оригінального та естетично привабливого дизайну допоможе вам виділитися серед конкурентів. При розробці дизайну слід враховувати привабливість кольорової палітри, типографії та графічних елементів.

Визначення ринкового сегмента: Важливо чітко визначити, на який конкретний ринковий сегмент ви плануєте орієнтуватися з вашим веб-сайтом. Це допоможе зрозуміти, які стратегії маркетингу та продажів будуть найбільш ефективними. Також це включає розуміння демографічних і поведінкових характеристик вашої цільової аудиторії.

Заповнення каталогу товарів: На цьому етапі варто підготувати чіткий список товарів, які будуть представлені на сайті, а також створити якісні описи та зображення для кожного товару. Важливо, щоб інформація про товари була актуальною та вичерпною, оскільки це може вплинути на рішення споживачів під час покупки.

Планування

Другий етап у процесі створення веб-ресурсу, що особливо стосується інтернет-магазину, фокусується на ретельному плануванні, яке включає в себе розробку детального технічного завдання (ТЗ). Це важливий документ, призначений для детального опису всіх аспектів, які стосуються майбутнього веб-сайту. Кілька ключових елементів, які повинні бути включені до ТЗ:

Основні напрямки розвитку: У технічному завданні необхідно визначити

стратегічні цілі, які буде переслідувати ваш інтернет-магазин. Це можуть бути такі цілі, як підвищення впізнаваності бренду, збільшення обсягів продажів, розширення асортименту товарів або надання нових послуг для споживачів. Визначення цих напрямків допоможе формувати чітку стратегію досягнення поставлених цілей.

Каталог товарів та їх опис: У ТЗ потрібно чітко описати, які товари чи послуги будуть запропоновані. Для кожної категорії товарів необхідно зазначити конкретні деталі, такі як характеристики, ціни, а також наявність. Це має бути детальний розгляд вашого асортименту, щоб користувачі могли легко орієнтуватися у ваших пропозиціях.

Характеристики цільового користувача: Визначте профіль типового відвідувача вашого сайту. Це може включати в себе демографічні дані, як-от вік, стать, доходи та інтереси. Знання про те, хто саме є вашими потенційними клієнтами, дозволить вам точно налаштувати контент і маркетингові кампанії.

Визначення структури сайту: Важливим аспектом є те, як ваш сайт буде організований в інформаційному плані. Необхідно чітко визначити, які сторінки і секції будуть представлені на сайті, як вони будуть пов'язані одна з одною та які елементи навігації будуть використовуватися. Це дозволить створити зрозумілу і логічну структуру для користувачів.

Функціонал сайту: Визначте, які функції повинні бути реалізовані на веб-сайті, щоб задовольнити потреби користувачів. Це може включати можливість фільтрації товарів, пошук за ключовими словами, реєстрацію користувачів, інтеграцію платіжних систем та інші корисні інструменти, які допоможуть забезпечити ефективну роботу вашого інтернет-магазину.

Технічні вимоги: Необхідно пам'ятати про технічні аспекти, такі як швидкість завантаження сайту, шифрування даних, адаптивна верстка та інші вимоги, що впливають на продуктивність і безпеку ресурсу.

Зібравши всі ці елементи, ви зможете перейти до наступного етапу — створення базового дизайну інтернет-магазину. Процес розробки дизайну

необхідно планувати так, щоб він відповідав структуруванню каталогу товарів і спрощував навігацію по сайту.

Навігація та користувацький досвід

Особливо важливо подумати про те, як відвідувачі будуть пересуватися по сайту. Це означає, що вам необхідно розглянути різноманітні шляхи переміщення користувачів, передбачити їхній досвід взаємодії з різними елементами сайту. Основна мета цього етапу полягає в розробці найбільш зручних і простих варіантів навігації, щоб користувачі могли легко знайти потрібну їм інформацію і товари.

Результати цього конфігураційного процесу стануть основою для подальшого створення вашого веб-ресурсу. Чітка навігація не лише підвищить задоволеність користувачів, а й вплине на їхнє рішення про купівлю.

Розробка дизайну веб-сайту

Наступним етапом є створення дизайну сайту, який є ключовим елементом у процесі розробки інтернет-магазину. Дизайн формується на основі технічного завдання з урахуванням наявного фірмового стилю або власному стилі, якщо такий існує.

Головна мета цього етапу — передати особливу атмосферу, створивши привабливий та оригінальний стиль, що відповідає вподобанням цільової аудиторії. Важливо, щоб дизайн не тільки виглядав естетично, але й був практичним у використанні.

При розробці дизайну слід врахувати такі аспекти:

Адаптивність сайту: Веб-сайт може бути адаптивним або мобільним, що означає, що елементи сторінки, такі як кнопки та зображення, автоматично підлаштовуються під різні розміри екранів. Це дозволяє забезпечити комфортний перегляд та взаємодію з сайтом на різних пристроях.

Оптимізація для мобільних пристроїв: Важливо створити спеціальні версії сайту, які будуть відповідати потребам користувачів мобільних телефонів і

планшетів. Це допоможе зберегти функціональність і зручність використання ресурсу для мобільних користувачів.

Окрім візуальних аспектів, особливу увагу слід приділити зрозумілості сайту. Він має бути інтуїтивно зрозумілим і легким у навігації, щоб користувачі могли швидко знаходити потрібну інформацію та товари. Створення логічної структури і зрозумілої навігації допоможе підвищити задоволеність користувачів і полегшити їхній досвід взаємодії з вашим інтернет-магазином.

Процес створення дизайну веб-сайту включає в себе кілька важливих етапів, кожен з яких спрямований на створення естетично привабливого та функціонального інтерфейсу. Основні етапи виглядають наступним чином:

Вибір кольорової палітри: На цьому етапі важливо визначити основні кольори, які будуть використовуватися для оформлення сайту. Правильна кольорова гамма може суттєво вплинути на сприйняття бренду і навколишнього простору. Вибір кольорів повинен відповідати іміджу компанії та вподобанням цільової аудиторії.

Формування єдиного, цілісного вигляду елементів: Це включає в себе створення гармонійного та узгодженого стилю для всіх елементів інтерфейсу, таких як кнопки, поля вводу, меню тощо. Важливо, щоб усі ці елементи добре поєднувалися один з одним і створювали єдину візуальну мову, що сприяє зручності користування сайтом.

Упорядкування шрифтів та стилів: На цьому етапі потрібно обрати відповідні шрифти, які будуть використовуватися на сайті. Важливо забезпечити читабельність тексту й узгодженість стилю в заголовках, підзаголовках та основному контенті. Кожен елемент тексту повинен мати свої стилістичні характеристики, які підкреслюють зміст та структуру контенту.

Створення графічного наповнення: Цей етап може включати як розробку оригінальних графічних елементів із нуля, так і вибір із вже готових шаблонів або зображень. Якісне графічне наповнення підвищує естетичну привабливість сайту і може суттєво покращити враження від взаємодії з ним.

Розробка анімацій для переходів: Анімації допомагають зробити досвід користувача більш інтерактивним і динамічним. На цьому етапі потрібно створити анімації, які будуть використовуватися при переходах між сторінками або в реакції на дії користувача. Грамотно розроблені анімації здатні покращити загальне сприйняття сайту та зробити його більш привабливим для відвідувачів.

Наступним важливим етапом є безпосередня розробка веб-сайту. Цей процес може бути здійснений різними способами, і існує кілька основних підходів до створення веб-ресурсів.

Перший підхід — це ручна розробка веб-сайту з використанням мов веб-програмування. Цей метод, відомий як ручне створення, вимагає знань таких мов програмування, як HTML (мова гіпертекстової розмітки) і CSS (каскадні таблиці стилів). HTML відповідає за формування структури майбутнього сайту, тоді як CSS використовується для налаштування його візуального оформлення.

Якщо ви плануєте створити динамічний веб-сайт, вам буде потрібно використовувати мову програмування JavaScript. Ця мова дозволяє створювати скрипти для веб-сторінок, що забезпечують інтерактивність клієнтської частини (пристрій, який використовує кінцевий користувач). JavaScript взаємодіє з користувачем, дозволяє керувати браузером, а також забезпечує асинхронний обмін даними з сервером, що дозволяє змінювати як структуру, так і зовнішній вигляд веб-сторінки.

Цей підхід також підходить для проектів, де необхідна інтеграція з базами даних. При ручній розробці веб-сайтів графічний дизайн створюється безпосередньо вручну, і ви можете використовувати графічні редактори для цієї мети.

Другий підхід до розробки веб-сайтів — це автоматизоване створення ресурсів. Це здійснюється за допомогою спеціальних конструкторів веб-сайтів або систем управління контентом (CMS).

CMS — це програмне забезпечення, яке дозволяє управлінню, редагуванню, відображенню та контролюванню вмісту на веб-сайті. Сучасні системи управління контентом пропонують широкий набір функцій та модулів, кожен з

яких відповідає за конкретні елементи сайту. Такі платформи дають можливість швидко створити типовий веб-сайт, використовуючи готові блоки, що спрощує процес розробки та знижує вимоги до технічних знань користувачів.

Для розробки веб-сайтів з використанням системи управління контентом (CMS) не потрібні спеціальні навички програмування, що робить цей підхід доступним навіть для тих, хто не має технічної освіти. CMS пропонують користувачам ряд переваг, завдяки яким процес створення та обслуговування веб-ресурсів стає значно простішим і ефективнішим.

Однією з основних переваг є можливість легко управляти сайтом. Завдяки інтуїтивно зрозумілому інтерфейсу, користувачі можуть без зусиль наповнювати сайт контентом, вносити зміни в наявні ресурси та контролювати їх. Це надзвичайно корисно, оскільки дозволяє підтримувати актуальність інформації на сайті, зокрема новини, оновлення продуктів чи послуг, а також інші важливі дані.

Ще одним позитивним аспектом є доступність для широкого кола користувачів. Немає потреби бути експертом у сфері ІТ або мати серйозні навички програмування, щоб з легкістю запускати та управляти сайтом на платформі CMS. Це дозволяє підприємствам та окремим особам реалізовувати свої ідеї та бізнес-проекти без необхідності залучати додаткових спеціалістів.

Також важливо зазначити, що нові сторінки можуть бути створені за короткий час. Це дає змогу швидко адаптуватися до змін на ринку, реагувати на запити клієнтів або швидко виводити нові продукти на ринок без великих фінансових витрат. Це особливо актуально в умовах високої конкуренції, де швидкість реагування на запити споживачів може стати вирішальним фактором.

