

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій

"Допущений до захисту"

Завідувач кафедри ІУСТ



к.т.н, доц. Михелєв І.Л.

"22" червня 2026 р.

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття ступеня вищої освіти "бакалавр"

на тему: «Розробка веб додатку для підтримки активного та
здорового способу життя»

Виконав: студент групи 4141



Резецький В.М.
(підпис)

Керівник роботи:

К.т.н., доцент

(посада, науковий ступінь вчене звання)



Гайдаєнко О.В.
(підпис)

м. Миколаїв – 2026 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Національний університет кораблебудування імені адмірала Макарова
Навчально-науковий інститут комп'ютерних наук та управління проектами
Кафедра інформаційних управляючих систем та технологій
Спеціальність 122 "Комп'ютерні науки"
Освітня програма "Комп'ютерні науки"

"ЗАТВЕРДЖУЮ"

Гарант освітньої програми


к.т.н, доц. Гайда А.Ю.

"25" березня 2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ
на здобуття ступеня вищої освіти "бакалавр"

Студенту Резецькому Владиславу Миколайовичу

(прізвище, ім'я, по батькові)

1. Тема роботи: Розробка веб додатку для підтримки активного та
здорового способу життя

Керівник роботи к.т.н., доцент Гайдаєнко О.В.

Затверджені наказом ректора № 270-уч від "31" березня 2026 року.

2. Термін подання роботи: 22 червня 2026 р.

3. Вихідні дані до проекту (роботи) ДСТУ щодо обробки інформації,
літературні джерела, технічна документація на існуючі аналоги систем,

4. Перелік питань, що належать до розробки (найменування розділів) Аналіз
предметної області та постановка задачі. Проєктування веб додатку для
підтримки активного та здорового способу життя. Розробка веб додатку для
підтримки активного та здорового способу життя. Охорона праці.

5. Перелік презентаційних матеріалів Актуальність, концепція системи,
Загальна схема роботи, Порівняльний аналіз додатку, Діаграма Use Case ER-
діаграми інформаційної бази даних, Діаграма розгортання, Екранні форми
розробки.Висновки

6. Консультанти розділів роботи

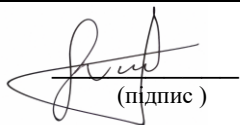
Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1	Гайдаєнко О.В.	02.02.2026	23.02.2026
2	Гайдаєнко О.В.	23.02.2026	20.03.2026
3	Гайдаєнко О.В.	20.03.2026	24.04.2026
4	Гайдаєнко О.В.	24.04.2026	01.06.2026

7. Дата видачі завдання 02 лютого 2026 р.

КАЛЕНДАРНИЙ ПЛАН

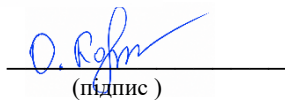
№ з/п	Назва етапів дипломного проекту (роботи)	Строк виконання етапів роботи	Примітка
1	Аналіз предметної галузі	02.02.2026	
2	Постановка задачі	16.02.2026	
3	Розробка концепції	23.02.2026	
4	Розробка проекту ІС	23.03.2026	
5	Реалізація проекту	27.04.2026	
6	Розробка питань з охорони праці	01.06.2026	
7	Розробка графічних матеріалів	05.06.2026	
8	Подання роботи на перевірку на плагіат	08.06.2026	
9	Подання роботи рецензенту	15.06.2026	
10	Подання роботи на захист	22.06.2026	

Студент


(підпис)

Резецький В.М.
(ПІБ)

Керівник роботи


(підпис)

Гайдаєнко О.В.
(ПІБ)

АНОТАЦІЯ

Дипломна робота присвячена розробці веб-застосунку для підтримання активного та здорового способу життя. У роботі розглянуто підходи до створення інформаційних веб-систем, що забезпечують користувачам зручний доступ до тематичних матеріалів, пов'язаних із фізичною активністю, харчуванням та формуванням корисних звичок.

Метою дослідження є створення веб-застосунку, який дозволяє користувачам швидко та без зайвих труднощів знаходити статті за обраною темою, отримувати структуровану інформацію та ознайомлюватися з матеріалами, що сприяють покращенню якості життя.

Об'єктом дослідження є процес організації та подання інформаційних матеріалів у веб-середовищі. Предметом дослідження є методи та засоби розробки динамічних веб-застосунків, орієнтованих на інтерактивну взаємодію з користувачем та адаптивне управління контентом.

У ході роботи було розроблено веб-систему на основі фреймворку Django, реалізовано динамічний інтернет-ресурс із можливістю гнучкого керування вмістом кожної статті, а також забезпечено структурування матеріалів за тематичними розділами. Система підтримує масштабування, розширення функціоналу та подальшу інтеграцію додаткових сервісів.

Практична цінність роботи полягає в тому, що створений веб-застосунок може бути корисним широкому колу користувачів — як новачкам, які прагнуть розпочати шлях до здорового способу життя, так і досвідченим спортсменам чи фахівцям, що шукають якісні інформаційні матеріали.

Робота складається з 4 таблиць, 26 рисунків, близько 40 сторінок, містить 6 літературних джерел та 1 додаток.

Ключові слова: веб-застосунок, здоровий спосіб життя, динамічний контент, Django, Python, MySQL, інформаційна система, користувацький інтерфейс.

ABSTRACT

The thesis is devoted to the development of a web application aimed at supporting an active and healthy lifestyle. The work examines approaches to creating informational web systems that provide users with convenient access to thematic materials related to physical activity, nutrition, and the formation of healthy habits.

The purpose of the study is to develop a web application that enables users to quickly and effortlessly find articles on selected topics, obtain structured information, and explore materials that contribute to improving their quality of life.

The object of the study is the process of organizing and presenting informational materials in a web environment. The subject of the study is the methods and tools for developing dynamic web applications focused on interactive user engagement and adaptive content management.

During the course of the work, a web system based on the Django framework was developed. A dynamic online resource was implemented, allowing flexible management of article content and providing structured organization of materials across thematic sections. The system supports scalability, functional expansion, and future integration of additional services.

The practical value of the work lies in the fact that the developed web application can be useful to a wide range of users — from beginners who wish to start leading a healthy lifestyle to experienced athletes and professionals seeking high-quality informational resources.

The thesis includes 4 tables, 26 figures, 55 pages, 6 literary sources, and 1 appendix.

Keywords: web application, healthy lifestyle, dynamic content, Django, Python, MySQL, information system, user interface.

ЗМІСТ

АНОТАЦІЯ.....	1
ABSTRACT.....	3
ВСТУП	6
РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАНЬ	8
1.1 Аналіз предметної області.....	8
1.2 Недоліки предметної області.....	11
1.3 Аналіз альтернативних рішень	13
1.4 Концепція системи	15
1.5 Математичне забезпечення системи.....	18
1.6 Постановка задачі	21
Висновки до розділу 1	23
РОЗДІЛ 2. ПРОЄКТУВАННЯ ВЕБ-ЗАСТОСУНКУ ДЛЯ ПІДТРИМКИ ЗДОРОВОГО ТА АКТИВНОГО СПОСОБУ ЖИТТЯ.....	25
2.1 Загальносистемні рішення.....	25
2.2 Архітектура системи.....	28
2.3 Діаграма послідовностей.....	31
2.4 Концептуальна модель бази даних.....	35
Висновки до розділу 2.....	40
РОЗДІЛ 3. РОЗРОБКА ВЕБ-ЗАСТОСУНКУ ДЛЯ ПІДТРИМКИ ЗДОРОВОГО ТА АКТИВНОГО СПОСОБУ ЖИТТЯ	41
3.1 Вимоги до програмного забезпечення	41

3.2 Вимоги до апаратного забезпечення	44
3.3 Діаграма розгортання	45
Висновки до розділу 3.....	48
РОЗДІЛ 4. ОХОРОНА ПРАЦІ.....	50
4.1 Умови які повинні бути забезпечені.....	50
Висновки до розділу 4.....	52
ВИСНОВКИ	53
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	55
ДОДАТОК Б - ІНСТРУКЦІЯ КОРИСТУВАЧА ВЕБ-ПОРТАЛУ	

ВСТУП

У сучасних умовах швидкого розвитку цифрових технологій особливу актуальність набувають веб-застосунки, спрямовані на підтримання здоров'я, фізичної активності та формування корисних звичок. Зростання популярності фітнес-сервісів, мобільних трекерів, онлайн-платформ з рекомендаціями щодо харчування та тренувань свідчить про підвищений інтерес суспільства до активного способу життя. Проте значна частина існуючих рішень залишається фрагментованою, орієнтованою на вузькі завдання або не забезпечує достатньої гнучкості в керуванні контентом. Це створює потребу в веб-системах, які поєднують інформативність, зручність використання та можливість адаптації під різні категорії користувачів.

