



Звіт про оригінальність

● Оцінка схожості

% 2

● Ризик плагіату

НИЗЬКИЙ

👤 Ігор Кагало 🕒 2025-06-14 10:25

Посилання на звіт: 10biG / Посилання користувача: qAHy



Ось вона – Ваша звіт про оригінальність!

Ми раді повідомити, що перевірка вашого документа завершена, і результати вже готові! Наші алгоритми старанно працювали, щоб знайти збіги в наших базах даних.

На наступних сторінках ви знайдете результати перевірки:

Бали

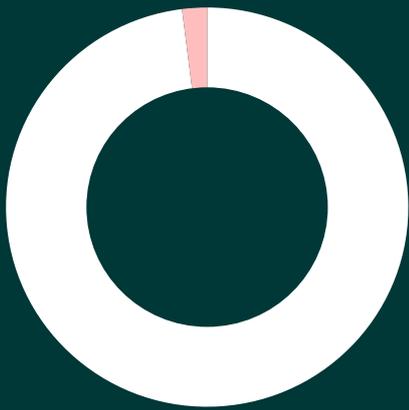
Збіги

Посилання

Ваш документ було перевірено за такими джерелами:

- База даних інтернет-джерел
- База даних наукових статей
- Глибока перевірка (наш вдосконалений алгоритм)

Бали



● Збіги тексту	2%
● Перефразування	0%
● Цитований текст	0%
● Неправильне цитування	0%
● Збігів не знайдено	98%

Ризик плагіату

НИЗЬКИЙ

Ризик плагіату вказує, як збіги тексту розподілені по документу. Вищий ризик виникає, коли збіги з'являються близько один до одного, наприклад, у тому самому абзаці або розділі.

Оцінка схожості

Оцінка схожості показує, скільки слів або символів у вашому документі збігаються з текстами інших документів, включаючи перефразовані тексти або неправильні цитати.

% **2**

Збіги

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ТЕХНІЧНИХ РІШЕНЬ

1.1 Аналіз ринку тематичних товарів в сфері корейської попкультури

У сучасному цифровому середовищі електронна комерція стає ключовим каналом реалізації продукції, орієнтованої на молодіжну аудиторію. Одним із помітних сегментів є торгівля товарами, пов'язаними з корейською попмузикою (K-pop), що активно розвивається внаслідок глобальної популярності жанру. Цей ринок вирізняється високою динамікою, значною емоційною залученістю споживачів і потребою в персоналізованому підході до фанатської спільноти.

Огляд ключових конкурентів показує, що вже існує кілька онлайн-магазинів, які частково задовольняють попит на K-pop мерч. Наприклад, сайт kpopshop.com.ua вирізняється доволі зручним дизайном та поділом товарів за чоловічими та жіночими групами, що полегшує навігацію. Разом із тим, недоліками є незручна верхня панель і футер, які можуть викликати естетичне несприйняття. У товарному асортименті переважає офіційний стаф та корейська косметика.

Інший ресурс — beyourselfkpop.com — має функціональний фільтр, але загальне враження знижується через слабкий візуальний стиль: строкаті кольори, різні шрифти, що ускладнює сприйняття. Асортимент включає бокси з неофіційною атрибутикою та товари на стику з аніме-тематикою, що розширює аудиторію, але відхиляє від цільового образу K-pop.

Сайт також пропонує фільтрацію, однак головна сторінка виглядає перевантаженою, а візуальне оформлення в чорно-біло-фіолетовій палітрі не всім буде до вподоби. Перевагою є широкий вибір товарів: від боксових наборів до канцтоварів, що свідчить про орієнтацію на широку фан-базу.

Поточні тенденції на ринку вебплатформ для фанатських товарів передбачають фокус на емоційний UX-дизайн, мобільну оптимізацію, швидкий доступ до новинок, соціальну інтеграцію (зокрема, через Instagram або TikTok) та використання елементів гейміфікації (наприклад, бонуси за покупки чи участь у голосуваннях за групу). Також популярними стають функції попереднього замовлення на релізи та персоналізовані

рекомендації, які покращують користувацький досвід і утримують увагу.

Відтак, нова платформа має змогу зайняти конкурентну нішу за умови поєднання візуальної привабливості, функціональної зручності та уважного ставлення до потреб фанатської спільноти. Сайт повинен не лише продавати, а й створювати відчуття приналежності до особливого онлайн-ком'юніті.

1.2 Дослідження сучасних тенденцій у веброзробці

У цьому підрозділі розглядаються основні напрямки веброзробки, актуальні інструменти та технології, які визначають якість та ефективність сучасних вебплатформ. Аналіз проводиться з урахуванням особливостей цільової аудиторії платформи — шанувальників K-pop, що формує вимоги до дизайну, функціональності та інтерфейсу.

1.2.1 Основні технології веброзробки

Сучасний процес веброзробки неможливо уявити без використання фундаментальних технологій, які слугують базисом для створення будь-якого повноцінного сайту. HTML (HyperText Markup Language) та CSS (Cascading Style Sheets) залишаються основними будівельними блоками інтернету, оскільки саме за їх допомогою створюється структура, зміст та візуальне представлення вебсторінок. HTML формує каркас сторінки, організовуючи заголовки, абзаци, зображення, таблиці та інші елементи, тоді як CSS визначає зовнішній вигляд цих елементів — кольори, шрифти, розташування, а також поведінку при зміні розміру вікна браузера. В епоху, коли користувачі взаємодіють із сайтом із різноманітних пристроїв, адаптивність і гнучкість дизайну стали критично важливими, що й забезпечує CSS.