Крім того, завдяки CMS забезпечується оптимізація зовнішнього вигляду сайту та підвищення якості контенту. Системи управління контентом зазвичай мають інструменти для SEO-оптимізації, що дозволяє покращити видимість сайту в пошукових системах. Це критично важливо для залучення нових відвідувачів і збільшення трафіку на ресурс.

Функціональність сайту, його ємність і зручність використання в значній мірі залежать від вибору правильної CMS. Вибір правильної системи управління

контентом може забезпечити основу для успішного розвитку веб-ресурсу та зробити його привабливим для потенційних клієнтів. Користуваці отримують надійний ресурс, що бездоганно працює та надає широкий спектр можливостей для створення веб-сайтів будь-якої складності.

Однією з ключових переваг використання CMS є гнучкість налаштувань. Це дозволяє користувачам легко редагувати як саму систему, так і окремі елементи. Використовуючи CMS, ви можете швидко вносити зміни в вміст, додавати нові елементи та адаптувати сайт під потреби користувачів. Це створює можливість для динамічного розвитку вашого ресурсу, що в свою чергу сприятиме збільшенню зацікавленості та задоволення відвідувачів.

Після успішного запуску вашого інтернет-магазину, наступним важливим етапом є процес наповнення сайту контентом. Цей крок є надзвичайно суттєвим, оскільки саме від якості і кількості контенту залежить, наскільки привабливим та інформативним буде ваш веб-ресурс для користувачів. На цьому етапі розробки важливо підготувати всю необхідну текстову та графічну інформацію, яка буде використана для заповнення сайту.

Вам потрібно ретельно зібрати та організувати всі текстові матеріали, які допоможуть клієнтам краще зрозуміти ваші пропозиції. Це можуть бути детальні описи товарів, інформація про їх переваги, вартість, способи доставки та інші важливі моменти, які можуть зацікавити потенційних покупців.

Крім тексту, також важливо забезпечити сайт високоякісними графічними зображеннями, які продемонструють ваші товари у найкращому світлі. Яскраві та чіткі фотографії здатні значно підвищити зацікавленість клієнтів і стимулювати їх до покупки.

Контент грає вирішальну роль у формуванні іміджу вашого бренду і може істотно вплинути на рішення споживачів. Якісний і добре структурований контент не тільки полегшує навігацію по сайту, але й створює у користувачів відчуття довіри до вашого бізнесу. Слід також пам'ятати, що для інтернет-магазинів виділяють три основні типи контенту, що є особливо актуальними в цій

сфері, і їх правильне використання допоможе вам досягти максимальних результатів у залученні і утриманні клієнтів.

Одночасно з процесом заповнення вітрини вашого інтернет-магазину товарами, настає етап тестування, який разом з публікацією формує фінальну частину розробки сайту. Ці процеси є надзвичайно важливими, оскільки вони забезпечують готовність ресурсу до запуску та його подальшого функціонування.

На етапі випробування відбувається всебічне тестування сайту, яке включає перевірку різних аспектів його роботи. Одним із найбільш критичних параметрів є адаптивність дизайну. Це означає, що важливо переконатися в тому, що сайт зберігає свій запланований вигляд і коректно функціонує на різних пристроях, таких як смартфони, планшети та настільні ПК.

Тестування адаптивності охоплює перевірку не лише візуальних елементів, але й забезпечення зручності навігації, легкості доступу до інформації та магазинних функцій на різних екранах. Важливо, щоб сайт був оптимізований для всіх доступних браузерів, щоб забезпечити користувачам однаковий досвід незалежно від того, яким пристроєм вони користуються.

Цей етап тестування є критично важливим, оскільки гарантує, що усі користувачі отримають позитивний досвід використання вашого ресурсу. Надійна адаптивність значно підвищує шанси на успішне залучення та утримання клієнтів, адже зручність та доступність сайту впливають на рішення споживачів при покупках.

Тепер звернімо увагу на хостинг вашого веб-сайту. Як і будь-які інші комп'ютерні програми чи електронні документи, ваш сайт складається з набору файлів, які зберігаються на комп'ютері, відомому як сервер. Сервери зазвичай розташовуються в спеціально обладнаних дата-центрах, де забезпечуються технічне обслуговування, безпека та постійний доступ до Інтернету на високій швидкості.

В залежності від обсягу та навантаження вашого сайту, ви можете орендувати різні ресурси — від невеликого дискового простору до цілого фізичного сервера, або ж розмістити свій сайт в дата-центрі. Організація, що

надає ці послуги, називається хостинг-провайдером або просто хостом, а сама послуга, яку вони пропонують, називається хостингом.

При виборі хостинг-провайдера слід звернути увагу на місце розташування дата-центру. Ви також повинні переконатися, що обрана вами послуга пропонує наступне: необхідне програмне забезпечення та достатній обсяг диску.

2.2 Опис інструментів для розробки

Для розробки інтернет-магазину мною був обраний досить відомий редактор Brackets.

Brackets – редактор з досить простим та зручним дизайном у якому є підтримка більшості мов програмування.

Brackets позиціонується як текстовий редактор, але він все більше нагадує повноцінну IDE. При базовому встановленні цього редактора ви отримуєте:

- Плагін для Live Preview – будь які зміни у коді відображаються на сторінці сайту у реальному часі. (Працює тільки з Google Chrome)
- Підказки під час редагування CSS, JS і HTML-коду
- Підсвічування синтаксису
- Можливість встановлювати додаткові плагіни дає можливість зробити з цього редактора справжню «Машину» для написання будь-якого ПО, не тільки сайтів.

Інтерфейс виконано в стилі мінімалізму, тому він легко сприймається і простий в освоєнні (рис. 2.1).

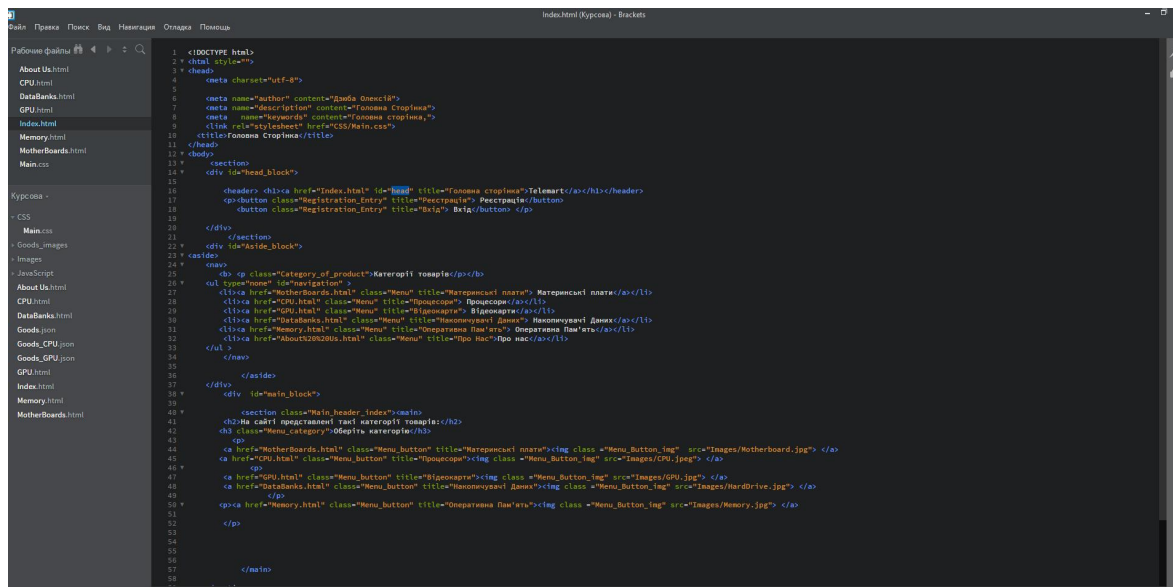


Рисунок 2.1 – Редактор коду Brackets

Редактор коду Brackets займає унікальну позицію на ринку завдяки своїй філософії розробки та набору функцій, орієнтованих на спрощення та прискорення процесу веб-розробки. Дві ключові характеристики Brackets це відкритий вихідний код та розробка спільнотою розробників. Це означає, що його функціональність постійно вдосконалюється, а сам код доступний для вивчення та модифікації. Завдяки цьому, досвідчені розробники можуть повністю адаптувати Brackets під свої потреби, створюючи власні розширення та доповнення. Це відкриває широкі можливості для кастомізації та інтеграції з іншими інструментами.

Крім того, Brackets виділяється своєю швидкістю роботи та зручністю використання. Він відрізняється швидким завантаженням та швидким відгуком на дії користувача, що дозволяє значно економити час під час розробки. Інтерфейс Brackets інтуїтивно зрозумілий та мінімалістичний, що спрощує навігацію та пошук необхідних функцій.

Однак, найбільш значущою особливістю Brackets є функція інтерактивного перегляду (Live Preview). Цей потужний інструмент дозволяє розробникам спостерігати зміни у коді HTML та CSS в режимі реального часу без необхідності постійного оновлення сторінки в браузері. Коли ви редагуєте

CSS-правило, Brackets автоматично підсвічує відповідні елементи на сторінці в браузері, що дозволяє миттєво бачити результат внесених змін. Аналогічно, при редагуванні HTML-коду, Brackets підсвічує відповідні елементи, що значно спрощує процес налагодження та покращує розуміння взаємодії між HTML та CSS.

3 ОПИС РОЗРОБКИ ВЛАСНОГО ПРОДУКТУ

3.1 Структура веб-сайту інтернет-магазину

Вибір оптимальної структури веб-сайту, особливо для інтернет-магазину, є критично важливим рішенням, що впливає на зручність використання, ефективність пошукової оптимізації (SEO) та загальний успіх проекту. Хоча існують різні підходи до організації контенту, такі як лінійна (послідовне розташування сторінок), блокова (групування контенту за тематичними блоками) та інші, для електронної комерції найефективнішими є деревоподібна та тегована структури, часто використовуючись разом для синергетичного ефекту.

1. Деревоподібна структура: Ця структура, що нагадує дерево з розгалуженням, є найпоширенішою для інтернет-магазинів завдяки інтуїтивній ієрархічній організації. Головна сторінка виступає як корінь, від якого відгалужуються основні категорії товарів. Кожна категорія, у свою чергу, може містити підкатегорії, створюючи багаторівневу навігацію. Ця ієрархічна структура дозволяє користувачам легко переходити від загальних категорій до більш специфічних, поступово звужуючи коло пошуку та знаходячи потрібний товар.