Підтримання активного способу життя передбачає регулярний доступ до структурованої та достовірної інформації: рекомендацій щодо фізичних вправ, порад з харчування, статей про здоров'я, мотиваційних матеріалів та інструментів для відстеження прогресу. Ефективна організація такого контенту вимагає використання сучасних веб-технологій, здатних забезпечити швидке оновлення матеріалів, їх категоризацію, персоналізацію та зручну навігацію. Традиційні статичні сайти не можуть повною мірою задовольнити ці вимоги, оскільки не передбачають гнучкого редагування інформації та масштабування функціональності.

У зв'язку з цим виникає потреба у створенні веб-застосунку, який би поєднував інформаційний ресурс про активний спосіб життя з інструментами для динамічного керування контентом. Використання фреймворку Django дозволяє реалізувати надійну серверну частину, забезпечити структуроване зберігання даних та створити міні-CMS для адміністрування матеріалів. Такий підхід робить систему гнучкою, масштабованою та придатною для подальшого розвитку, дозволяючи оперативно оновлювати інформацію та підтримувати її актуальність.

Актуальність розробки зумовлена потребою в сучасному веб-інструменті, який забезпечує доступ до якісної інформації про здоров'я та фізичну активність, сприяє формуванню корисних звичок та допомагає користувачам орієнтуватися у великому обсязі матеріалів, пов'язаних із підтриманням активного способу життя. Створення такого застосування дозволяє мінімізувати дублювання даних, забезпечити структурованість інформації та сформувавши основу для подальшого розширення функціональних можливостей, включаючи інтеграцію персональних рекомендацій, трекерів активності та інших сервісів, спрямованих на покращення якості життя.

РОЗДІЛ 1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА ПОСТАНОВКА ЗАВДАНЬ

1.1 Аналіз предметної області

Предметна сфера даного дослідження охоплює цифрові інформаційні системи, спрямовані на підтримання здорового способу життя, популяризацію фізичної активності та формування корисних звичок. До неї належать веб-платформи, що надають користувачам структуровану інформацію про продукти харчування, спортивне обладнання, тренування, методики оздоровлення та сучасні тренди у сфері фітнесу. Особливе місце займають системи, які поєднують інформаційний контент із можливістю його динамічного оновлення через інтегровані засоби управління — міні-CMS.

В межах цієї предметної області розглядаються рішення, що дозволяють користувачам отримувати достовірні дані про користь і потенційні ризики продуктів, порівнювати різні товари чи тренажери, знайомитися з оглядами тренувальних програм та аналізувати актуальні тенденції у сфері здоров'я. Таким чином, предметна область поєднує елементи digital-wellness, контент-керованих систем та аналітичних платформ.

Сучасний інформаційний простір перенасичень суперечливими порадами щодо харчування, тренувань та здоров'я. Користувач змушений витратити значну годину на пошук достовірних джерел, порівняння характеристик продуктів чи спортивного обладнання, аналіз трендів та відсіювання рекламних матеріалів. Відсутність єдиного ресурсу, який би поєднував аналітичний підхід, структурованість та зручність використання, створює бар'єр для формування здорових звичок.

Потреба в таких системах зростає також через популяризацію домашніх тренувань, дистанційної роботи та загальне зниження фізичної активності. Люди шукають прості інструменти, які дають змогу швидко отримати інформацію про

користь продуктів, підібрати тренування, ознайомитися з оглядами спортивних товарів або дізнатися про нові тенденції у сфері фітнесу.

Саме тому виникає необхідність у веб-застосунках, які забезпечують:

- структуровану подачу контенту;
- достовірність інформації;
- можливість порівняння продуктів чи тренажерів;
- швидке оновлення матеріалів через міні-CMS;
- зручну навігацію та фільтрацію;

У сфері цифрових рішень для здорового способу життя існує ряд популярних платформ, які часто вирішують окремі аспекти цієї предметної області. Кожна з них має власну спеціалізацію, що дозволяє користувачам отримувати певний тип інформації чи функціональності. Проте жодна з них не поєднує в одному рішенні огляди продуктів, порівняння товарів, аналітичні статті, тренування, тренди та міні-CMS для модерації контенту, як це реалізовано в даному проекті.

MyFitnessPal

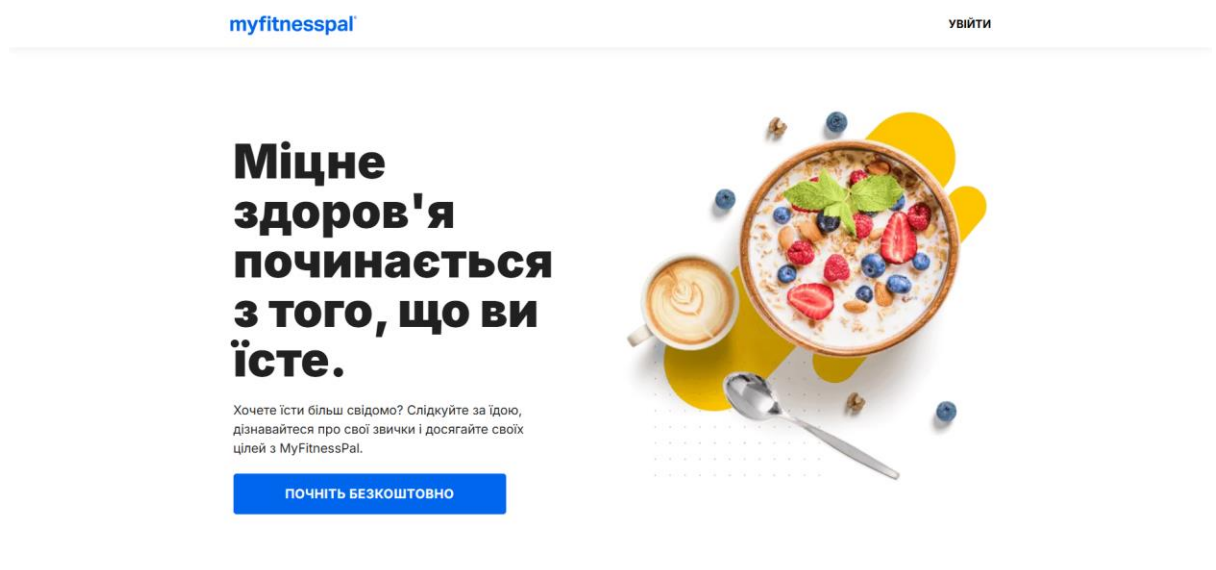


Рис. 1.1 - Головна сторінка сайту myfitnesspal

MyFitnessPal — один із найпопулярніших сервісів у сфері здорового харчування. Платформа надає інструменти для підрахунку калорій, ведення харчового щоденника та аналізу раціону. Вона містить велику базу продуктів, але не надає глибоких аналітичних оглядів, порівнянь товарів чи структурованих статей про користь/шкоду продуктів. Контент не є редакційним, а система не має CMS для модерації матеріалів.