Для підвищення ефективності та швидкості розробки в цьому проєкті використовується також фреймворк Bootstrap. Це потужний інструмент, який надає набір попередньо розроблених стилів, компонентів та сіткових систем, що дозволяє значно скоротити час розробки інтерфейсу. Bootstrap також підтримує мобільну адаптацію за замовчуванням, що особливо важливо в умовах переважаючого мобільного трафіку.

Ще одним ключовим елементом сучасної веброзробки є JavaScript — мова програмування, що забезпечує динамічну поведінку сторінки. Завдяки їй користувач може взаємодіяти із сайтом у режимі реального часу: відкривати меню, змінювати дані на сторінці без її перезавантаження, переглядати галереї, залишати коментарі тощо. У рамках проєкту застосовується бібліотека jQuery, яка є спрощенням класичного JavaScript-коду. Вона дозволяє легко маніпулювати елементами DOM, створювати анімації, обробляти події, а також виконувати AJAX-запити.

Технологія AJAX (Asynchronous JavaScript and XML) дає змогу асинхронно обмінюватися даними із сервером, не перезавантажуючи сторінку. Це створює відчуття «живого»

сайту, який миттєво реагує на дії користувача, підвищуючи тим самим рівень взаємодії та задоволеності.

На серверному рівні реалізація логіки та **10** обробки даних здійснюється за допомогою мови програмування PHP. PHP широко застосовується в індустрії завдяки своїй гнучкості, простоті інтеграції з базами даних і сумісності з численними CMS, зокрема WordPress. Саме PHP дозволяє виконувати динамічні обчислення, обробляти запити користувачів, працювати з формами, керувати сесіями та створювати персоналізований вміст.

1.2.2 Інструменти для розробки дизайну

UI/UX-дизайн є невід'ємною частиною сучасного процесу веброзробки, оскільки саме він формує те, як користувач сприймає сайт і як з ним взаємодіє. Від того, наскільки інтерфейс зручний, зрозумілий та естетично привабливий, залежить не лише перше враження від платформи, а й бажання користувача повертатися на сайт знову. У наш час увага до деталей у дизайні стає одним із головних чинників успіху проєкту. Особливо це актуально в контексті створення сайту, орієнтованого на молодіжну фанатську аудиторію, яка звикла до яскравих, стильних і динамічних цифрових рішень.

Для розробки дизайну вебплатформи у цьому проєкті було обрано Figma — одну з найпрогресивніших систем для створення макетів інтерфейсів, що набула величезної популярності серед дизайнерів у всьому світі. Її вибір обумовлений не лише функціональністю, але й зручністю користування, хмарною природою, підтримкою командної роботи та інтуїтивно зрозумілим інтерфейсом. На відміну від більш традиційних графічних редакторів, Figma не потребує встановлення — вона працює прямо в браузері, що суттєво полегшує доступ до файлів **1** з будь-якого пристрою і **1** в будь-який час.

1 Однією з ключових переваг Figma є її здатність підтримувати спільну **1** роботу **1** в реальному часі, що особливо важливо при командній розробці проєктів. Дизайнери, розробники та замовники можуть водночас переглядати, коментувати або редагувати макети, що значно підвищує ефективність комунікації. Крім того, система компонентів дозволяє створювати бібліотеки стилів і повторно використовуваних елементів, що забезпечує візуальну узгодженість інтерфейсу й економить час.

Figma також має потужний функціонал для створення інтерактивних прототипів, які дають змогу наочно представити, як саме буде працювати сайт ще до його розробки. Це особливо цінно на етапі погодження дизайну з клієнтом або тестування користувацьких сценаріїв. У рамках цього проєкту за допомогою Figma було створено повноцінну систему макетів, що охоплює головну сторінку, каталог товарів, картку

товару, особистий кабінет користувача та інші ключові елементи вебплатформи.

Окремо варто зазначити, що завдяки можливості створення адаптивного дизайну Figma дозволяє ефективно опрацьовувати версії сайту для різних типів пристроїв — від мобільних телефонів до настільних комп'ютерів. Це надзвичайно важливо з огляду на те, що значна частина користувачів споживає контент саме з мобільних пристроїв.

Отже, використання Figma як основного інструменту для UI/UX-дизайну забезпечує не лише високу якість візуальної частини проєкту, а й дозволяє ефективно організувати роботу над розробкою, враховуючи сучасні вимоги до гнучкості, інтерактивності та кросплатформеності інтерфейсу.

1.2.3 Системи управління вмістом

У сучасному веброзробленні системи управління контентом (CMS) **9** відіграють надзвичайно важливу роль, оскільки дозволяють ефективно керувати динамічним вмістом сайту без потреби в постійному втручанні розробників. Завдяки таким системам адміністратор платформи отримує можливість швидко додавати, змінювати чи видаляти контент, керувати користувачами, а також налаштовувати функціональні модулі в зручному візуальному інтерфейсі. У межах цього проєкту для реалізації вебплатформи було обрано WordPress — одну з найпопулярніших у світі CMS, яка вже давно зарекомендувала себе як надійне, гнучке й масштабоване рішення.

WordPress розроблений як відкрита платформа з потужною спільнотою, що активно створює теми, плагіни й технічну документацію. Його перевагою є інтуїтивно зрозумілий інтерфейс, завдяки чому ним можуть користуватися навіть ті, хто не має технічного досвіду. Це особливо актуально в умовах швидкозмінного середовища електронної комерції, де адміністратор повинен мати змогу оперативного оновлювати каталог товарів, додавати новини чи анонси подій, не звертаючись щоразу до програміста.