Переваги деревоподібної структури:

Зручність навігації: Інтуїтивно зрозуміла структура спрощує пошук товарів, покращуючи користувацький досвід.

Універсальність: Підходить для інтернет-магазинів з різним асортиментом товарів.

SEO-оптимізація: Чітка ієрархічна структура сприяє кращому ранжуванню в результатах пошуку.

Легкість оновлення: Додавання нових категорій та підкатегорій є відносно простим процесом.

Недоліки деревоподібної структури:

Складність навігації при надмірному розгалуженні: Занадто багато рівнів можуть заплутати користувачів, ускладнюючи пошук потрібного товару. Тому важливо ретельно планувати структуру та обмежувати кількість рівнів.

Обмежена гнучкість: Не завжди дозволяє ефективно представляти товари з перетином характеристик.

2. Тегована структура: Ця структура доповнює деревоподібну, надаючи додаткові можливості для пошуку та фільтрації товарів. Вона базується на використанні тегів (ключових слів), які описують характеристики товару: бренд, колір, розмір, матеріал, функціональність тощо. Користувачі можуть використовувати фільтри за тегами, щоб звужити коло пошуку та знайти потрібний товар за конкретними параметрами. Це особливо корисно для великих інтернет-магазинів з широким асортиментом.

Переваги тегованої структури:

Гнучкий пошук: Дозволяє знаходити товари за різними характеристиками, незалежно від їхньої категорії.

Покращена SEO-оптимізація: Використання релевантних тегів покращує видимість сайту в пошукових системах.

Збільшення конверсії: Швидкий та точний пошук збільшує ймовірність покупки.

Недоліки тегованої структури:

Складність у підтримці: Вимагає ретельного планування та постійного моніторингу для забезпечення актуальності та релевантності тегів.

Потенційна надмірність: Неправильне використання тегів може призвести до дублювання контенту та зниження ефективності SEO.

Моя головна сторінка буде містити в собі категорії товарів, а з врахуванням того, що популярні інтернет магазини тієї ж категорії (Електроніка) мають схожу структуру головної сторінки, можна сказати що саме така структура буде мати найбільший вплив на користувачів і буде спонукати їх до

купівлі товару.

- Структура головної сторінки мого сайту:
- Головна сторінка з категоріями товарів: Категорії товарів:
 - Материнські плати;
 - Процесори;
 - Відеокарти;
 - Накопичувачі даних;
 - Оперативна пам'ять;
- Про нас.

Шрифти

Вибір шрифтів для веб-сайту – це не просто естетичне рішення, а важливий аспект юзабіліті та загального враження від ресурсу. Шрифти безпосередньо впливають на читабельність тексту, швидкість сприйняття інформації та загальний стиль сайту. Неправильний вибір може призвести до зниження зручності використання та негативно позначитися на конверсії (в разі комерційного сайту). Тому, до вибору шрифтів необхідно підходити з розумінням їхніх типографічних властивостей та призначення. Основні класифікації шрифтів за технічним стилем включають чотири основні групи:

1. Рукописні шрифти: Ці шрифти імітують рукописний текст, створюючи відчуття натуральності та індивідуальності. Вони часто використовуються для створення неформальної та дружньої атмосфери, але їх надмірне використання може призвести до зниження читабельності, особливо для великих обсягів тексту. Найкраще використовувати рукописні шрифти для коротких фрагментів тексту, наприклад, заголовків, підписів до зображень або коротких цитат (рис.3.1).



Рисунок 3.1 – Рукописний текст

2. Антиква: Це найбільш поширена група шрифтів, що характеризуються наявністю зарубок – коротких штрихів на кінцях основних ліній літер. Зарубки покращують читабельність, оскільки вони допомагають очам плавно переміщатися по тексту. Антиква має багату історію та традиції, а її різноманітні стилі дозволяють використовувати її в широкому спектрі дизайнерських рішень. Антиква підходить як для основного тексту, так і для заголовків, забезпечуючи високий рівень читабельності та естетичної привабливості (рис. 3.2).



Рисунок 3.2 – Атика

3. Акцидентні шрифти: На відміну від антикви та гротесків, акцидентні шрифти призначені не для основного тексту, а для привернення уваги. Це декоративні шрифти з унікальним дизайном та нестандартними формами літер. Їх використовують для заголовків, логотипів, акцентів та інших елементів, які потребують виділення. Надмірне використання акцидентних шрифтів може перевантажити дизайн та знизити читабельність (рис. 3.3).



Рисунок 3.3 – Акцидентний шрифт

4. Гротески (санс-серіфи): Це шрифти без зарубок, що характеризуються простотою та чіткістю. Гротески добре підходять для веб-дизайну, оскільки вони добре відображаються на екранах з різним дозволом. Їх часто використовують для основного тексту, особливо на сайтах з великою кількістю інформації, забезпечуючи високу читабельність та сучасний вигляд. Гротески можуть бути різних стилів – від мінімалістичних до більш виражених (рис. 3.4).



Рисунок 3.4 – Гротеск-шрифти

Крім основних класифікацій, Потрібно враховувати також вагу, стиль (курсив, напівжирний) та розмір шрифту. Правильне поєднання різних шрифтів з урахуванням їхніх властивостей та призначення є ключем до створення гармонійного та зручного для сприйняття веб-дизайну. Не можна забувати про контраст між фоном та текстом для забезпечення оптимальної читабельності. Занадто яскравий або темний текст на невідповідному фоні може значно знизити зручність читання та загальне враження від сайту.

Вибір кольорів

Колір є потужним інструментом у веб-дизайні, здатним викликати емоції, впливати на настрій та формувати враження про бренд. Однак, робота з кольором ускладнюється його суб'єктивним характером. Сприйняття кольору – це не просто фізичний процес, а складне психофізіологічне явище, що залежить від багатьох факторів.

По-перше, сприйняття кольору є індивідуальним. Один і той самий колір може сприйматися різними людьми по-різному, залежно від віку, статі, раси, культурного бекграунду та навіть особистих асоціацій. Наприклад, колір, що асоціюється з радістю та оптимізмом для однієї людини, може викликати тривогу та неспокій у іншої. Крім того, стан здоров'я, навколишнє освітлення та навіть попередньо переглянуті кольори можуть впливати на сприйняття.

По-друге, контекст відіграє ключову роль. Один і той самий колір може сприйматися по-різному залежно від того, як він використовується. Наприклад, яскравий червоний колір може асоціюватися з енергією та пристрасстю в одному контексті, але з агресією та небезпекою в іншому. Сусідні кольори також впливають на сприйняття, створюючи контрастні або гармонійні поєднання.

По-третє, культурні асоціації значно впливають на сприйняття кольору. У різних культурах колір може символізувати різні речі. Наприклад, білий колір у західній культурі асоціюється з чистотою та невинністю, тоді як у деяких азійських культурах – з трауром. Тому, при виборі кольорів для веб-сайту, необхідно враховувати цільову аудиторію та культурний контекст.

Незважаючи на всю цю суб'єктивність, фізичні властивості кольору можна досить точно описати за допомогою моделей кольорового простору (наприклад, RGB, CMYK, HSV). Ці моделі дозволяють кодувати та відтворювати кольори, забезпечуючи їх стабільність. Однак, навіть при використанні цих моделей, важливо тестувати дизайн на різних пристроях та в різних умовах освітлення, щоб переконатися, що кольори відображаються правильно та сприймаються цільовою аудиторією бажаним чином. З урахуванням цих нюансів, ретельний підбір кольорів є ключовим для створення успішного та ефективного веб-дизайну.

У моєму випадку я обрав темні кольори через їх серйозність (рис. 3.5).

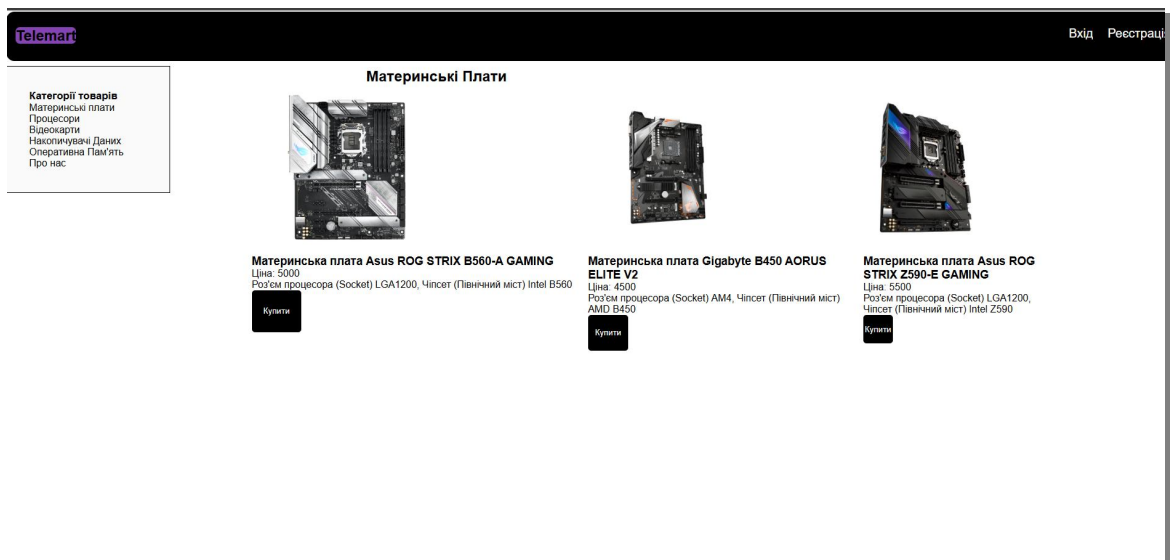


Рисунок 3.5 – Приклад Кольорів на сторінці веб-сайту

3.2 Опис необхідних мов програмування та їх використання

Процес розробки веб-сторінки є ітеративним та послідовним, що вимагає чіткого розуміння та дотримання певного порядку дій. Після вибору відповідного редактора коду та визначення загальної структури сайту, можна приступати безпосередньо до написання коду. Важливо пам'ятати, що верстка – це не одномоментний процес, а поетапна робота, що передбачає послідовне додавання елементів та функціональності.