Nike Training Club (NTC)

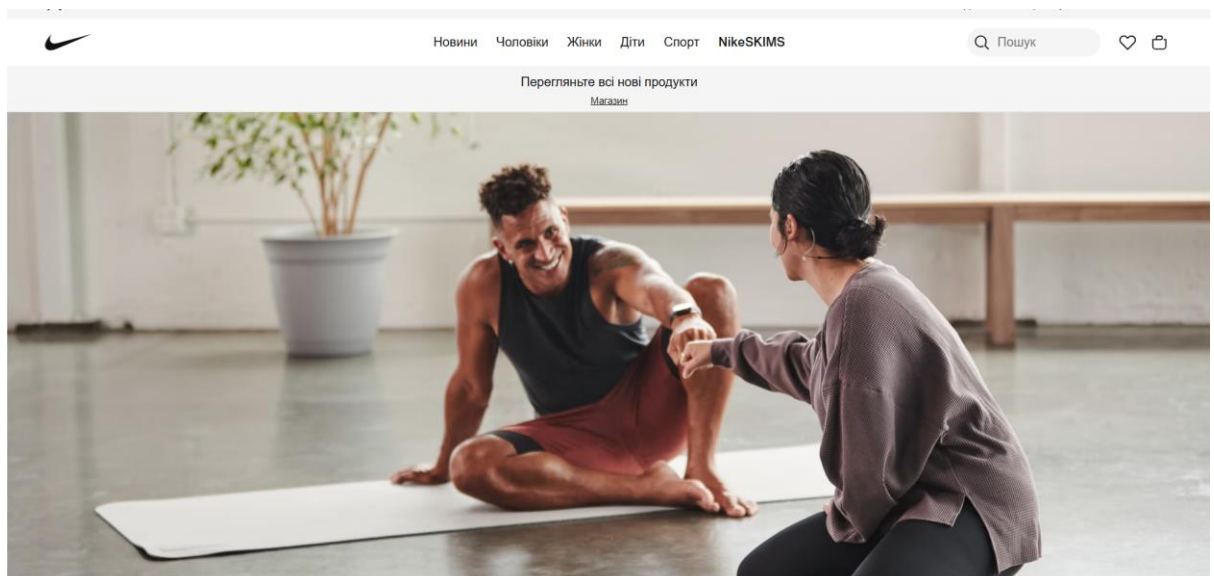


Рис. 1.2 - Головна сторінка сайту Nike Training Club

Nike Training Club спеціалізується на тренуваннях та фітнес-програмах. Платформа пропонує відеоуроки, плани тренувань та рекомендації від тренерів. Проте вона не охоплює інформацію про продукти, харчування, огляди товарів чи аналітичні статті. Контент є фіксованим і не оновлюється через CMS.

Strava

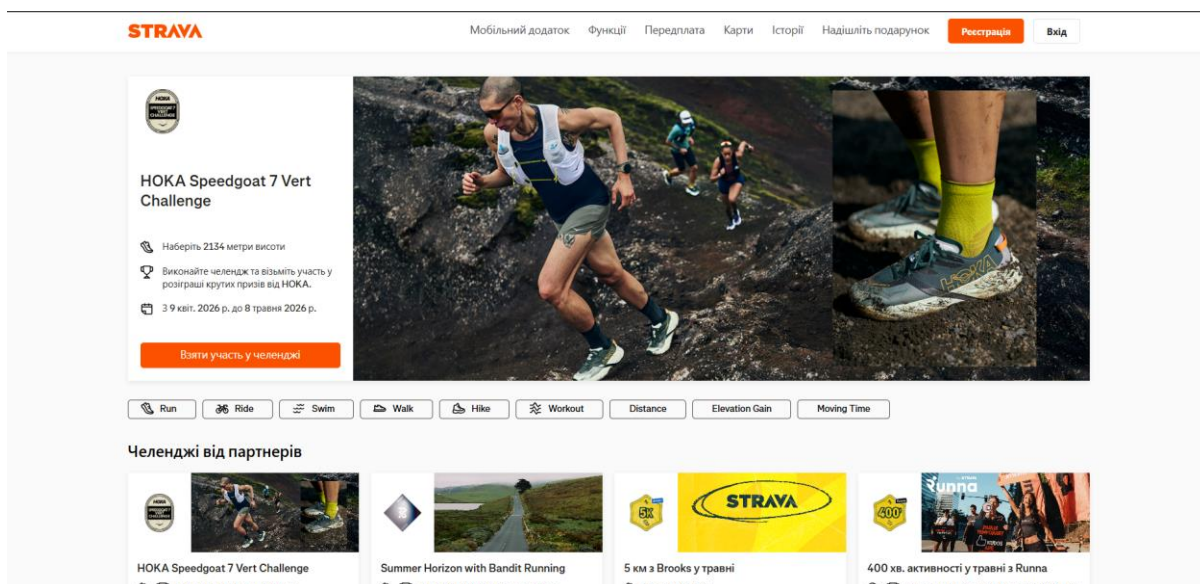


Рис. 1.3 - Головна сторінка сайту STRAVA

Strava — соціальна мережа для спортсменів, орієнтована на біг, велоспорт та інші види активності. Вона надає аналітику тренувань, маршрути та статистику. Проте Strava не працює з оглядами продуктів, трендів чи порівнянь товарів. Це трекер активності, а не інформаційний портал.

Вирішені задачі	Невирішені задачі
Трекінг харчування (MyFitnessPal)	Відсутність єдиної платформи, що поєднує продукти + тренування + тренди
Готові фітнес-програми (Nike Training Club)	Немає системи порівняння товарів/тренажерів
Аналітика активності (Strava)	Немає структурованих карток продуктів з користю/шкодою
Статті про здоров'я	Відсутність міні-CMS для модерації контенту
Огляди тренажерів у блогах	Контент неструктурований, немає фільтрів
Бази даних продуктів (калорійність)	Немає аналітичних оглядів продуктів у форматі CMS

Відеоуроки тренувань	Немає інтегрованої системи трендів та аналітики ЗСЖ
----------------------	---

Таблиця 1.4 - Порівняльний аналіз аналогічних веб-ресурсів

1.2 Недоліки предметної області

Незважаючи на значне розвиток цифрових рішень у сфері здорового способу життя, предметна область залишається фрагментованою та нерівномірно охопленою. Існуючі платформи зазвичай зосереджуються на окремих аспектах — харчуванні, тренуваннях, трекінгу активності або публікації статей — що створює інформаційні розриви та ускладнює формування цілісного уявлення про здоровий спосіб життя. Відсутність комплексних рішень призводить до того, що користувач змушений звертатися до кількох різних ресурсів, витрачаючи годину на пошук, порівняння та перевірку достовірності інформації.

Одним із ключових недоліків є нестача структурованих та систематизованих даних. Більшість платформ не пропонують уніфікованих карток продуктів чи тренажерів, де були б зібрані користь, потенційні ризики, рекомендації щодо використання та можливість порівняння з іншими товарами. Інформація часто подається у вигляді статей або відео, що ускладнює її аналіз і не дозволяє швидко отримати ключові факти.

Ще однією проблемою є відсутність інструментів порівняння. Навіть популярні фітнес-додатки та інформаційні портали не надають можливості зіставляти продукти, тренажери чи методики тренувань за визначеними параметрами. Це обмежує користувача у прийнятті обґрунтованих рішень, особливо коли йдеться про вибір спортивного інвентарю або продуктів харчування.

Суттєвим недоліком є також нестача редакційного контролю та модерації контенту. Блоги, YouTube-канали та соціальні мережі містять величезну кількість матеріалів, але їх якість, достовірність та наукова обґрунтованість часто залишаються під питанням. Відсутність міні-CMS або централізованої системи управління контентом призводить до хаотичності та швидкої втрати актуальності інформації.

Окремо варто відзначити обмеженість інтегрованих рішень. Існуючі платформи рідко поєднують у собі:

- огляди продуктів;
- Тренування;
- Тренди;
- аналітичні статті;
- порівняння товарів;
- структуровані картки;
- динамічне оновлення контенту;

Таким чином, основні недоліки предметної області полягають у фрагментованості інформації, відсутності інструментів порівняння, нестачі структурованих даних, слабкій модерації контенту та відсутності комплексних рішень, здатних охопити всі ключові аспекти здорового способу життя. Саме ці проблеми визначають актуальність розробки нового веб-застосунку, що пропонує цілісний, систематизований та керований підхід до подачі інформації.