Система WordPress підтримує інтеграцію з безліччю розширень, серед яких особливе значення мають WooCommerce — плагін для електронної комерції, а також різні SEO-плагіни, модулі захисту, аналітики та багатомовності. Таким чином, CMS перетворюється на повноцінну екосистему, де кожен сайт можна адаптувати під конкретні бізнес-потреби — чи то блог, чи інтернет-магазин, чи портал для фанатів певного культурного напрямку, як у цьому проєкті.

З технічного боку WordPress базується на мові програмування PHP і взаємодіє з базою даних MySQL або MariaDB. Це дозволяє гнучко формувати структуру сайту, зберігати великий обсяг контенту та ефективно обробляти запити користувачів. Для розробника це означає можливість створення власних шаблонів і плагінів, глибоке налаштування

функціоналу, а також доступ до всіх аспектів роботи вебплатформи.

Інтеграція CMS WordPress у рамках цього проєкту дала змогу поєднати потужний інструментарій управління контентом із сучасним візуальним і технічним рішенням. Це дозволило досягти високого рівня зручності як для кінцевих користувачів сайту, так і для адміністратора ресурсу. У результаті створено не лише зручну торгову платформу, а й простір, що може легко масштабуватися, оновлюватися та адаптуватися до нових потреб цільової аудиторії.

1.3 Обґрунтування вибору інструментів і технологій для розробки вебплатформи

Для реалізації фронтенд-частини платформи обрано базові технології HTML5, CSS3 та JavaScript. Це найпоширеніші і найбільш стандартизовані інструменти веброзробки, які дозволяють створити логічну структуру сторінок, оформити їх візуально привабливо та забезпечити інтерактивність для користувачів. Застосування цих технологій є оптимальним вибором з огляду на їх універсальність, підтримку усіма сучасними браузерями та наявність великої кількості ресурсів для розробників.

Для більш швидкого та ефективного створення адаптивного дизайну було використано CSS-фреймворк Bootstrap 5. Цей інструмент є **12** популярним серед розробників завдяки своїй зручності, великому набору готових компонентів, а також простоті інтеграції з JavaScript. Bootstrap значно спрощує процес розробки, дозволяючи не витрачати багато часу на написання базових стилів та елементів інтерфейсу з нуля. Крім того, його адаптивна сітка забезпечує коректне відображення сайту на різних пристроях — від великих моніторів до мобільних телефонів. Хоча на ринку існують альтернативи, такі як Tailwind CSS чи Foundation, Bootstrap 5 був обраний саме за його баланс між простотою, готовими компонентами і широкою підтримкою спільноти.

Для управління контентом платформи використовувався **6** WordPress — одна з найпопулярніших систем управління контентом (CMS) у світі. Вона дозволяє швидко створювати сайти різного рівня складності, маючи при цьому зручний інтерфейс для редагування та налаштування. Велика кількість доступних плагінів і тем робить WordPress гнучким рішенням, яке можна легко адаптувати під специфічні потреби. На відміну від інших CMS, таких як Joomla чи Drupal, WordPress має більш дружній інтерфейс і менший поріг входу, що робить його ідеальним вибором для проєктів з обмеженими ресурсами часу і технічної підтримки.

8 На основі проведеного аналізу сучасних технологій, розглянутих у розділі 1, було обрано оптимальний стек інструментів для створення вебплатформи. Вибір HTML5, CSS3 та JavaScript зумовлений їх універсальністю, широкою підтримкою, а також можливістю реалізувати адаптивний, інтерактивний та швидкий інтерфейс. Фреймворк Bootstrap 5 забезпечує ефективну розробку мобільно-орієнтованого дизайну з

мінімальними затратами часу.

Використання бібліотеки jQuery спрощує взаємодію з DOM-структурою та реалізацію інтерактивних елементів, а AJAX дозволяє реалізувати асинхронну взаємодію без перезавантаження сторінки, що є важливим для покращення користувацького досвіду. Для візуального проектування обрано Figma як найзручніший інструмент із можливістю спільної роботи та створення адаптивних макетів.

На рівні бекенду вибір зупинився на WordPress у поєднанні з PHP, що забезпечує гнучке керування вмістом і дозволяє легко розширювати функціонал за рахунок плагінів. Зокрема, інтеграція WooCommerce дозволила створити повноцінний інтернет-магазин, адаптований до потреб фанатської спільноти.

Таким чином, використання зазначених технологій дозволило створити ефективну, адаптивну та привабливу вебплатформу, яка відповідає потребам сучасної молодіжної аудиторії та відповідає вимогам до сучасних цифрових сервісів.

2 ПРОЕКТУВАННЯ СТРУКТУРИ ТА ПІДГОТОВКА МАКЕТІВ ДИЗАЙНУ

2.1 Проектування структури вебплатформи

На першому етапі розробки вебплатформи «Fenix K-pop shop» було здійснено проектування її структури з урахуванням зручності користувача, логіки роботи системи та подальшої реалізації у середовищі WordPress. Основна мета — створити зрозумілу, функціональну та адаптивну платформу для продажу товарів, пов'язаних із K-поп тематикою.