Короткий опис HTML

HTML (HyperText Markup Language) – це стандартна мова розмітки для створення структури веб-сторінок. Вона служить своєрідним “каркасом” сайту, визначаючи, як різні елементи (текст, зображення, відео тощо) будуть розташовані на сторінці. Був розроблений британськими вченими Тімом Бернерсом-Лі у середині XX сторіччя. Створювався як мова, для обміну науковою та технічними документаціями. В ідеалі, текст з розміткою HTML повинен був без стилістичних та структурних спотворень відтворюватися на обладнанні з різним технічним оснащенням (кольоровий екран сучасного комп'ютера, монохромний екран органайзера, обмежений за розмірами екран смартфона), але сучасний HTML відійшов дуже далеко від своєї початкової задумки та використання.

HTML у розробці веб-сайтів

Як вже було написано вище: HTML (HyperText Markup Language) – це стандартна мова розмітки для створення структури веб-сторінок. Виконує роль каркасу сайту.

Основні елементи HTML-документа:

- `<!DOCTYPE html>`: Ця декларація вказує браузеру, що документ відповідає стандарту HTML5.
- `<html></html>`: Кореневий елемент, який охоплює весь документ.
- `<head></head>`: Містить метадані про документ, які не відображаються безпосередньо на сторінці, але важливі для браузерів та пошукових систем.

Сюди входять:

- `<title></title>`: Заголовок сторінки, який відображається у вкладці браузера.
- `<meta>`: Використовується для встановлення додаткових метаданих, таких як опис сторінки, ключові слова, кодування та ін.
- `<link>`: Підключає зовнішні стилі CSS.
- `<script>`: Підключає зовнішні скрипти JavaScript.
- `<body></body>`: Містить видимий вміст сторінки: текст, зображення, таблиці, форми та інші елементи.

Приклад простої HTML сторінки:

```
HTML<!DOCTYPE html><html><head>
  <title>Сторінка</title>
</head>
<body>
  <h1>Заголовок сторінки</h1>
  <p>Текст.</p>
  
</body>
</html>
```

Детальніше про структуру:

- Теги: Елементи в HTML визначаються тегами, які заключаються в кутові дужки (< i >). Наприклад, <html>, <head>, <body>, <h1>, <p>, .
- Атрибути: Теги можуть мати атрибути, які надають додаткову інформацію про елемент. Наприклад, в тегу атрибут src вказує шлях до зображення, а alt задає альтернативний текст для зображення.
- Вкладеність: Елементи можуть бути вкладені один в одного. Наприклад, параграф <p> може бути всередині розділу <div>.
- Семантика: Важливо використовувати семантичні теги, щоб правильно описати структуру документа і полегшити роботу пошуковим системам. Наприклад, для заголовків використовують теги <h1>, <h2> і т.д., для абзаців — <p>, для списків — і і т.д.

Дуже важливо, щоб структура коду була правильною, адже це допоможе браузерам краще інтерпретувати код та відображати сторінку коректно. Пошукові системи зазвичай використовують структуру HTML для індексації сторінок і визначення їх релевантності до пошукових запитів. Зрозуміло написаний код допоможе іншим розробникам. Наприклад якщо ви працювали у компанії і зробили її сайт, але з будь-якої причини ви звільнилися, і на ваше місце прийшов хтось інший, у зрозуміло написаному коді буде легше розібратися та легко продовжити роботу з вашим кодом.

Але HTML не дає можливостей задати дизайн сторінки, він відповідає в основному за текст. За дизайн відповідає інша мова програмування - CSS.

Короткий опис CSS

CSS (Cascading Style Sheets) – це спеціальна мова стилів, яка використовується для оформлення веб-сторінок. Вона відповідає за те, як виглядатимуть елементи на сторінці: кольори, шрифти, розміри, розташування та багато іншого. Один і той же HTML документ може виглядати зовсім по-іншому, якщо використовувати CSS.

CSS у розробці веб-сайтів

CSS- файл складається з наборів правил, які складаються з:

- Селектора - він визначає до яких елементів HTML будуть застосовані стилі.

- Властивості – вказують на те, що саме ми хочемо змінити у стилі (колір, шрифт, відступи, тощо).
- Значення - визначає конкретне значення для властивості. Приклад коду CSS:

```
p {
  color: black;
  font-size: 16px;
}
```

p - селектор який вибирає всі елементи абзацу.

color: black; - встановлює колір тексту абзацу на чорний.

font-size: 16px; - встановлює розмір шрифту на 16 пікселів.

Існують різні типи селекторів, які дозволяють більш точно задавати стиль частин сторінки:

- Селектори за типом елемента: p, h1, div
- Селектори з ідентифікатором: .class (клас), #id (ідентифікатор)
- Селектори за атрибутами: a[href] (посилання з атрибутом href).
- Селектори нащадків: div p (всі елементи <p> всередині <div>).

```
Приклад коду: .important {
  font-weight: bold;
  color: red; }
#header {
  background-color: black;
  padding: 20px;
}
```

Підключити стилі CSS до коду HTML можна декількома способами:

- Внутрішній стиль: Стилi пишуться безпосередньо у кодi HTML у тезi <style>.

- Стиль елемента: Стиль пишеться безпосередньо в атрибуті style елемента.
- Зовнішній стиль: Підключається зовнішній CSS файл за допомогою тегу <link>. Найпоширеніший та найзручніший спосіб задання стилю.

CSS дозволяє зробити код більш зрозумілим і легшим для підтримки. Також один і то же стиль можна застосовувати для багатьох елементів коду сторінки.

CSS дає можливість зробити сайт, який сам адаптується під різні розміри екранів, від звичайних моніторів ПК, до смартфонів з дуже малими дисплеями та величезними екранами Смарт ТВ.

Створювання анімацій та динамічних ефектів також стає можливим при використанні CSS.

Короткий опис JavaScript

JavaScript - мова програмування, яка дає можливість додати інтерактивності до статичних сторінок HTML. Для роботи використовую безпосередньо браузер користувача, що дозволяє створити будь-які веб-додатки.

JavaScript надає такі переваги:

- Інтерактивність: можливість створювати кнопки, що реагують на натискання, меню, що розкриваються при наведенні курсором, анімації, тощо.
- Динамічні елементи: додає або видаляє елементи на сторінці за необхідності.
- Праця з DOM: отримує доступ до структури документа і дозволяє змінювати вміст сторінки, стилі та атрибути елементів.
- Валідує дані: перевіряє правильність введених даних у формах.
- Взаємодія з сервером: відправляє та отримує дані з сервера(наприклад авторизація користувачів).

JavaScript у розробці веб-сайтів

JavaScript відіграє дуже важливу роль у існуванні веб-сайтів такими, якими ми їх знаємо. Він лежить у основі багатьох веб-додатків по типу Google Docs, Facebook, Gmail. JavaScript ж невід'ємною частиною фронт-енд розробки. Він дає змогу створювати користувацькі інтерфейси. А за допомогою Node.js JavaScript можна створювати серверні додатки.

Основні концепції JavaScript:

- Змінні
- Типи даних
- Оператори
- Умови та цикли
- Функції
- Об'єкти
- Модулі

Змінні - контейнери для даних та їх зберігання. Для оголошення змінних використовуються слова `let`, `const` та `var`.

Типи даних - JavaScript має динамічну типізацію. Тип змінної буде визначатися автоматично. Є декілька типів змінних:

- Числа - цілі або дробові числа.
 - Рядки - Текстові дані в лапках.
 - Булеві - Логічні значення `True` `False`.
 - `Null` - Спеціальне значення, що означає відсутність значення.
 - `Undefined` - Значення що надається змінній без значення, але вже була оголошена.
 - Об'єкти - Збірники значень, що використовуються для моделювання реальних об'єктів.
 - Масиви - Велика кількість впорядкованих елементів.
- Оператори** - спеціальні символи, що виконують дії над даними. Бувають таких типів:
- Арифметичні : додавання(=), віднімання(-), множення(*), ділення(/), залишок від ділення(%) та степіннь(**).
 - Порівняння : Дорівнює(==), не дорівнює (!=), строго дорівнює (===), строго не дорівнює (!==), більше(>), менше(<) та більше, менше або дорівнює(<= ; >=).
 - Логічні
 - Присвоєння : `Let a = 3`.
 - Інкременту та декременту : збільшення (++) або зменшення (--) значення на одиницю.

Умови та цикли :Умови використовуються при виконанні різних дій в

залежності від умов (If...else)Цикли дають змогу повторювати блоки коду декілька разів (for, while, do...while).

Функції

Функції – це блоки коду, що виконують певну задачу. Приймають змінні та повертають значення.

Об'єкти

Об'єкти – основа ООП в JavaScript. Кожен об'єкт має властиві йому ключі та методи(функції).

Прототипи – механізм наслідування в JavaScript. Кожен об'єкт має свій прототип, що містить спільні властивості та методи для всіх об'єктів цього типу.

Модулі

Модулі – це спосіб організувати код у більш зрозумілішій структурі, кращого доступу та можливості використовувати блоки коду декілька разів без необхідності його весь час копіювати.

JavaScript дуже важливий для сучасних веб-сайтів інтернет-магазинів, адже саме завдяки ньому магазини мають змогу бути зручними у використанні, давати максимум можливостей з мінімумом витрат часу та нервів потенціальних покупців.

JavaScript дає змогу зробити сайт інтерактивним, самодостатнім у повній мірі, щоб не треба було постійно покупцю робити щось в ручну(наприклад телефонувати у магазин заради того, щоб замовити товар, адже можна просто додати товар у корзину, і натиснути “Підтвердити замовлення”.

За допомогою CSS веб-сайти стають зручними привабливими для ока, що може стати причиною вибору саме цього магазину, а без HTML на веб-сторінках інтернет магазину не було б тексту і інформації в цілому.

Написання HTML

Це перший етап створення веб-сторінки веб-сайту. Необхідно створити HTML-структуру яка буде виконувати роль скелету, основи веб-сторінки.

На етапі написання HTML-коду важливо дотримуватися принципів чистого коду та семантичної розмітки. Використання HTML5 стандартів є

ключовим для забезпечення сумісності з сучасними браузерами та покращення SEO-оптимізації. Однак, не всі браузери однаково підтримують нові функції HTML5. Для забезпечення коректної роботи веб-сторінки в старіших браузерах, рекомендується використовувати спеціальні інструменти, такі як поліфіл `html5shiv`, який емулює відсутні в старих браузерах елементи HTML5, або бібліотеку `Modernizr`, яка визначає можливості браузера та завантажує необхідні поліфіли лише за потреби. Це дозволяє забезпечити сумісність з максимально широким колом браузерів без зайвого навантаження на сторінку.