1.3 Аналіз альтернативних рішень

У сфері здорового способу життя існує безліч популярних платформ, які пропонують користувачам інформацію про харчування, тренування, спортивні

товари та загальні рекомендації щодо фітнесу. Проте більшість із них працюють за комерційною моделлю, мають обмежений доступ до контенту або зосереджуються лише на окремих аспектах здоров'я. Нижче наведено огляд найвідоміших сервісів, які можна вважати альтернативами, але які не забезпечують комплексного підходу, реалізованого в даному проєкті.

MyFitnessPal — один із найвідоміших сервісів у сфері харчування та контролю калорій. Він пропонує велику базу продуктів, автоматичний підрахунок калорій, рекомендації щодо раціону та інтеграцію з фітнес-трекерами.

Однак більшість корисних функцій, включаючи детальну аналітику, доступні лише за платною підпискою Premium. Крім того, сервіс не надає оглядів продуктів, не порівнює товари між собою та не пропонує структурованих статей про користь чи шкоду продуктів.

Nike Training Club спеціалізується на тренуваннях і пропонує якісні відеоуроки, програми для різних рівнів підготовки та рекомендації від професійних тренерів.

Проте платформа не містить інформації про продукти, спортивні товари чи харчування. Частина тренувальних програм доступна лише після реєстрації або за підпискою, а контент не оновлюється через CMS, що робить його статичним.

Strava - це соціальна мережа для спортсменів, яка надає аналітику тренувань, маршрути та статистику. Вона чудово підходить для трекінгу активності, але не містить оглядів продуктів, тренажерів, трендів чи аналітичних статей.

Багато функцій Strava також доступні лише у платній версії Strava Premium.

Healthline, VeryWellFit, WebMD - Ці портали пропонують редакційні статті про здоров'я, харчування та фітнес. Вони мають велику базу матеріалів, але: не порівнюють товари, не мають структурованих карток продуктів, не пропонують тренувальних програм та інше.

Порівняння з розроблюваним рішенням.

На відміну від перелічених платформ, розроблений веб-застосунок має низку принципових переваг:

- Цілком безкоштовний доступ - жодних підписок, преміум-функцій чи платних розділів;
- Відсутність реклами та маркетингових маніпуляцій - контент не містить прихованих рекламних інтеграцій, що часто зустрічається на YouTube та в блогах;
- Структуровані картки продуктів - користь, шкода, рекомендації, склад, з чим поєднується - все в одному місці;
- Порівняння товарів та тренажерів - функція, якої немає у більшості популярних платформ;
- Огляд тренувань, трендів та досліджень;
- Міні-CMS для модерації контенту – адміністратор може швидко оновлювати інформацію, додавати нові статті, тренди, огляди;

1.4 Концепція системи

Концепція розроблюваної системи ґрунтується на створенні інформаційного веб-порталу, який поєднує огляди продуктів, аналітичні матеріали, тренди у сфері здорового способу життя та огляди тренувань в єдиній інтерактивній середовищі. Основна ідея полягає у формуванні доступної, структурованої та достовірної бази знань, що дозволяє користувачам швидко отримувати потрібну інформацію без рекламних впливів, платних підписок чи маркетингових обмежень.

На відміну від більшості існуючих платформ, які зосереджуються на окремих аспектах харчуванні, тренуваннях, трекінгу активності чи публікації статей запропонована система поєднує ці елементи в єдиному рішенні.

Для наочності відмінності між існуючими рішеннями та розроблюваною системою наведено порівняльну таблицю:

Критерій	Було	Стало	Планується
Структура інформації	Інформація розпорошена між різними сайтами, блогами, соцмережами	Єдина структурована база знань у форматі Topic та Review	Персоналізовані добірки та рекомендації
Достовірність контенту	Часто неперевірені джерела, рекламні матеріали	Контент формується вручну через міні-CMS, без реклами	Автоматична перевірка джерел, модерація
Зручність пошуку	Пошук на сторонніх ресурсах неточний або відсутній	Пошук за title, фільтрація за категоріями та типами	Розширений пошук за тегами, параметрами, ключовими фактами
Порівняння товарів	Порівняння потрібно робити вручну	Є механізм порівняння за спільними характеристиками	Інтерактивні графіки, розширені параметри
Оновлення інформації	Дані швидко застарівають, немає	Оновлення через Django Admin без зміни коду	Автоматичне оновлення через API

	централізованого оновлення		
Інтерфейс	Різні формати, відсутність єдиного стилю	Адаптивний, уніфікований інтерфейс	Персоналізація інтерфейсу під користувача
Взаємодія користувача	Відсутня або мінімальна	Зручна навігація, фільтри, пошук	Коментарі, лайки, спільнота

Таблиця 1.5 – Порівняння стану до впровадження системи та після її розробки

Портал складається з двох ключових інформаційних блоків:

- Торіс — структуровані картки продуктів, спортивних товарів або інгредієнтів із детальним описом користі, потенційної шкоди, рекомендацій та додаткових фактів.
- Review — аналітичні матеріали, порівняння товарів, огляди трендів, дослідження та тематичні статті, що дозволяють користувачеві отримати глибше розуміння певної теми.

У першій версії системи реалізовано повний функціонал, необхідний для роботи інформаційного порталу:

- динамічне керування контентом через міні-CMS на основі Django Admin;
- створення та редагування карток продуктів;
- публікація аналітичних статей, трендів та оглядів;
- система категоризації та фільтрації матеріалів;
- адаптивний інтерфейс для зручного перегляду контенту;
- структурована база даних MySQL з підтримкою зв'язків між сутностями;
- можливість швидкого оновлення інформації без втручання у програмний код;

На даному етапі система повністю відповідає вимогам до першої версії продукту. Вона забезпечує стабільну роботу, зручну навігацію та можливість централізованого керування контентом. Такий підхід дозволяє зосередитися на якості інформації та її структурованості, що є ключовим для інформаційних порталів про здоровий спосіб життя.

Разом із тим концепція системи передбачає подальше розвиток функціональності. У майбутніх версіях планується:

- додавання особистого кабінету користувача, що дозволить зберігати улюблені статті, формувати власні добірки та відстежувати прогрес;
- впровадження системи коментарів, яка забезпечити взаємодію між користувачами та створити елемент спільноти;
- розширення можливостей порівняння товарів, включаючи додаткові параметри та інтерактивні графіки;
- інтеграція рекомендаційних алгоритмів, що пропонують контент на основі інтересів користувача;
- створення розділу з персональними тренувальними планами;
- оптимізація міні-CMS для більш гнучкого керування структурою сторінок;

Таким чином, концепція системи передбачає поетапне розвиток: від інформаційного порталу з базовою міні-CMS до повноцінної інтерактивної платформи, яка поєднує аналітику, персоналізацію та зручні інструменти для підтримання активного способу життя.

1.5 Математичне забезпечення системи

Математичне забезпечення розроблюваної системи охоплює сукупність алгоритмів, логічних моделей та структур даних, які забезпечують обробку,

систематизацію та подання інформації у веб-застосунку. Оскільки система є інформаційним порталом, математичні методи зосереджені не на числових обчисленнях, а на логічних операціях, що визначають структуру контенту, правила його відбору та механізми взаємодії між елементами даних.

Основою математичного забезпечення є модель категоризації контенту, яка передбачає поділ інформації на два ключові блоки: Topic (картки продуктів) та Review (аналітичні матеріали, тренди, порівняння). Кожен блок має власну структуру даних, що описується через реляційні зв'язки типу «один-до-багатьох» та «багато-до-одного». Така модель дозволяє ефективно організувати інформацію та забезпечити швидкий доступ до неї.