Було визначено основні сторінки сайту: головна, каталог товарів, окремі сторінки товарів, кошик, форма оформлення замовлення, а також сторінки з інформацією про доставку, оплату та контактами. Така структура дозволила охопити всі потреби типового користувача — від перегляду продукції до здійснення замовлення. На рисунку 2.1 зображено схематичну структуру сайту.

Рисунок 2.1 — схематична структура сайту.

Кожна сторінка мала чітке призначення. Головна сторінка містила банер, актуальні товари та навігацію. Каталог товарів мав блок із фільтрами та адаптивну сітку товарів. Сторінки товарів — з детальним описом, фото, ціною та кнопкою «Додати в кошик». Сам кошик був реалізований у вигляді модального вікна з можливістю змінювати кількість товару та переглядати загальну суму. Етап оформлення замовлення передбачав вибір доставки, способу оплати та введення контактної інформації.

Також на цьому етапі було запроектовано взаємодію компонентів: як саме працює

кошик, як передаються дані про товари, як відображається замовлення, які сценарії взаємодії може пройти користувач. Уся структура сайту була розроблена відповідно до принципів UX/UI-дизайну та логіки CMS WordPress для подальшого впровадження функціоналу через шаблонні файли, PHP-функції, JavaScript та плагіни.

2.2 Розробка макетів дизайну

2.2.1 Постановка завдання і вимог до дизайну

Розробка дизайну вебплатформи розпочалася з формування основних вимог до її функціональності та візуального оформлення. Одним із ключових завдань стало забезпечення інтуїтивно зрозумілої навігації між основними розділами сайту — зокрема, інформацією, каталогом товарів та сторінками гуртів. Важливою вимогою була можливість зручного перегляду продукції за категоріями, що дозволяє швидко орієнтуватися у великому асортименті товарів.

Крім того, у процесі проектування враховано необхідність реалізації повноцінного функціоналу для оформлення замовлень, включно з формами зворотного зв'язку, що забезпечують комунікацію користувача з адміністрацією сайту. Значну увагу приділено адаптивності, адже більшість користувачів потенційно взаємодіятимуть із платформою **2** через мобільні пристрої. Саме **2** тому **2** сайт має коректно відображатися на різних екранах і зберігати зручність користування незалежно від розміру дисплея.

Щодо візуальних аспектів, дизайн орієнтований на естетику, притаманну K-рор культурі. Було вирішено використовувати яскраву, проте збалансовану стилістику з пастельною кольоровою гамою та динамічними графічними елементами, які створюють емоційний настрій, характерний для фанатських середовищ. Контент подається у структурованому вигляді, що сприяє зручності сприйняття інформації. Сторінки доповнено привабливими банерами із зображеннями гуртів, новинними блоками та візуально акцентованими зонами для реклами та акцій. У дизайні гармонійно поєднано типографіку та графіку — шрифти підбрано з урахуванням читабельності та загального стилю, що допомагає створити цілісний і привабливий образ платформи.

Загалом, розробка дизайну спрямована на поєднання зручного користувацького досвіду з візуальною ідентичністю, яка відповідає очікуванням фанатської аудиторії K-рор.

2.2.2 Розробка макетів дизайну сторінок сайту

На цьому етапі розробки було здійснено проектування візуального вигляду вебсайту, що є ключовим чинником для ефективної взаємодії користувача з ресурсом. Дизайн макету — це не лише естетичне оформлення сторінки, а й інструмент, який забезпечує

логічну структуру подання інформації, зручність навігації, відповідність тематиці сайту **4** та формує перше враження про заклад в очах відвідувача.

Під час створення макету головної сторінки вебсайту було враховано специфіку діяльності розважального закладу. Дизайн побудовано з орієнтацією на молодіжну аудиторію, що шукає яскравих емоцій, якісного сервісу та зручного способу отримання інформації про заходи, послуги й новини клубу. Відповідно, у структурі макету було передбачено кілька основних логічних блоків.

Верхня частина сторінки містить логотип, який візуально репрезентує бренд, та головне меню з базовими розділами: «Головна», «Про нас», «Інформація», «Каталог». Праворуч також розміщено іконки соціальних мереж, **4** що дозволяє швидко перейти до сторінок клубу у Facebook, Instagram та Twitter. На рисунку 2.2 зображено дизайн шапки сторінки.

Рисунок 2.2 — дизайн шапки сторінки.

Це забезпечує додаткові канали взаємодії з аудиторією та сприяє формуванню онлайн-спільноти.

Центральний банер виконує роль візуального акценту, а також слоган, який передає основне послання бренду — створення атмосфери, що дарує незабутні враження. Цей елемент одразу привертає увагу відвідувача та формує емоційне сприйняття сайту (рис. 2.3)

Рисунок 2.3 — банер та гасло.

Далі розміщено блок з найпопулярнішими покупками. Такий формат дозволяє швидко ознайомити користувача з найважливішим контентом, використовуючи картки з зображенням, коротким описом і кнопкою для переходу до повної інформації (рис. 2.4)

Рисунок 2.4 — блок де представлені найпопулярніші покупки

Для забезпечення інтерактивності та збору зворотного зв'язку передбачено перегляд відгуків інших користувачів та форму оцінювання, де користувач може залишити відгук, вказавши кількість зірок та коментар. Це не лише підвищує залученість аудиторії, але й допомагає адміністрації отримувати важливу інформацію щодо якості обслуговування (рис.2.5)

Рисунок 2.5 — зворотній зв'язок.

Завершує макет футер, в якому розміщено контактні дані, години роботи, посилання на соціальні мережі та інтерактивну карту Google, що дозволяє легко знайти місце

розташування закладу. На рисунку 2.6 зображений футер.