Замість застарілих методів верстки за допомогою таблиць та `iframe`, сучасні веб-розробники використовують блокову модель. Це значно спрощує управління структурою сторінки, покращує SEO-оптимізацію та робить код більш зрозумілим та підтримуваним. Таблиці, в свою чергу, використовуються виключно для представлення даних у табличній формі.

HTML-теги служать основою для створення скелета сторінки, визначаючи логічну структуру та ієрархію елементів. Перед початком написання коду рекомендується створити прототип сторінки, наприклад, на папері або в спеціальному інструменті для прототипування. Це допомагає візуалізувати структуру сторінки, визначити необхідні елементи та їх взаємозв'язок, що значно спрощує процес кодування та покращує якість кінцевого результату.

На етапі створення прототипу можна також визначити класи та ідентифікатори для елементів, що полегшує подальше написання CSS-стилів та JavaScript-скриптів.

Для забезпечення чистоти та зрозумілості коду, важливо дотримуватися певних правил найменування класів. Назви класів мають бути осмисленими, зрозумілими та відображати функціональне призначення елемента. Не слід використовувати короткі та неінформативні назви класів. Рекомендується використовувати описові назви, що відображають роль елемента в контексті сторінки.

Одним з найпопулярніших підходів до структурування та найменування

класів є BEM (Block, Element, Modifier).

BEM – це методологія, яка пропонує чіткий та зрозумілий спосіб організації HTML-структури та CSS-стилів. Вона базується на принципі розбиття сторінки на блоки, кожен з яких може містити елементи та модифікатори. Блоки є незалежними компонентами, елементи є внутрішніми частинами блоків, а модифікатори визначають різні стани або варіанти відображення елемента.

Використання BEM покращує читабельність, масштабованість та підтримуваність коду, що є критично важливим для великих та складних проєктів. Наприклад, клас `button` може бути блоком, `button__icon` - його елементом, а `button--primary` - модифікатором, що визначає колір кнопки.

Написання CSS

Завершення етапу HTML-розмітки та ретельного визначення класів є сигналом для переходу до наступного критичного етапу веб-розробки: створення CSS-стилів та верстки макета. Цей етап визначає візуальний вигляд веб-сторінки, її адаптивність та загальне враження на користувача. Перед початком роботи над власними стилями, розробники часто використовують один з двох підходів до управління стилями за замовчуванням, які застосовуються браузерами до HTML-елементів: `reset.css` або `normalize.css`. Вибір між ними впливає на подальший процес стилізації та вимагає розуміння переваг та недоліків кожного підходу.

reset.css: Цей підхід, популяризований Еріком Мейєром, працює за принципом повного скидання всіх стилів за замовчуванням, що застосовуються браузерами до HTML-елементів. В результаті, ви отримуєте чистий аркуш, без будь-яких попередньо визначених стилів. Це дає повний контроль над візуальним оформленням кожного елемента, дозволяючи створювати унікальний дизайн з нуля. Однак, такий підхід вимагає значно більших зусиль, оскільки вам доведеться самостійно визначати стилі для всіх елементів, включаючи базові, такі як `<h1>` - `<h6>`, `<p>`, `<a>`, та інші. Це може бути досить трудомістким та потребує глибокого розуміння типографіки та веб-дизайну.

normalize.css: На відміну від `reset.css`, `normalize.css` не повністю скидає стилі

браузера, а нормалізує їх. Він прагне привести стилі за замовчуванням до більш уніфікованого вигляду між різними браузерами, мінімізуючи відмінності у відображенні елементів. Це спрощує процес розробки, оскільки вам не доведеться перевизначати стилі для базових елементів, а лише доповнювати їх або змінювати за необхідності. `normalize.css` забезпечує більш передбачуване та стабільне відображення елементів на різних платформах та браузерах, що зменшує час налагодження. Однак, це не дає абсолютно повного контролю над стилями, як у випадку з `reset.css`.

"Швидке скидання" (Quick Reset): Існує також спрощений метод "швидкого скидання", який полягає у використанні універсального селектора `* { margin: 0; padding: 0; }`. Цей метод скидає внутрішні та зовнішні відступи для всіх елементів сторінки. Хоча він простий у використанні, він не рекомендується для професійної розробки через потенційні проблеми з продуктивністю та непередбачуваність у відображенні. Цей метод може сповільнити завантаження сторінки та створити проблеми з сумісністю.

На сьогоднішній день `normalize.css` є більш популярним вибором серед розробників, завдяки його балансу між уніфікацією стилів та простотою використання.

Написання JavaScript

Завершальним етапом розробки веб-сайту є додавання JavaScript-скриптів, які надають інтерактивності та динаміки. Однак, вибір між використанням бібліотеки `jQuery` та нативного JavaScript вимагає обережного підходу та ретельного аналізу потреб проекту. Хоча `jQuery` протягом тривалого часу був стандартом де-факто для маніпулювання DOM (Document Object Model) та додавання інтерактивності, сучасний JavaScript досяг значного рівня розвитку, надаючи розробникам потужні вбудовані можливості, які часто роблять `jQuery` зайвим.

Перед тим, як включати `jQuery` до проекту, необхідно критично оцінити необхідність його використання. У багатьох випадках завдання, які раніше вимагали `jQuery`, тепер можна ефективно вирішити за допомогою нативного

JavaScript, що призводить до спрощення коду, покращення продуктивності та підвищення безпеки. Особливо це стосується випадків, коли JavaScript використовується виключно для зміни стилів елементів. Замість того, щоб додавати чи видаляти стилі за допомогою JavaScript, набагато ефективніше та елегантніше використовувати CSS-класи для визначення різних станів елементів (активний, неактивний, прихований, обраний тощо). Це дозволяє відокремити логіку від стилів, покращити читабельність коду та спростити його підтримку.

Наприклад, замість того, щоб використовувати JavaScript для зміни кольору елемента при наведенні курсора, краще визначити два CSS-класи: `element--hover` та `element`. JavaScript лише додає або видаляє клас `element--hover` при відповідних подіях, а CSS визначає візуальні стилі для кожного класу.

Нативний JavaScript залишається незамінним для складніших завдань, таких як:

Обробка подій: Реагування на дії користувача (кліки, наведення курсора, зміна вводу тощо).

Взаємодія з API: Отримання та обробка даних від сервера.

Маніпуляції з DOM: Створення, видалення та зміна елементів сторінки, але лише у випадках, коли це виходить за рамки простих CSS-маніпуляцій.

Анімація: Створення складних анімаційних ефектів.

Реалізація бізнес-логіки: Виконання складних розрахунків та алгоритмів на клієнтській стороні.

Використання нативного JavaScript дозволяє створювати більш ефективний, швидкий та безпечний код, покращує SEO-оптимізацію та спрощує подальшу підтримку та розширення функціональності.

3.3 Опис власного веб-сайту інтернет магазину

Мій веб-сайт складається з різних сторінок:

- Основна сторінка;
- Сторінка на якій представлено Материнські плати;

- Сторінка на якій представлено Процесори;
- Сторінка на якій представлено Відеокарти;
- Сторінка на якій представлено Накопичувачі Даних;
- Сторінка на якій представлено Оперативну Пам'ять;
- Сторінка з короткою інформацією про мене (у випадку розробки сайту компанією інформація на сторінці Про Нас буде стосуватися саме компанії, та її контактних даних).

Головіна сторінка сайту

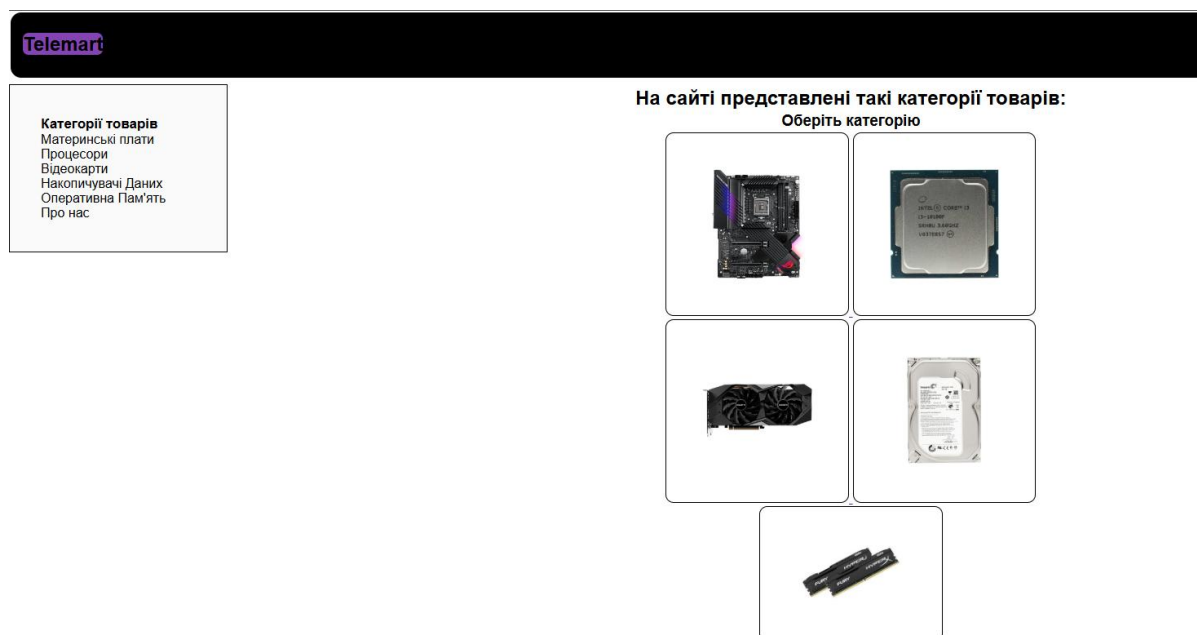


Рисунок 3.1 – Головна сторінка веб-сайту

Головна сторінка складається з блоків з картинками товарів, за які ці блоки відповідають, що дає змогу зрозуміти які товари знаходяться у якому блоці. Самі блок представляють з себе кнопки, що посиляють на сторінку з відповідними товарами (рис. 3.1).