Для формального опису структури контенту введемо множини категорій та матеріалів, а також відображення між ними:

- $C = \{ c_1, c_2, \dots, c_n \}$
- $T = \{ t_1, t_2, \dots, t_m \}$
- $F : T \rightarrow C$

Алгоритми роботи з контентом

Для роботи з контентом використовуються такі алгоритми:

Алгоритм фільтрації контенту можна формалізувати через логічну кон'юнкцію умов, що визначають критерії вибірки: $Q = \bigwedge_{i=1}^k (a_i = v_i)$

1. Алгоритми фільтрації:

- побудова динамічного WHERE-виразу;
- комбінування умов через оператори AND та OR;

У випадку комбінування кількох умов фільтрації використовується диз'юнкція або кон'юнкція логічних виразів:

$$- Q = C1 \vee C2 \vee \dots \vee Cn$$

або (якщо AND):

$$- Q = C1 \wedge C2 \wedge \dots \wedge Cn$$

Приклад SQL-запиту, фільтрація за категорією та типом контенту:

```
SELECT * FROM topics
WHERE category_id = X AND content_type = Y;
```

2. Алгоритми сортування:

- датою публікації;
- Популярністю;
- часом читання;
- типом огляду;

Timsort - це алгоритм який використовується Python/Django для сортування даних після отримання з бази, або ж MySQL-вбудовані алгоритми сортування

3. Алгоритми пошуку:

- використовується оператор LIKE або ILIKE;
- пошук здійснюється по полю title;

Пошук реалізовано за принципом часткового співпадіння рядків $S = \{t \in T \mid \text{keyword} \subseteq t.\text{title}\}$

```
SELECT * FROM topics
WHERE title LIKE '%keyword%';
```

Такий запит дозволяє знаходити матеріали навіть при неповному введенні слова.

4. Алгоритм порівняння:

- Визначаються спільні параметри (перетин множин) $P \cap B$;
- Для кожного параметра виконується вибірка значень з бази;

- Дані структуруються у форматі “параметр → значення для об’єкта 1 → значення для об’єкта 2”;

Таким чином, це дозволяє порівнювати об’єкти навіть із різних категорій, якщо вони мають спільні атрибути.

Реляційна база даних як математична основа

Реляційна база даних MySQL забезпечує математичну основу зберігання інформації. Структура таблиць побудована таким чином, щоб мінімізувати дублювання даних та забезпечити цілісність інформації.

Для цього використовуються:

- первинні ключі — унікальна ідентифікація записів;
- зовнішні ключі — забезпечення зв’язків між сутностями;
- індекси — прискорення пошуку та фільтрації;
- нормалізація — усунення надлишковості та дублювання.

1.6 Постановка задачі

На основі проведеного аналізу предметної області, виявлених недоліків та оцінки існуючих альтернатив сформульовано комплекс завдань, вирішення яких є необхідним для створення ефективного веб-застосунку, що забезпечує структуровану, достовірну та доступну інформацію про здоровий спосіб життя, продукти, тренування та сучасні тренди. Постановка завдання визначає логічну послідовність дій, які необхідно виконати для усунення інформаційної фрагментованості та забезпечення користувача якісним контентом без платних підписок, реклами та маркетингових обмежень.

Для досягнення поставленої мети необхідно вирішити такі основні завдання:

1. Аналіз та формалізація вимог до системи :
 - визначити типи контенту, які мають бути представлені на порталі (продукти, тренування, тренди, аналітичні статті);
 - сформуванати структуру подання інформації у вигляді карток, оглядів та порівнянь;
 - визначити ролі користувачів та їх можливості (гостьовий доступ, адміністратор);
2. Розробка концептуальної моделі даних:
 - визначити сутності системи та їх атрибути;
 - побудувати реляційні зв'язки між сутностями;
 - забезпечити можливість масштабування бази даних у майбутньому;
3. Проектування архітектури веб-застосунку:
 - обрати архітектурний підхід (MVT у Django);
 - визначити логіку обробки запитів, формування сторінок та взаємодії з базою даних;
 - забезпечити розмежування між інтерфейсом, логікою та даними;
4. Реалізація міні-CMS для управління контентом :
 - створити інструменти для додавання, редагування та видалення матеріалів;
 - забезпечити можливість динамічного оновлення інформації без зміни програмного коду;
 - реалізувати механізми модерації та структурування контенту.
5. Розробка інтерфейсу користувача:

- створити зручну навігацію між розділами Topic та Review;
 - забезпечити адаптивність інтерфейсу для різних пристроїв;
 - реалізувати механізми фільтрації, пошуку та сортування матеріалів.
6. Реалізація функцій порівняння та аналітики :
- створити алгоритми порівняння продуктів чи тренажерів за ключовими параметрами;
 - забезпечити відображення ключових фактів (key facts) у структурованому вигляді;
 - реалізувати механізми формування аналітичних оглядів.
7. Забезпечення стабільності та продуктивності системи:
- оптимізувати роботу з базою даних MySQL;
 - забезпечити швидке завантаження сторінок;
 - реалізувати базові механізми безпеки (валідація, захист від атак).
8. Підготовка основи для подальшого розвитку системи:
- передбачити можливість додавання особистого кабінету користувача;
 - закласти основу для системи коментарів;
 - забезпечити можливість інтеграції рекомендаційних алгоритмів;
 - створити умови для розширення функціоналу без зміни базової архітектури;

Висновки до розділу 1

У першому розділі було проведено комплексний аналіз предметної галузі, що охоплює сучасні цифрові рішення у сфері здорового способу життя, харчування, тренувань та аналітичного контенту. Встановлено, що при значній кількості популярних платформ, інформаційний простір залишається фрагментованим, а користувачі змушені звертатися до різних джерел для отримання повної та достовірної інформації. Існуючі сервіси переважно орієнтовані на окремі аспекти — трекінг харчування, тренування, аналітику активності або публікацію статей — і не забезпечують комплексного підходу.

У ході аналізу були визначені ключові недоліки предметної області: відсутність структурованих карток продуктів, нестача інструментів порівняння товарів та тренажерів, слабка модерація контенту, наявність платних підписок та рекламних обмежень, а також відсутність єдиної платформи, яка б поєднувала огляди, тренди, тренування та аналітичні матеріали в одній середі. Розгляд існуючих альтернатив показав, що жодний із популярних сервісів не пропонує безкоштовного, комплексного та структурованого рішення без маркетингових впливів.

На основі виявлених проблем сформовано концепцію системи, яка передбачає створення інформаційного веб-порталу з двома ключовими блоками — *Topic* та *Review* — та інтегрованою міні-CMS для динамічного управління контентом. У першій версії реалізовано всі необхідні функції для повноцінної роботи порталу, а також визначено можливості подальшого розвитку системи, включаючи особистий кабінет, коментарі та розширені інструменти аналітики.

Постановка завдання окреслила основні етапи розробки, необхідні для створення ефективного веб-застосунку: аналіз вимог, проектування структури даних, розробка архітектури, реалізація міні-CMS, створення інтерфейсу, впровадження алгоритмів порівняння та забезпечення продуктивності системи.

Отримані результати підтверджують актуальність проекту та обґрунтовують необхідність створення комплексної платформи для підтримання активного способу життя.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ВЕБ-ЗАСТОСУНКУ ДЛЯ ПІДТРИМКИ ЗДОРОВОГО ТА АКТИВНОГО СПОСОБУ ЖИТТЯ

2.1 Загальносистемні рішення

У цьому підпункті описуються загальні принципи функціонування системи, ролі користувачів та їх взаємодія з основними можливостями веб-порталу. Метою є формалізація поведінки системи на високому рівні та визначення ключових сценаріїв використання, що забезпечують коректну роботу інформаційного ресурсу.

Розроблюваний веб-портал має дві основні ролі: Користувач та Адміністратор. Користувач отримує доступ до інформаційних матеріалів, включаючи картки продуктів (Topic), аналітичні огляди (Review), тренди, порівняння та результати пошуку. Адміністратор, у свою чергу, має розширені можливості, що включають створення, редагування та модерацію контенту через міні-CMS, а також підтримку структури порталу.