Рисунок 2.6 — підвал сторінки.

Після створення головної сторінки, яка виконує роль вітрини сайту, було розроблено також внутрішні сторінки, що дозволяють користувачеві детальніше ознайомитися з діяльністю закладу, умовами обслуговування та конкретними товарами. Кожна зі сторінок зберігає єдиний стиль оформлення: темне тло, виразні червоні акценти, логічна структура, іконографіка та чітка типографіка.

Сторінка «Про нас» виконує функцію ознайомлення користувача з суттю діяльності закладу, його концепцією, цінностями та перевагами. Інформація подана стисло, у зручній формі, з акцентом на ключових моментах. Використано маркований список, який дозволяє легко сприймати текст і виділяти основне.

Особливу роль відіграє візуальний елемент — стилізоване зображення червоного фенікса, яке є важливою складовою візуального образу бренду. Воно символізує енергію, переродження та унікальність, підкреслюючи дух закладу.

Нижче розташовано три інтерактивні картки, кожна з яких репрезентує окрему послугу або перевагу, наприклад, якість обслуговування, індивідуальний підхід, комфортну атмосферу тощо. Такий блок допомагає коротко окреслити ключові аспекти без перевантаження користувача зайвими деталями. На рисунку 2.7 зображено дизайн сторінки «Про нас».

Рисунок 2.7 — дизайн сторінки «Про нас»

Сторінка «Інформація» орієнтована на інформування користувача про організаційні та логістичні аспекти обслуговування: умови доставки, способи оплати, гарантії та правила повернення товару.

Верхня частина містить фотогалерею, в якій зображено процес пакування та відправлення товарів, що підкреслює відкритість і чесність бренду. Всі фото обрамлені в червоні рамки, що відповідає стилістиці сайту й створює візуальну цілісність.

Текстові блоки чітко структуровані. Кожен підрозділ (доставка, гарантія, повернення, оплата) супроводжується коротким і доступним описом. Для зручності користувача важливі деталі (наприклад, терміни, умови) виділені червоним кольором. Це покращує навігацію по сторінці та дозволяє швидко знайти необхідну інформацію. На рисунку 2.8 зображено дизайн сторінки «Інформація».

Рисунок 2.8 — дизайн сторінки «Інформація»

Сторінка товару, слугує для детального представлення окремого товару. Вона включає

ключові елементи, необхідні для ознайомлення з продуктом і здійснення покупки. У верхній частині розміщено велике зображення товару.

Поряд вказано назву товару, його ціну, короткий опис. Текстовий опис добре структурований, з розподілом на абзаци, що полегшує сприйняття.

Під описом розміщено кнопку «Додати в кошик», яка виконує функцію заклику до дії. Це стандартний елемент для інтернет-магазинів, який дозволяє швидко оформити замовлення.

Нижче представлений блок із рекомендованими або подібними товарами. Це підвищує ймовірність повторної взаємодії користувача з сайтом і стимулює продаж. На рисунку 2.9 зображено дизайн сторінки з товаром.

Рисунок 2.9 — дизайн сторінки товару

Усі макети реалізовані з урахуванням принципів UX/UI-дизайну: вони прості у використанні, логічно структуровані, мають привабливий вигляд і сприяють утриманню користувача на сайті. Послідовність оформлення та візуальна єдність забезпечують позитивне враження та довіру з боку відвідувача.

3 РОЗРОБКА ВЕБПЛАТФОРМИ

3.1. Реалізація вебплатформи

На цьому етапі дипломної роботи було здійснено безпосереднє втілення розробленого дизайну та функціоналу у вигляді повноцінної вебплатформи. Для реалізації обрано систему керування контентом WordPress, що дозволяє ефективно поєднувати інформаційний портал і онлайн-магазин у межах одного ресурсу. Завдяки гнучкій архітектурі та розширеній екосистемі плагінів WordPress забезпечує можливість швидкого налаштування, адаптації під потреби цільової аудиторії та підтримки масштабування проєкту в майбутньому.

Реалізація включала встановлення CMS, налаштування базових параметрів, створення структури сайту, інтеграцію дизайн-макетів, підключення необхідного функціоналу через плагіни та тестування роботи платформи. Усі етапи були спрямовані на досягнення головної мети — створення зручного, привабливого й функціонального вебресурсу для фанатів K-pop, який би задовольняв як інформаційні, так і комерційні потреби користувачів.

3.1.1 Створення структури сайту на базі WordPress

Початковий етап розробки полягав у створенні повноцінної HTML-структури сайту з

використанням фреймворку Bootstrap 5. Спочатку були зверстані всі основні сторінки: головна, каталог товарів, сторінка окремого товару, кошик, оформлення замовлення, а також сторінки «Про нас» і «Інформація». На рисунку 3.1 зображена структура HTML-верстки.

Рисунок 3.1 — структура HTML-верстки.

При верстці враховувалися адаптивність, візуальна привабливість та стилістика, характерна для кпоп-культури. Усі елементи інтерфейсу — включно з кнопками, блоками товарів, формами та слайдерами — були реалізовані вручну за допомогою компонентів Bootstrap.

Після завершення верстки ці шаблони поступово переносилися у середовище WordPress. Для цього була створена кастомна тема, в якій вся верстка була поділена на окремі шаблонні файли згідно зі структурою WordPress: header.php, footer.php, index.php, page.php, single.php, functions.php. Це дозволило інтегрувати раніше створену HTML-структуру у динамічну CMS, забезпечивши її повноцінне функціонування.