У верхній частині сторінки є кнопка-текст, що відповідає за посилання користувача на головну сторінку веб-сайту, тобто на цю саму сторінку (рис. 3.2).

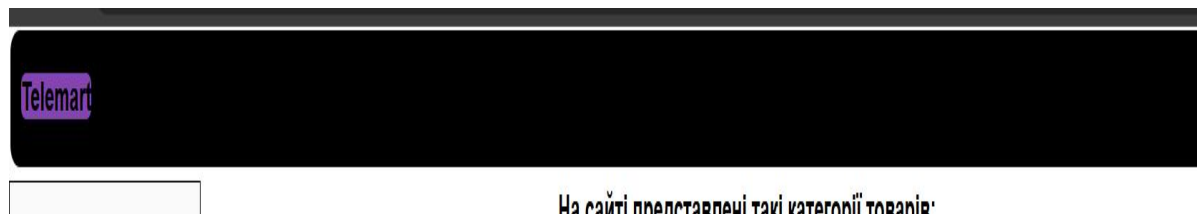


Рисунок 3.2 – Верхня частина сторінки веб-сайту

У лівій сторінці веб-сайту знаходиться блок з категоріями товарів. Цей блок відповідає за швидке орієнтування та зручний доступ до будь-яких товарів представлених на веб-сайті (рис. 3.3).

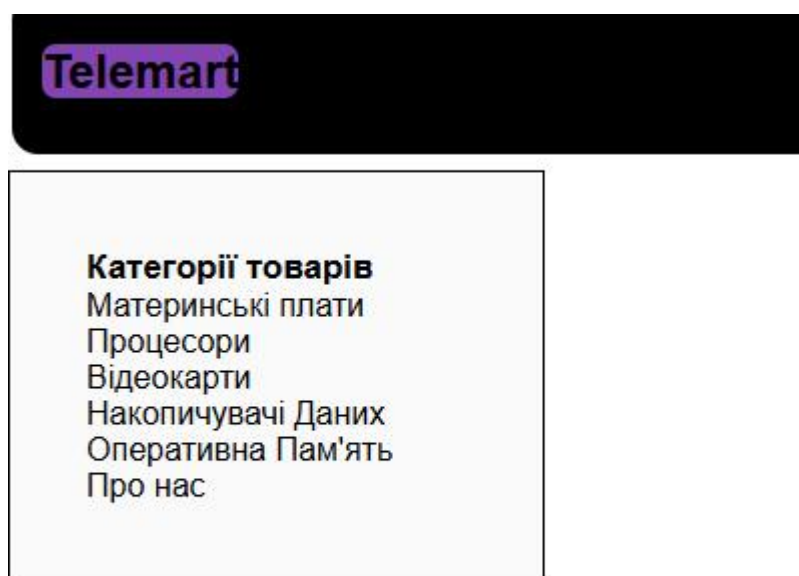


Рисунок 3.3 – Бокова частина сторінки веб-сайту

Сторінка з товаром (відеокарти)

Сторінка з товарами представляє з себе сторінку з товарами. На цій сторінці також є блок навігації збоку, для доступу до інших сторінок з іншими товарами. (рис. 3.4).



Рисунок 3.4 – Сторінка веб-сайту з товарами

На цій сторінці представлені блоки з товарами відповідної до сторінки категорії. У цьому випадку це відеокарти.

В кожному блоці є картинка відповідного товару, його коротка характеристика, ціна, назва. Під кожним блоком є кнопка “Купити”. Кожна сторінка з товарами виглядає так само, але має інші представлені товари.

Сторінка з товаром (процесори)

На цій сторінці з товарами представлені блоки з товарами, у даному випадку з процесорами. В кожному блоці з товаром є картинка товару в залежності від того, який товар у блоці, коротка інформація про товар, кнопка Купити та ціна за одиницю товару. Збоку сторінки розташовано блок з кнопками навігації по сайту, для швидкого доступу до інших товарів магазину (рис. 3.5).

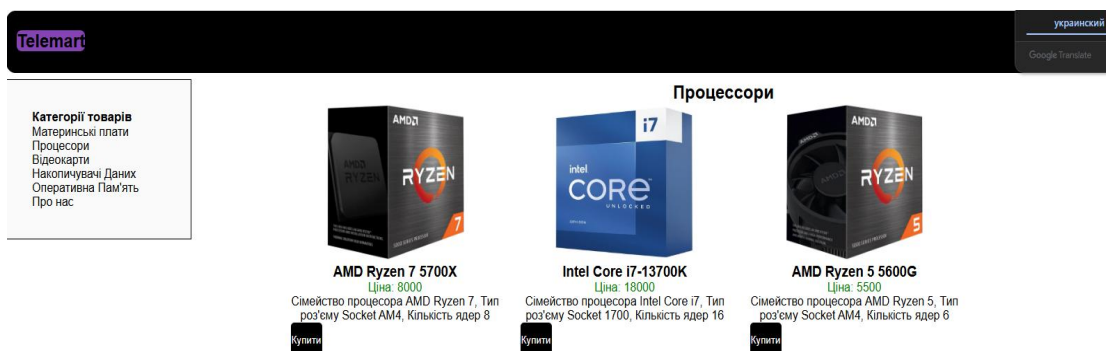


Рисунок 3.5 – Сторінка товару з процесорами

Сторінка з товарами (Материнські плати)

На цій сторінці з товарами представлені блоки з товарами, у даному випадку з материнськими платами. В кожному блоці з товаром є картинка товару в залежності від того, який товар у блоці, коротка інформація про товар, кнопка Купити та ціна за одиницю товару. Збоку сторінки розташовано блок з кнопками навігації по сайту, для швидкого доступу до інших товарів магазину (рис. 3.6)

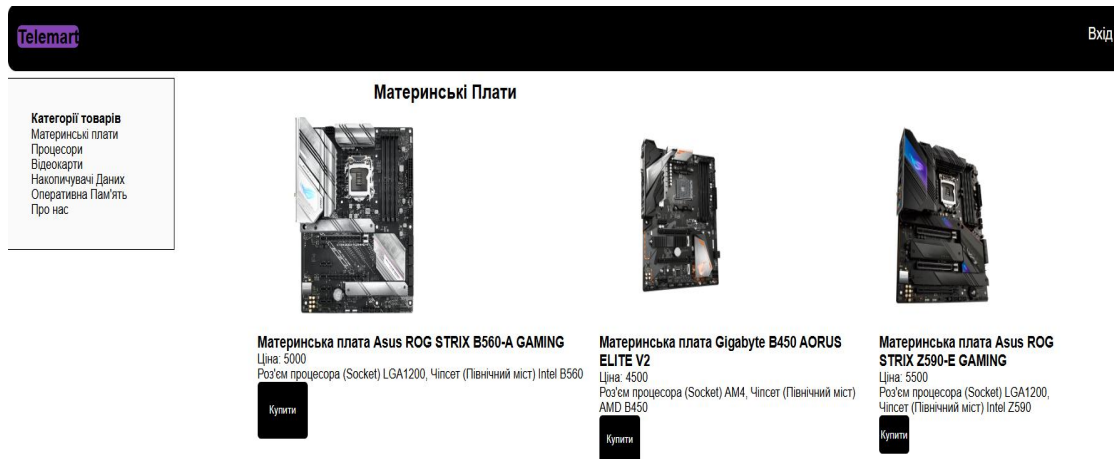


Рисунок 3.6 – Материнські плати

Сторінка веб-сайту про нас

На цій сторінці представлено коротку інформацію про людину, що розробила цей веб-сайт і її контактні дані.

У центрі сторінки розташовано саму інформацію про сайт та контактні данні людини що його розробила.

Ця сторінка має таку ж саму бокову панель з категоріями товарів з посиланнями, та верхню кнопку, що повертає на головну сторінку.

ВИСНОВКИ

Дипломний проект був присвячений розробці функціонального інтернет-магазину під назвою Telemart, використовуючи метод ручної розробки та сучасні веб-технології. Було зроблено огляд існуючих веб-сайтів інтернет-магазинів, таких як Rozetka та OLX, виявлено їх функціонал, зв'язано їх переваги та недоліки. Це дозволило визначити ключові вимоги до розроблюваного інтернет-магазину та обрати оптимальні рішення для його архітектури та функціональності.

Наступним етапом було вивчення та порівняння різних середовищ розробки. Після ретельного аналізу, було обрано Brackets як основне середовище розробки, враховуючи його легкість використання, потужні функції та гнучкість. Brackets, незважаючи на позиціонування як текстовий редактор, надає можливості, близькі до повноцінної інтегрованої середовища розробки (IDE), що полегшує процес розробки та налагодження коду.

Власне розробка інтернет-магазину Telemart здійснювалася за допомогою трьох основних мов веб-програмування: HTML, CSS та JavaScript. HTML був використаний для створення структури сторінок та розміщення контенту, CSS – для визначення візуального стилю та оформлення, а JavaScript – для додавання інтерактивності та динамічних елементів. В результаті було створено функціональний інтернет-магазин з добре продуманим дизайном, що включає в себе:

Функціональний каталог товарів: Реалізовано зручну систему навігації та пошуку товарів за категоріями.

Детальні сторінки товарів: Кожен товар представлений з детальним описом, якісними зображеннями та ціною.

На даному етапі проект Telemart являє собою функціонуючий інтернет-магазин. Для подальшого розвитку та покращення функціональності,

планується в напрямку розширення можливостей за рахунок додавання нових функцій, таких як: система оплати, корзина, система облікових записів користувачів, інтеграція з платіжними системами, система відстеження замовлень та інше.

Підсумовуючи, дипломний проект успішно продемонстрував процес розробки інтернет-магазину з використанням сучасних веб-технологій та ручного кодування. Результат – веб-сайт, який може бути використаний як основа для подальшого розвитку та масштабування. Розвиток електронної комерції продовжується, і інтернет-магазини надаватимуть все більше можливостей для зручної покупки та пошуку інформації, а також покращуватимуть свої показники ефективності.