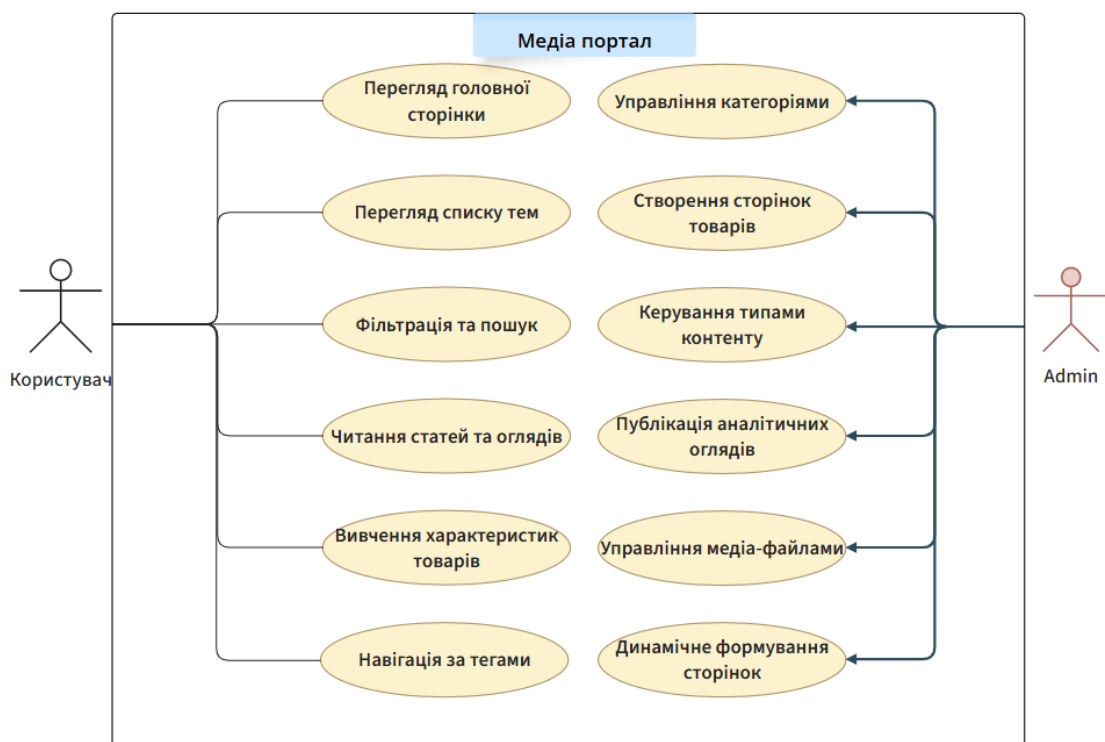


Рис. 2.1 – Діаграма варіантів використання системи

На діаграмі зображено основні ролі та їх взаємодію з функціональними можливостями порталу. Користувач може переглядати Topic, Review, виконувати пошук, фільтрацію та порівняння товарів. Адміністратор має доступ до керування контентом, включаючи додавання та редагування матеріалів. Це забезпечує чітке розмежування прав доступу та логічну структуру взаємодії із системою.

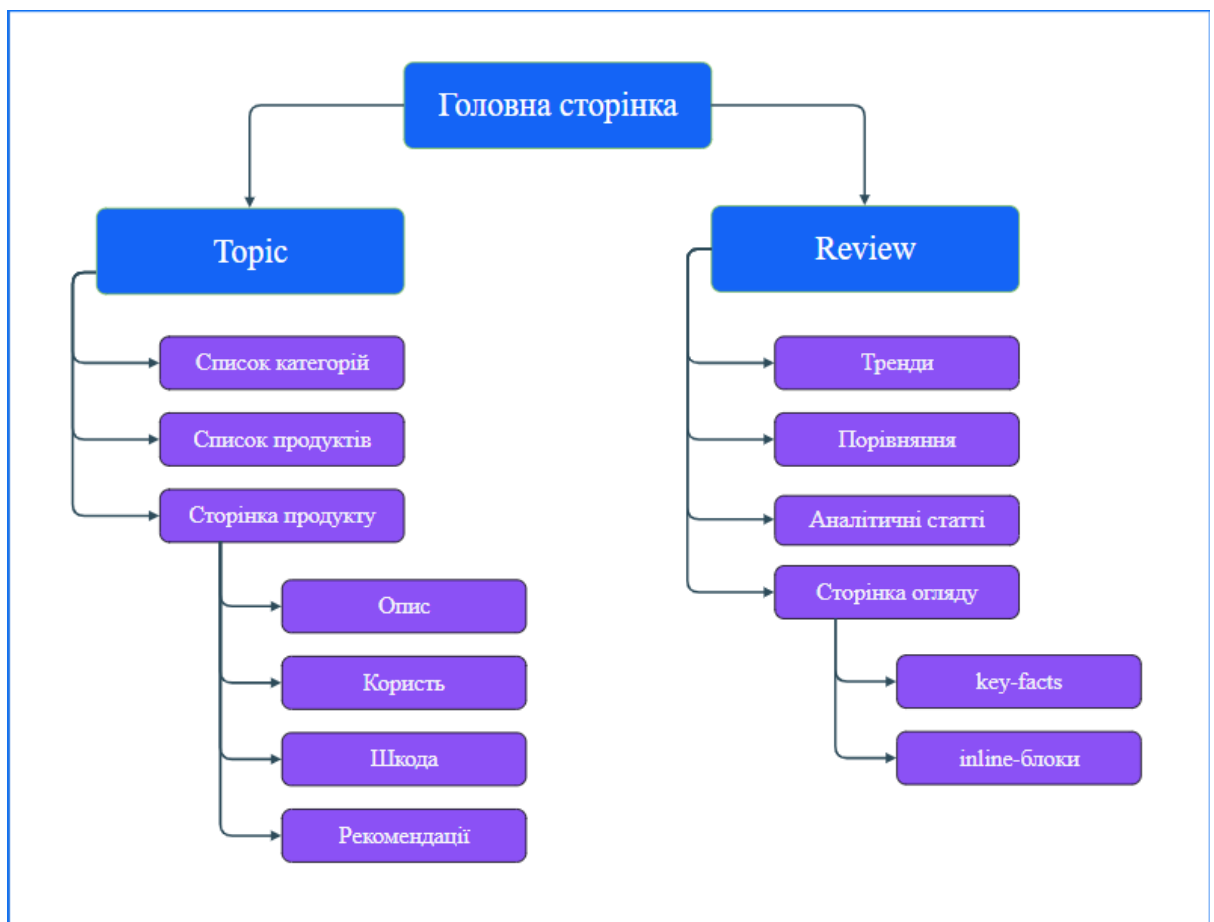


Рис. 2.2 – Структурна схема навігації веб-порталу

Схема відображає логічну організацію сторінок порталу. Головна сторінка є центральною точкою входу, з якої користувач може перейти до розділів Topic та Review. Кожен із цих розділів містить підрозділи: картки продуктів, тренди, аналітичні статті, порівняння та інші інформаційні блоки. Така структура забезпечує інтуїтивну навігацію та швидкий доступ до необхідної інформації.

Взаємодія користувача з порталом відбувається через стандартні сценарії:

- відкриття сторінки Topic або Review;
- виконання пошуку за ключовими словами;
- застосування фільтрів;
- перегляд порівнянь;
- навігація між розділами;
- отримання структурованого контенту у вигляді карток або статей.

Ці сценарії визначають загальну поведінку системи та формують основу для подальшого проектування архітектури та моделі даних.

2.2 Архітектура системи

Архітектура веб-порталу побудована на основі фреймворку Django, який реалізує архітектурний підхід MVT (Model–View–Template). Така структура забезпечує чітке розмежування між логікою обробки даних, бізнес-процесами та представленням інформації користувачу. Архітектура системи визначає, яким чином відбувається обробка запитів, взаємодія з базою даних, формування HTML-сторінок та повернення результатів користувачу.

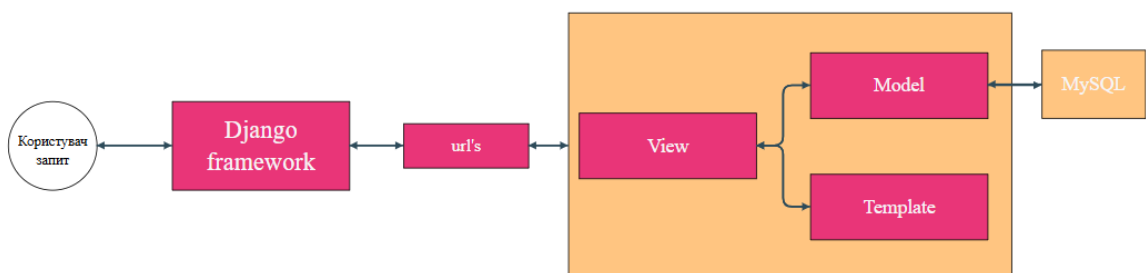


Рис. 2.3 – Архітектурна модель MVT у системі

У моделі MVT компонент Model відповідає за роботу з базою даних, включаючи отримання, збереження та оновлення інформації. Компонент View обробляє HTTP-запити, викликає необхідні моделі та передає дані у шаблони. Компонент Template відповідає за формування кінцевої HTML-сторінки, яку отримує користувач. Такий підхід забезпечує масштабованість, гнучкість та простоту підтримки системи.