Особливу увагу було приділено навігаційному меню. Для правильного виведення багаторівневої навігації було реалізовано власний Walker-клас, який дозволив коректно відображати всі рівні меню з підтримкою dropdown у стилі Bootstrap. Меню автоматично генерується на основі даних з панелі адміністратора WordPress, що значно спрощує подальше редагування. На рисунку 3.2 зображена структура PHP-верстки

Рисункунок 3.2 - структура PHP-верстки.

В підсмку вдалося досягнути структури представленої на рисунку 3.3

Рисунок 3.2 — схема стуркути вебсистеми на WordPress

3.1.2 Верстка та налаштування дизайну згідно з макетами

На першому етапі була створена власна тема WordPress. Для цього у папці wp-content/themes було створено нову директорію з унікальною назвою теми — наприклад, fenix-krop-shop. У цій папці були розміщені базові файли теми: style.css, index.php, functions.php та поступово додані інші шаблонні файли. У файлі style.css було задано службовий блок коментаря з основною інформацією про тему: назва, автор, версія, опис тощо. Цей блок потрібен, щоб WordPress розпізнав директорію як повноцінну тему (рис 3.1.1)

Рисунок 3.3 — Створення власної теми

Файл `index.php` став базовим шаблоном, що використовується за замовчуванням, якщо не задані інші конкретні шаблони. Через файл `functions.php` підключались стилі, скрипти, підтримка меню, реєстрація віджетів та інші можливості теми.

2 Після того, як тема була активована через панель адміністратора WordPress, відбувалося поступове перенесення HTML-розмітки з готової верстки в шаблонні файли теми, з урахуванням специфіки WordPress.

Перенос готової верстки у WordPress розпочався зі структурування HTML-коду на окремі шаблонні файли, відповідно до логіки роботи системи управління контентом. Головні елементи сторінки, такі як шапка, підвал, меню, а також контентні блоки були виділені в окремі компоненти, що полегшило їх подальше повторне використання та підтримку

Далі відбувалося підключення стилів і скриптів у функціональному файлі теми (`functions.php`). Стилів та Bootstrap, розроблені згідно з макетами, були підключені через правильні WordPress-хуки, що гарантувало їх коректне завантаження на сторінках сайту і уникнення конфліктів з іншими плагінами чи темами. Аналогічно, JavaScript-файли, які відповідали за інтерактивність і анімації, також підключалися централізовано. На рисунку 3.4 зображено підключення стилів на скриптів у `functions.php`.

Рисунок 3.4 — підключення стилів та скриптів у `functions.php`.

Для динамічного відображення контенту використовувалися шаблонні теги WordPress у поєднанні з плагіном Advanced Custom Fields (ACF). За допомогою ACF були створені кастомні поля для товарів, описів, фотографій та інших елементів, що дозволило адміністраторам сайту легко і гнучко керувати вмістом через адмін-панель без необхідності змінювати код. Це значно спростило процес оновлення інформації на сайті і забезпечило більшу зручність у роботі з контентом.

3.2. Реалізація функціоналу вебплатформи

Після створення базової структури теми подальша робота зі сторінками значно спростилася. Для формування кожної сторінки використовували стандартні функції WordPress — `get_header()` та `get_footer()`, які відповідали за підключення шапки і підвалу сайту. Основний контент виводився за допомогою шаблонних тегів і функцій, зокрема через `the_content()` або ж за допомогою виведення полів, створених через плагін Advanced Custom Fields (ACF), за допомогою функції `the_field()`.

Для реалізації багаторівневого меню, сумісного з Bootstrap 5, у темі було підключено кастомний Walker-клас. Він відповідає за генерацію правильної HTML-структури меню з усіма необхідними класами та атрибутами. Це особливо важливо, оскільки стандартний механізм WordPress не підтримує вкладені меню у форматі, який вимагає Bootstrap 5

для коректної роботи дропдаунів. Підключення цього класу виконувалося у файлі `functions.php` за допомогою функції `register_navwalker()`, яка завантажує файл `class-bootstrap-multilevel-navwalker.php`. Меню було зареєстроване як 'primary' для подальшого керування через адмін-панель WordPress.

Використання кастомного Walker дозволило створити багаторівневу навігацію до трьох рівнів вкладеності без хардкоду, повністю динамічно, із збереженням адаптивності та стилістики Bootstrap. Це забезпечило зручність не лише для розробника, а й для адміністратора сайту, який міг самостійно змінювати структуру меню без втручання в код.

Також у `functions.php` відбувалася ініціалізація віджетів сайдбару через функцію `fenix_widgets_init()`, що реєструвала область для віджетів із необхідними HTML-обгортками для коректного виводу. Наявність такої зони дозволяє адміністратору сайту самостійно додавати текстові блоки, пошук, списки категорій або інші віджети через стандартний інтерфейс WordPress, без необхідності змінювати шаблони вручну.

На рисунках 3.5 та 3.6 показано підключення кастомного Walker-класу та ініціалізацію віджетів сайдбару відповідно.

Рисунок 3.5 — підключення кастомного Walker-класу.

Рисунок 3.6 - підключення ініціалізацію віджетів сайдбару.

3.2.1 Налаштування каталогу товарів

Каталог товарів був реалізований на основі стандартного механізму WordPress із використанням кастомних типів записів та плагіна Advanced Custom Fields (ACF) для створення необхідних полів товару. Замість класичного використання WooCommerce **11** було прийнято рішення створити власну структуру для повного контролю над виглядом, версткою та логікою відображення товарів у каталозі.