ПЕРЕЛІК ПОСИЛАНЬ

1. Інтернет-магазин як різновид веб-сайтів. Доступ: https://mir-fin.ru/internet_magasin.html. (Дата звернення 15.11.2024.)
2. Суть електронної комерції. Доступ: <https://www.seonews.ru/glossary/elektronnaya-kommertsiya>. (Дата звернення 15.11.2024.)
3. Юрасов А.В. Електронна комерція: навчальний посібник / Юрасов А.В. - М.: Справа, 2003. - 482 с
4. What is CSS? Доступ: <https://www.computerhope.com/jargon/c/css.htm>. (Дата звернення 16.11.2024.)
5. CSS Tutorial. Доступ: https://www.tutorialspoint.com/css/what_is_css.htm. (Дата звернення 16.11.2024.)
6. HTML Wikipedia. Доступ: <https://uk.wikipedia.org/wiki/HTML> (Дата звернення 16.11.2024.)
7. HTML для починаючих. Доступ:
8. <https://msiter.ru/tutorials/html-nachalnogo-urovnya> (Дата звернення 16.11.2024.)
9. HTML5 для починаючих. Доступ: <https://itproger.com/ua/course/html> (Дата звернення 16.11.2024.)
10. JavaScript Wikipedia. Доступ: <https://uk.wikipedia.org/wiki/JavaScript> (Дата звернення 20.11.2024.)
11. JavaScript Features To Forget. Доступ: <https://davidflanagan.com/2020/05/12/javascript-to-forget.html> (Дата звернення 20.11.2024.)
12. Changes in the Seventh Edition of JavaScript: The Definitive Guide. Доступ: <https://davidflanagan.com/2020/05/03/changes-in-the-seventh-edition.html> (Дата звернення 20.11.2024.)
13. Douglas Crockford: JavaScript Doesn't Suck. Доступ: <https://www.sitepoint.com/interview-doug-crockford/> (Дата звернення 21.11.2024.)

14. A Survey of the JavaScript Programming Language. Доступ: <http://crockford.com/javascript/survey.html> (Дата звернення 21.11.2024.)
15. The Little JavaScripter. Доступ: <https://www.crockford.com/little.html> (Дата звернення 21.11.2024.)
16. The lazy-loading property pattern in JavaScript. Доступ: <https://humanwhocodes.com/blog/2021/04/lazy-loading-property-pattern-javascript/> (Дата звернення 21.11.2024.)
17. Brackets Wikipedia. Доступ: https://uk.wikipedia.org/wiki/Adobe_Brackets (Дата звернення 22.11.2024.)
18. Муравка И. HTML, CSS и JavaScript: ключевые аспекты веб-разработки. [Електронний ресурс]. Доступ: <https://tto-studio.com.ua/ru/html-css-ta-javascript-klyuchovi-aspekty-veb-rozrobky>
19. Росс В.С. Створення сайтів: HTML, CSS, PHP, MySQL: підручник, ч. 1 – М.: МГДД(Ю)Т, 2020. – 107 с.
20. Флэнаган Д. JavaScript: Посібник. СПб: Символ Плюс, 2008. – 992 с.
21. CSS-TRICKS [Електронний ресурс]. Доступ: <https://css-tricks.com>
22. Підручник learn Javascript [Електронний ресурс] .Доступ: <https://learn.javascript.ru>
23. W3Schools [електронний ресурс]. Доступ: <https://www.w3schools.com>
24. Зелковиц М., Шоу А., Гэннон Дж. Принципы разработки программного обеспечения. – М.: Мир, 2002. – 368 с.
25. Ван Тассел Д. Стил, разработка, эффективность, отладка и испытание программ. – М.: Мир, 2005. – 332 с.
26. Дейкстра Э. Дисциплина программирования. – М.: Мир, 2003. – 275 с.

ДОДАТОК А. ФРАГМЕНТИ ПРОГРАМНОГО КОДУ

Index.html

```
<html style="">

<head>

    <meta charset="utf-8">

    <meta name="author" content="Дзюба Олексій">

    <meta name="description" content="Головна Сторінка">

    <meta name="keywords" content="Головна сторінка,">

    <link rel="stylesheet" href="CSS/Main.css">

    <title>Головна Сторінка</title>

</head>

<body>

    <section>

        <div id="head_block">

            <header>    <h1><a href="Index.html" id="head" title="Головна
сторінка">Telemart</a></h1></header>

                <p><button class="Registration_Entry" title="Реєстрація">
Реєстрація</button>

                    <button class="Registration_Entry" title="Вхід"> Вхід</button> </p>

            </div>

        </section>

        <div id="Aside_block">

<aside>

    <nav>

        <b> <p class="Category_of_product">Категорії товарів</p></b>

        <ul type="none" id="navigation" >

            <li><a href="MotherBoards.html" class="Menu" title="Материнські
```

плати"> Материнські плати

Процесори

Відеокарти

<a href="DataBanks.html" class="Menu" title="Накопичувачі
Даних"> Накопичувачі Даних

Оперативна Пам'ять

<a href="About%20%20Us.html" class="Menu" title="Про
Нас">Про нас

</nav>

</aside>

</div>

<div id="main_block">

<section class="Main_header_index"><main>

<h2>На сайті представлені такі категорії товарів:</h2>

<h3 class="Menu_category">Оберіть категорію</h3>

<p>

<a href="MotherBoards.html" class="Menu_button" title="Материнські
плати">

<p>

<a href="DataBanks.html" class="Menu_button" title="Накопичувачі


```

Даних"> </a>
    </p>
    <p><a href="Memory.html" class="Menu_button" title="Оперативна
Пам'ять"> </a>
    </p>
    </main>
</section>
</div>
<div id="footer_block">
<footer>
    Legs
    </footer>
</div>
<script src="JavaScript/main.js"></script>
</body>
</html>

```

MotherBoards.html

```

<html style="">
<head>
    <meta charset="utf-8">
    <meta name="author" content="Дзюба Олексій">
    <meta name="description" content="Головна Сторінка">
    <meta name="keywords" content="Головна сторінка,">
    <link rel="stylesheet" href="CSS/Main.css">
    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
    <title>Головна Сторінка</title>
</head>

```

```
<body>

    <section>

        <div id="head_block">

            <header>    <h1><a    href="Index.html"    id="head"    title="Головна
сторінка">Telemart</a></h1></header>

                <p><button        class="Registration_Entry"        title="Реєстрація">
Реєстрація</button>

                    <button class="Registration_Entry" title="Вхід"> Вхід</button></p>
        </div>

    </div>

    </section>

    <div id="Aside_block">
<aside>

    <nav>

        <b> <p class="Category_of_product">Категорії товарів</p></b>

        <ul type="none" id="navigation" >

            <li><a    href="MotherBoards.html"    class="Menu"    title="Материнські
плати"> Материнські плати</a></li>

                <li><a        href="CPU.html"        class="Menu"        title="Процесори">
Процесори</a></li>

                <li><a        href="GPU.html"        class="Menu"        title="Відеокарти">
Відеокарти</a></li>

                <li><a    href="DataBanks.html"    class="Menu"    title="Накопичувачі
Даних"> Накопичувачі Даних</a></li>

                <li><a href="Memory.html" class="Menu" title="Оперативна Пам'ять">
Оперативна Пам'ять</a></li>

                <li><a    href="About%20%20Us.html"    class="Menu"    title="Про
Нас">Про нас</a></li>
```

```
</ul>

</nav>

</aside>

</div>

<div id="main_block">

  <section class="Main_header_index">

    <h2 class="MotherBoard_title">Материнські Плати</h2></section>

    <div class="MotherBoards_Goods">

      <div                                class="MotherBoards_Goods-container"
id="MotherBoards_Goods-container"></div>

    </div>

  </div>

  <div id="footer_block">

    <footer>

      Тут поки-що нічого(

    </footer>

  </div>

  <script
src="https://ajax.googleapis.com/ajax/libs/jquery/3.7.0/jquery.min.js"></script>

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

</body>

</html>
```

Main.css

```
* {

  margin: 0;

  padding: 0;

  font-family: 'Open Sans', sans-serif;
```

```
}  
  
body{  
    height: 1000px;  
}  
  
#head{color: black;  
  
    font-family: 'Open Sans', sans-serif;  
  
    background: #8444b3;  
  
    border-radius: 7px;  
  
    text-decoration: none;  
  
margin: 25px 15px;  
  
position: relative;  
  
top: 25px;  
  
}  
  
#head:hover{  
  
    color: pink;  
  
}  
  
.Menu {  
  
    border-radius: 0px;  
  
    color: black;  
  
    text-decoration: none;  
  
}  
  
.Menu:hover {  
  
    color: white;  
  
    text-decoration: none;  
  
    cursor: pointer;  
  
    background: black;  
  
    transform: rotate(180deg)
```

```
border-radius: 1px;
border-color: #333;
}
.Menu:active{
    text-decoration: underline;
}
.Registration_Entry{float: right;
    color: white;
    font-size: 20px;
    text-decoration: none;
    position: relative;
    top: -8px;
    display: block;
    margin: 2px 10px;
    background: none;
    border: 0px;
    font-family: 'Open Sans', sans-serif;
    display: block;
    padding: 2px ;
}
.Registration_Entry:hover{
    text-decoration: underline;
}
#head_block{ float: left;
    color: white;
    height: 7%;
    width: 100%;
```

```
border: 2px solid white;
background-color: Black;
text-decoration: none;
margin: 0px;
border-radius: 15px;
padding-bottom: 15px;
display: block;
position: fixed;
z-index: 100;
}
```

```
#footer_block{float: left;
margin-bottom: 0;
margin-top: 50px;
color: white;
height: 10%;
width: 100%;
border: 2px solid purple;
background-color: black;
text-decoration: none;
padding: 20px 20px;
position: relative;
top:100%;
}
```

```
#Aside_block{
float: left;
width: 10%;
padding: 2% 2%;
```

```
background: #fafafa;
border: 1px solid black;
position: relative;
top: 9%
}
#navigation {  }
#main_block{ float: none;
height: 80%;
width: 70%;
max-width: 1600px;
background: none;
border: 0px solid silver;
padding: 20px;
position: absolute;
top: 10%;
right: 0%;
left: 20%;
bottom: 0;
z-index: 50;
}
.Main_header_index{
float: none;
width: 65%;
height: 10%;
max-width: 1600px;
background: none;
border: 0px solid silver;
```

```
padding: ;  
position: absolute;  
top: 0;  
right: 0%;  
left: 15%;  
bottom: 0;  
padding: 0%;  
text-align: center;  
}  
}
```

```
.Menu_button{float: none;  
padding: 2px;  
border: 0px;  
margin: 0;  
border-radius: 3px;  
width: 5%;  
height: 5%;  
position: absolute;  
right: 0%;  
top 10%;  
left: 0%;  
bottom: 0;  
padding: 10%;  
}
```

```
.Menu_Button_img{float: none;  
background: none;  
padding: 1px;
```



```
border: 1px solid black;

border-radius:10px;

width: 15%;

height: 15%;

position: bloc

right: 0%;

top 10%;

left: 0%;

bottom: 0;

padding:5%;

}

.Menu_button:hover{color:none;

background-color: none;

cursor: pointer;

}

.Category_of_product{

color: black;

font-size: 17px;

}

.Menu_category{background: white;

float: none;

padding: 3px;

}

.MotherBoard_title{

float: left;

}

.MotherBoards_Goods-container {
```

```
    display: flex;
}

.MotherBoards_Goods {
    display: inline-block;
    margin-right: 10px;
}

.MotherBoards_Goods-image {float: none;
display: flex;
    flex-wrap: wrap;
    justify-content: center;
    gap: 5px;
    width: 50%;
    object-fit: contain;
    height: 50%;
    margin: 10px;
    padding: 10px;
}

.BuyButton {float: none;
display: flex;
    flex-wrap: nowrap;
    justify-content: center;
    width: 15%;
    padding: 5%;
    background-color: black;
    color: white;
    border: none;
    border-radius: 5px;
```

```
    cursor: pointer;
}
}

.Goods_CPU-container {
    display: flex;
    flex-wrap: wrap;
}

.Goods_CPU {
    display: inline-block;
    width: 300px;
    margin-right: 10px;
    padding: 10px;
    border: 0px solid #ccc;
    text-align: center;
}

.Goods_CPU-image {
    width: 200px;
    height: 200px;
}

.Goods_CPU-name {
    font-weight: bold;
}

.Goods_CPU-cost {
    color: green;
}

.GPU_Goods-container {
    display: flex;
```

```
    flex-wrap: wrap;
}

.GPU_Goods {
    display: inline-block;
    width: 300px;
    margin-right: 10px;
    padding: 10px;
    border: 0px solid #ccc;
    text-align: center;
}

.GPU_Goods-image {
    width: 100%;
    height: 200px;
}

.GPU_Goods-name {
    font-size: 14px;
    margin: 10px 0;
}

.GPU_Goods-cost {
    font-weight: bold;
}
}
```