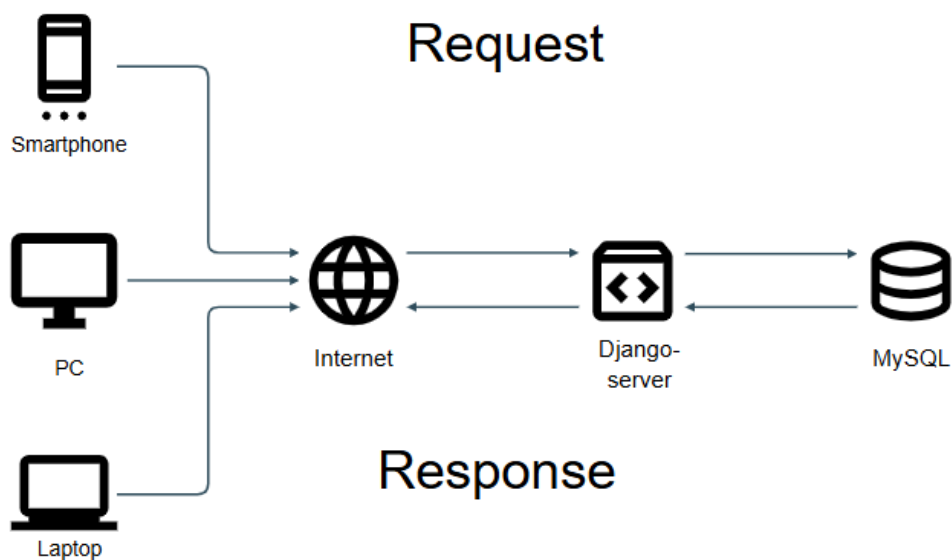


Рис. 2.4 – Клієнт–серверна архітектура веб-порталу

Система працює за класичною клієнт–серверною моделлю. Користувач взаємодіє з порталом через веб-браузер, який надсилає HTTP-запити на сервер. Серверна частина, реалізована на Django, обробляє запит, звертається до бази даних MySQL, отримує необхідні дані та формує відповідь у вигляді HTML-сторінки. Це забезпечує швидку обробку запитів та стабільну роботу системи.

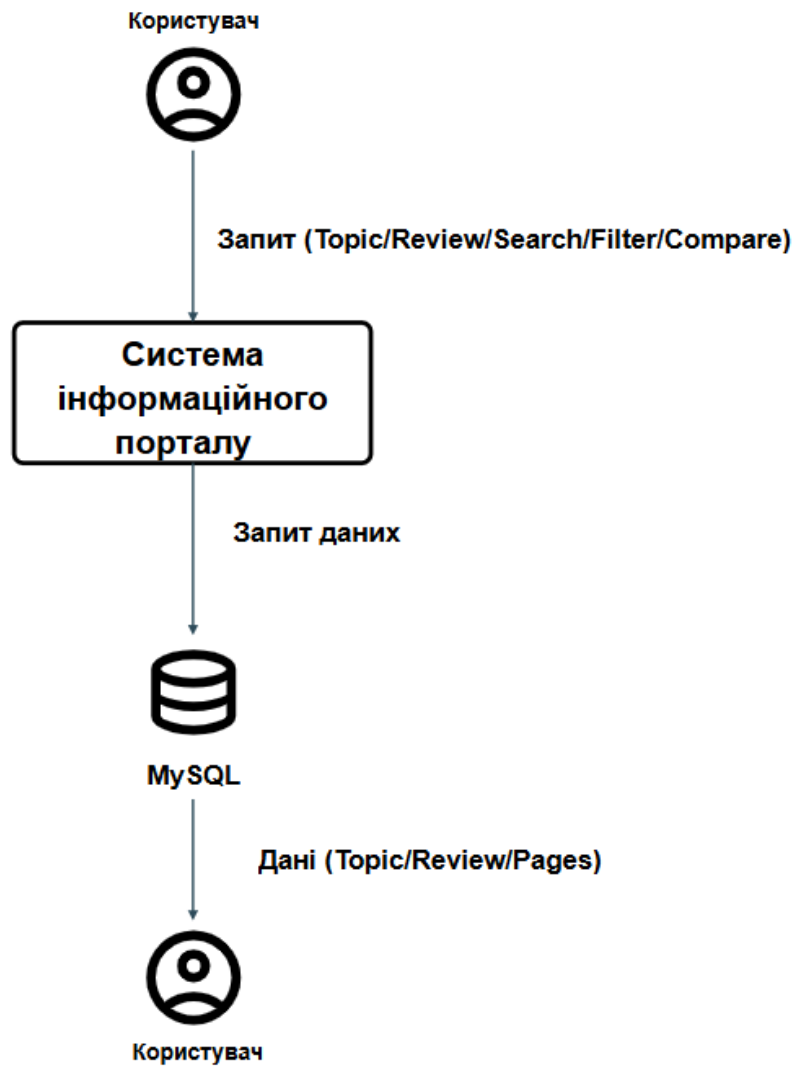


Рис. 2.5 – DFD-діаграма 0 рівня

DFD-діаграма нульового рівня відображає загальний потік даних у системі. Користувач надсилає запити на перегляд Topic, Review, пошук або фільтрацію. Система обробляє запит, звертається до бази даних та повертає користувачу структуровану інформацію. Ця діаграма демонструє загальну логіку взаємодії між користувачем, системою та базою даних.



Рис. 2.6 – DFD-діаграма 1 рівня

На діаграмі першого рівня деталізовано основні процеси системи:

- Обробка Topic
- Обробка Review
- Пошук контенту
- Фільтрація контенту
- Порівняння товарів
- Модерація контенту

Кожен процес взаємодіє з відповідними таблицями бази даних, отримує або оновлює інформацію та формує результат для користувача. DFD-діаграма дозволяє формалізувати логіку роботи системи та визначити ключові потоки даних.

2.3 Діаграма послідовностей

Діаграми послідовностей описують покрокову взаємодію між користувачем, клієнтською частиною (браузером), серверною логікою та базою даних під час виконання певних операцій у системі. Вони дозволяють наочно продемонструвати, як саме обробляються запити, які компоненти беруть участь у процесі та в якій послідовності відбувається обмін даними.

У рамках даного проєкту побудовано дві основні діаграми послідовностей:

- відкриття сторінки Toris,
- відкриття сторінки Review та виконання порівняння.

Ці діаграми відображають ключові сценарії роботи користувача з порталом.