Перш за все, у файлі `functions.php` був зареєстрований кастомний тип запису `product`, що дозволило створювати та керувати окремими товарами через адмін-панель WordPress. Для кожного товару за допомогою ACF були додані індивідуальні поля: назва, зображення, опис, ціна, категорія та наявність. Це дало змогу зберігати всю необхідну інформацію про продукт у зручному форматі.

Вивід товарів на фронтенді сайту реалізовувався через створення шаблонного файлу `archive-product.php`, який відповідав за загальний вигляд каталогу. У середині цього файлу виконувався цикл WordPress (The Loop), що перебирає всі опубліковані записи типу `product`. Для кожного елемента виводились поля, задані через ACF, за допомогою

функцій `the_field()` або `get_field()`. Вся розмітка була адаптована під Bootstrap 5 для збереження єдиного стилю сайту, включно з використанням карток (`.card`) та сітки (`.row`, `.col-*`) для зручного і візуально приємного представлення товарів.

Додатково, для зручності навігації та фільтрації, товари були згруповані за категоріями. Це було реалізовано через створення таксономії, аналогічної до рубрик (`categories`), яка дозволила легко організовувати контент каталогу й додати фільтри або меню категорій.

Результатом стало повноцінне виведення каталогу товарів, яким можна керувати через адмін-панель без потреби змінювати код. Такий підхід забезпечив гнучкість і простоту в оновленні контенту, а також надавав можливість масштабування функціоналу в майбутньому — наприклад, додавання кнопки “Замовити”, варіантів кольору, розміру тощо.

На рисунку 3.7 зображено приклад шаблону архіву товарів та виводу полів ACF для кожного елемента каталогу.

Рисунок 3.7 — приклад шаблону архіву товарів та виводу полів ACF для кожного елемента каталогу.

3.2.2 Організація кошика та замовлень

У рамках навчального проєкту вебплатформи реалізація кошика та оформлення замовлень була виконана за допомогою JavaScript та вбудованих можливостей браузера. Це **7** дозволило створити інтерактивний і зручний інтерфейс для користувача без потреби в складних серверних механізмах.

Функціонал кошика реалізовано через модальне вікно, яке відкривається після натискання на іконку кошика у шапці сайту. Основна логіка роботи знаходиться у файлі `script.js`. Додавання товару до кошика виконується після натискання на кнопку «Додати в кошик». Інформація про товар (назва, ціна, зображення) зчитується з елементів сторінки та додається до масиву `cart`. При повторному додаванні одного й того ж товару оновлюється лише його кількість.

Всі зміни у кошику — збільшення чи зменшення кількості товарів, їх видалення, підрахунок загальної суми — відбуваються динамічно. Для збереження даних навіть після перезавантаження сторінки використовується `LocalStorage`, що дозволяє тримати стан кошика незмінним до моменту очищення або оформлення замовлення.

Окрему увагу приділено зручності оформлення: користувач може вибрати спосіб доставки (Нова пошта або Укрпошта), і залежно від вибору з'являються відповідні поля для заповнення. Також реалізована функція вибору способу оплати — оплата онлайн або при отриманні. Після підтвердження замовлення з'являється відповідне модальне

повідомлення, яке імітує завершення покупки.

Такий підхід дозволяє ефективно продемонструвати принципи роботи кошика в онлайн-магазині та взаємодію з користувачем у зручній і наочній формі.

3.3 Тестування та оцінка якості вебплатформи

Після завершення основного етапу розробки було проведено комплексне тестування вебплатформи з метою перевірки її функціональності, сумісності з різними пристроями та браузерами, швидкості завантаження, а також зручності використання для цільової аудиторії.

На етапі функціонального тестування перевірялася коректність роботи ключових елементів: додавання товарів до кошика, оформлення замовлення, робота форм зворотного зв'язку та підписки на новини. Усі перевірені сценарії виконувалися відповідно до очікувань, без критичних помилок.

Особливу увагу було приділено адаптивності інтерфейсу. Тестування відбувалося на основних типах пристроїв — смартфонах, планшетах та ноутбуках. Додатково було перевірено відображення сайту в найпопулярніших браузерах, зокрема Google Chrome, Mozilla Firefox та Safari. Інтерфейс залишався стабільним, а контент — читабельним і зручним у користуванні.

Для оцінки швидкості завантаження використовувався сервіс Google PageSpeed Insights, що дозволяє виявити слабкі місця у продуктивності. Було виявлено потребу в оптимізації зображень та ввімкнено кешування на рівні сервера, що суттєво покращило загальні показники швидкості, зокрема для мобільних пристроїв.

У рамках юзабіліті-тестування було проведено опитування серед потенційних користувачів — шанувальників K-pop віком 16–40 років. Учасники тестування позитивно оцінили загальну структуру сайту, привабливість дизайну та простоту оформлення замовлень. Було зібрано низку відгуків, на основі яких внесено незначні корективи в розташування деяких елементів інтерфейсу.