ДОДАТОК Б. ДЕМОНСТРАЦІЙНИЙ МАТЕРІАЛ



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ ІНФОРМАЦІЙНО-
КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
Кафедра Комп'ютерних наук

ДИПЛОМНА РОБОТА

на ступінь вищої освіти магістр
із спеціальності 122 Комп'ютерні науки

Сайт Інтернет магазину Телемарт на основі JavaScript

Виконав: студент 6 курсу, групи КНДМ-62

Дзюба Олексій Валерійович

Керівник: професор кафедри Комп'ютерних наук
д.ф.-м.н., професор Шикун О.М.

Київ - 2025

1

E-Commerce

- Електронна комерція, більш відома як e-commerce, є однією з найбільш динамічних сфер сучасного бізнесу, яка змінила уявлення про купівлю та продаж товарів і послуг. Цей термін охоплює всі форми бізнес-операцій, які здійснюються через Інтернет, зокрема обмін фінансовими ресурсами та даними для здійснення транзакцій. Первісно електронна комерція асоціювалася саме з продажами фізичних товарів онлайн, але наразі спектр її дії значно ширший, включаючи найрізноманітніші форми взаємодії в бізнес-середовищі.



Види Електронної комерції

- Бізнес для бізнесу (B2B): коли одна компанія продає товари чи послуги іншій компанії, забезпечуючи великий обсяг та спеціалізовані рішення.
- Споживач для споживача (C2C): такі платформи дозволяють споживачам продавати свої товари чи послуги іншим споживачам, що сприяє взаємодії та обміну між користувачами.
- Бізнес для споживача (B2C): у цій моделі підприємства продають товари чи надають послуги безпосередньо кінцевим споживачам, створюючи базу для широкомасштабного продажу онлайн.
- Споживач для бізнесу (C2B): цей тип вказує на ситуації, коли споживачі пропонують свої власні продукти чи послуги бізнесу, зворотно впливаючи на ринок.



Методи розробки веб-сайту

Автоматизована розробка. За допомогою спеціального конструктора веб-сайтів або за допомогою системи управління контентом CMS.



Ручна розробка. З використанням однієї або декількох мов програмування. Необхідні спеціальні навички та знання HTML, CSS та JavaScript .

Опис інструментів для розробки



- Для розробки інтернет-магазину мною був обраний досить відомий редактор Brackets.
- Brackets – редактор з досить простим та зручним дизайном у якому є підтримка більшості мов програмування.
- Brackets позиціонується як текстовий редактор, але він все більше нагадує повноцінну IDE. При базовому встановленні цього редактора ви отримаєте:
- Плагін для Live Preview – будь які зміни у коді відображаються на сторінці сайту у реальному часі. (Працює тільки з Google Chrome)
- Підказки під час редагування CSS, JS і HTML-коду
- Підсвічування синтаксису
- Можливість встановлювати додаткові плагіни дає можливість зробити з цього редактора справжню «Машину» для написання будь-якого ПО, не тільки сайтів.

Оформлення

Дизайн інтернет-магазину важлива частина розробки сайту. Він розробляється на основі технічного завдання з урахуванням існуючого фірмового або власного стилю (якщо є).

Найголовніше завдання на етапі дизайну - передати певний настрій, творити гарний та оригінальний стиль з врахуванням особливостей вподобань цільової аудиторії.



Мова програмування HTML

HTML (HyperText Markup Language) — стандартизована мова розмітки документів для перегляду вебсторінок у браузері. Браузери отримують HTML документ від сервера за протоколами HTTP/HTTPS або відкривають з локального диска, далі інтерпретують код в інтерфейс, який відобразатиметься на екрані монітора.

- ❑ Потрібен для початкової стадії написання сайту.



Мова програмування CSS

- **CSS (Cascading Style Sheets)** — це мова стилю сторінок, що використовується для опису їхнього зовнішнього вигляду. Самі ж сторінки написані мовами розмітки даних (HTML).
- Потрібен для стилізації сторінок веб-сайту.



Мова програмування JavaScript

- **JavaScript (JS)** — динамічна, об'єктно-орієнтована прототипна мова програмування.
- Використовується для створення сценаріїв вебсторінок, що надає можливість на боці клієнта (пристрої кінцевого користувача) взаємодіяти з користувачем, керувати браузером, асинхронно обмінюватися даними з сервером, змінювати структуру та зовнішній вигляд вебсторінки.



Опис веб-сайту

- Веб-сайт був зроблений за допомогою HTML, CSS та JavaScript.
- В наявності є головна сторінка, сторінки з товарам, тощо

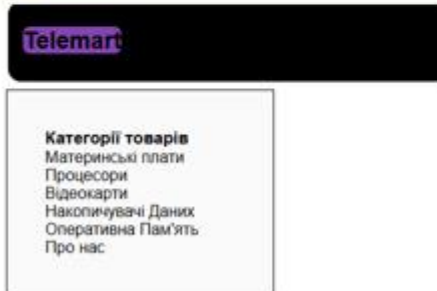


- Дизайн веб-сайту простий і достатньо зрозумілий для вільного орієнтування



Опис сторінок

Сторінки веб-сайту мають кнопки навігації зверху та злівого боку сторінки. Кнопка зверху сторінки представляє з себе лого магазину і перенаправляє користувача на головну сторінку веб-сайту. Кнопки навігації переносять тебе на сторінки з різними категоріями товарів. На головній сторінці розташовані блоки-кнопки з картинками товарів, ці кнопки пересилають користувача на сторінку з категорією товарів, що зображена на картинці.



На сайті представлені такі категорії товарів:
Оберіть категорію



Опис сторінок з товарами

Кожна сторінка з товарами має блоки з товарами.

Блоки з товарами містять у собі короткий інформаційний блок про товар, його опис, назву, картинку, ціну. Під кожним товаром є кнопка "Купити".

На кожній сторінці з боку від товарів розташовані кнопки з навігацією



Висновок

- Дипломний проєкт був присвячений розробці функціонального інтернет-магазину під назвою Telemart, використовуючи метод ручної розробки та сучасні веб-технології. Було зроблено огляд існуючих веб-сайтів інтернет-магазинів, таких як Rozetka та OLX, виявлено їх функціонал, з'ясовано їх переваги та недоліки. Це дозволило визначити ключові вимоги до розроблюваного інтернет-магазину та обрати оптимальні рішення для його архітектури та функціональності.
- Наступним етапом було вивчення та порівняння різних середовищ розробки. Після ретельного аналізу, було обрано Brackets як основне середовище розробки, враховуючи його легкість використання, потужні функції та гнучкість. Brackets, незважаючи на позиціювання як текстовий редактор, надає можливості, близькі до повноцінної інтегрованої середовища розробки (IDE), що полегшує процес розробки та налагодження коду.
- Власне розробка інтернет-магазину Telemart здійснювалася за допомогою трьох основних мов веб-програмування: HTML, CSS та JavaScript. HTML був використаний для створення структури сторінок та розміщення контенту, CSS – для визначення візуального стилю та оформлення, а JavaScript – для додавання інтерактивності та динамічних елементів. В результаті було створено функціональний інтернет-магазин з добре продуманим дизайном, що включає в себе:
- Функціональний каталог товарів: Реалізовано систему навігації та пошуку товарів за категоріями.
- Детальні сторінки товарів: Кожен товар представлений з детальним описом, якісними зображеннями та ціною.
- На даному етапі проєкт Telemart являє собою функціонуючий інтернет-магазин. Для подальшого розвитку та покращення функціональності, планується в напрямку розширення можливостей за рахунок додавання нових функцій, таких як: система оплати, система облікових записів користувачів, інтеграція з платіжними системами, система відстеження замовлень та інше.
- Підсумовуючи, дипломний проєкт успішно продемонстрував процес розробки інтернет-магазину з використанням сучасних веб-технологій та ручного кодування. Результат – веб-сайт, який може бути використаний як основа для подальшого розвитку та масштабування. Розвиток електронної комерції продовжується, і інтернет-магазини надаватимуть все більше можливостей для зручної покупки та пошуку інформації, а також покращуватимуть свої показники ефективності.