Таблиця 2.1 – Результати тестування вебплатформи

Тип тестування

Критерій перевірки

Результат

Коментар

Функціональне

Додавання товару до кошика

Успішно

Без збоїв

Оформлення замовлення

Успішно

Дані передаються корентно

Робота форм зворотного зв'язку

Успішно

Повідомлення надсилається

Підписка на новини

Успішно

Пошта додається до списку

Адаптивність

Відображення на мобільних пристроях

Коректно

Усі блоки адаптивні

Перевірка у Chrome/Firefox/Safari

Без помилок

Однакове відображення

Продуктивність

Швидкість завантаження

PageSpeed: 89 моб / 95 десктоп

Після оптимізації зображень

Юзабіліті

Інтерфейс користувача

Позитивна оцінка

За результатами опитування

3.4 SEO-оптимізація та адаптивність вебплатформи

3.4.1 SEO-оптимізація сайту

У межах реалізації вебплатформи було застосовано базові та ефективні стратегії внутрішньої SEO-оптимізації.

Насамперед, була створена семантично правильна HTML-структура. Кожна сторінка використовувала теги header, main, footer, section, що дозволяло логічно впорядкувати контент та покращити його індексацію пошуковими системами. Заголовки на сторінках структурувались у правильній ієрархії — h1 для головного заголовка, h2 для розділів і h3 для підрозділів. Це забезпечило логічну послідовність подання інформації як для користувачів, так і для пошукових роботів.

Для кожної сторінки були встановлені унікальні мета-теги: title та meta description. Вони містили ключові слова, пов'язані з тематикою сайту — наприклад, назви к-поп гуртів, товарів або акцій. Це допомагає **3** пошуковим системам точніше визначити зміст сторінки, а користувачам — зрозуміти, чому саме ця сторінка відповідає їхньому запиту. Мета-опис слугує коротким описом сторінки у видачі Google, і правильно сформульований текст може впливати на кількість переходів на сайт.

Зображення на сайті супроводжувалися атрибутами alt, що виконують дві функції: пояснюють зміст зображення для людей з порушеннями зору (через екранні читалки) та допомагають пошуковим системам індексувати графічний контент. Назви зображень і значення alt також були сформульовані з використанням ключових слів.

Крім того, сайт був оптимізований для швидкого завантаження. Використання стиснення зображень, відкладеного завантаження (lazy loading) та мінімізації CSS- і JS-файлів дозволило зменшити час завантаження сторінок. Це важливо не лише для зручності **3** користувачів, а й тому, що швидкість сайту є одним із факторів ранжування **3** у пошукових системах.

3 Загалом реалізована SEO-оптимізація забезпечила технічну готовність сайту до якісної індексації, підвищила релевантність контенту для пошукових запитів та створила фундамент для подальшого просування вебплатформи у результатах пошуку.

3.4.2 Адаптивна верстка для різних пристроїв

Однією з ключових вимог до сучасного вебсайту є його адаптивність — здатність коректно відображатися на різних типах пристроїв: комп'ютерах, планшетах та мобільних телефонах. У межах проєкту адаптивна верстка реалізовувалась із використанням фреймворку Bootstrap 5, що забезпечило зручну та швидку підтримку мобільної версії сайту.

Для досягнення повної адаптивності було застосовано сіткову систему Bootstrap (container, row, col-), яка дозволяє динамічно змінювати розташування елементів в залежності від ширини екрану. Наприклад, блоки товарів на головній сторінці відображаються у 4 колонки на десктопі, у 2 — на планшеті, і в одну — на мобільному пристрої. У додатках А та Б наведено, відображення одного й того ж товарного блоку на планшеті та мобільному пристрої відповідно.

Також особлива увага приділялася адаптації меню. Використано офіційний компонент Bootstrap Navbar, який автоматично перетворюється на бургер-меню **5 на мобільних пристроях**. Його роботу можна продемонструвати на рисунку 3.8, де видно, як головне меню згинається і відкривається при натисканні на кнопку.

Рисунок 3.8 — розгорнуте головне меню на екрані телефону

Окрім загального компонування, було передбачено зміну розмірів шрифтів, кнопок та зображень для кращої читабельності та взаємодії на малих екранах. Застосовано утиліти Bootstrap, такі як d-none d-md-block або text-center text-md-start, які дозволяють керувати видимістю та вирівнюванням елементів у залежності від ширини пристрою.

Ще один важливий момент — адаптивність форм. Усі форми (оформлення замовлення, вибору доставки тощо) автоматично масштабуються завдяки класам Bootstrap (form-control, form-group, w-100). Це дозволяє уникати горизонтального скролу **5 на мобільних пристроях, достатньо великі** для торкання пальцем, а зайві елементи — приховані.

Рисунок 3.9 — модальне вікно оформлення замовлення на мобільному пристрої

Загалом адаптивна верстка забезпечила зручну навігацію сайтом на будь-якому пристрої, що позитивно впливає не лише на користувацький **5 досвід, а й** на SEO-показники ресурсу.

Посилання

Це джерела виділених збігів у вашому документі. Кожен збіг позначено темно-зеленим числом, яке відповідає вказаному тут джерелу. Джерела впорядковані за схожістю — чим вищий бал, тим сильніше збіг.

#	Джерело	%
1	openarchive.nure.ua	0.4%
2	lviv.rozrobka-sajtiv.in.ua	0.3%
3	elbuz.com	0.3%
4	knute.edu.ua	0.2%
5	repository.hneu.edu.ua	0.2%
6	hostpro.ua	0.2%
7	dspace.znu.edu.ua	0.2%
8	ea.donntu.edu.ua	0.1%
9	catalog.library.tnpu.edu.ua	0.1%
10	ela.kpi.ua	0.1%
11	sci.ldubgd.edu.ua	0.1%
12	ir.nmu.org.ua	0.1%



Дякуємо, що перевірили
свій документ за допомогою
Plag!