Діаграма послідовностей для Toris

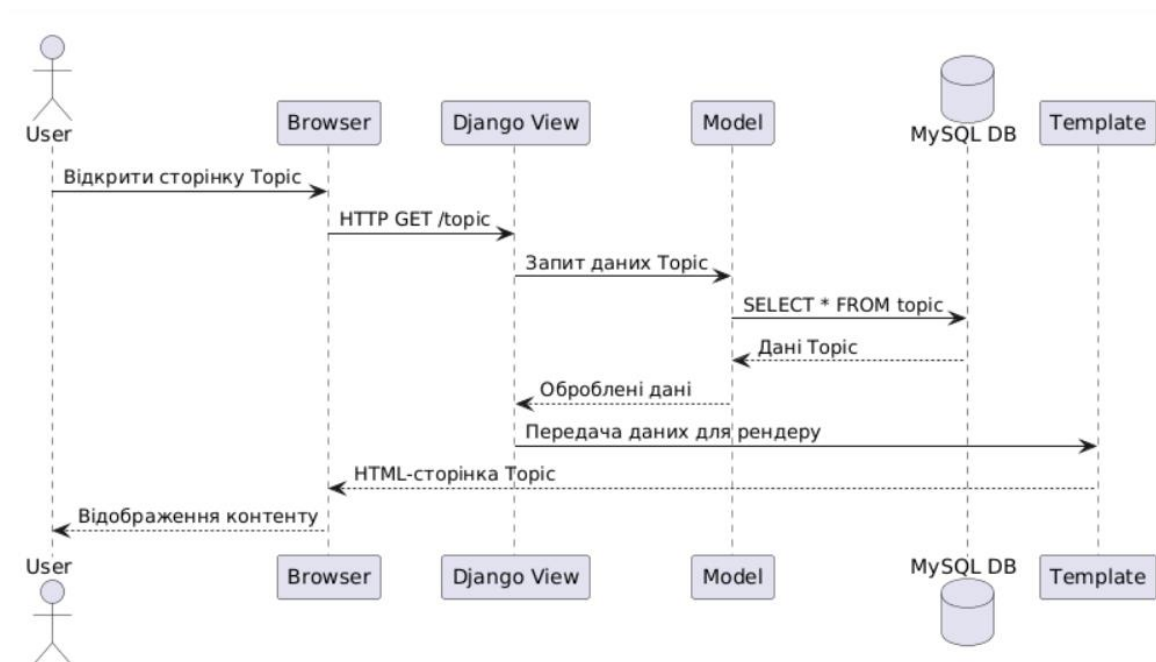


Рис. 2.7 – Діаграма послідовностей для відображення Topic

На діаграмі зображено процес отримання та відображення сторінки Topic. Користувач ініціює запит через браузер, який надсилає HTTP-запит до серверної частини. Компонент View обробляє запит, звертається до моделі та виконує вибірку даних з бази даних. Після отримання результатів View передає дані у Template, де формується HTML-сторінка. Готова сторінка повертається у браузер та відображається користувачу.

Такий підхід забезпечує чітку структурування логіки обробки запитів та дозволяє легко масштабувати систему.

Діаграма послідовностей для Review

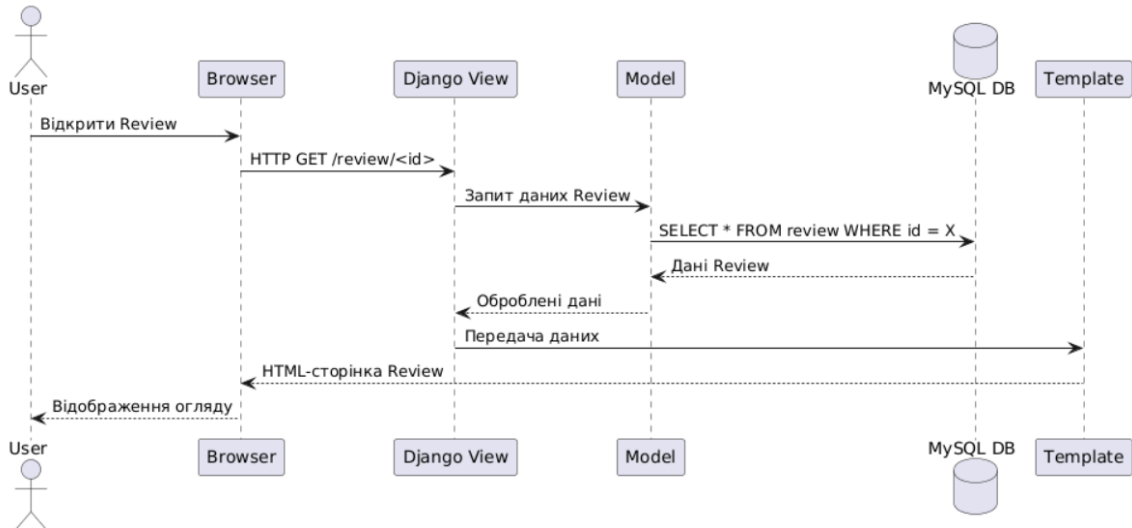


Рис. 2.8 – Діаграма послідовностей для Review та порівняння товарів

Користувач відкриває сторінку огляду, браузер надсилає запит до View, який звертається до моделі та виконує вибірку даних з таблиці Review. Після цього Template формує HTML-сторінку, яка повертається користувачу.

2.4 Концептуальна модель бази даних

Концептуальна модель бази даних визначає логічну структуру інформації, яка використовується у веб-порталі, а також взаємозв'язки між основними сутностями системи. Метою моделювання є забезпечення цілісності даних, уникнення дублювання інформації та створення оптимальної структури для подальшої обробки запитів.

База даних побудована за реляційною моделлю та реалізована у середовищі MySQL. Основними сутностями системи є: Topic, TopicPage, Review, ReviewPage, а також допоміжні сутності, такі як Category та Tag. Кожна сутність відповідає окремому типу контенту, що відображається на порталі, а зв'язки між ними забезпечують логічну цілісність та структурованість інформації.

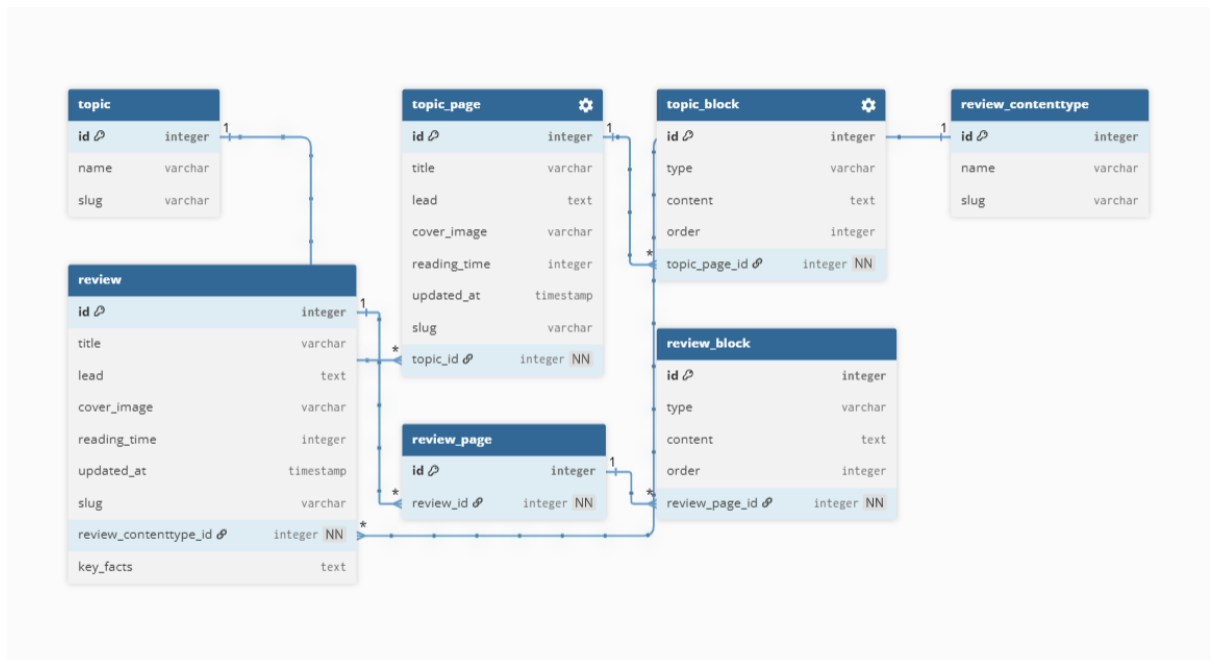


Рис. 2.9 – ER-діаграма логічної моделі бази даних

ER-діаграма відображає основні сутності та їх взаємозв'язки. Сутність *Topic* містить загальну інформацію про продукт або інгредієнт, тоді як сутність *TopicPage* зберігає структуровані блоки контенту, пов'язані з конкретним *Topic*. Аналогічно, сутність *Review* представляє аналітичний матеріал або тренд, а *ReviewPage* містить його деталізовані текстові та графічні блоки.

Зв'язки типу «один-до-багатьох» між *Topic* → *TopicPage* та *Review* → *ReviewPage* забезпечують можливість гнучкого формування контенту та його розширення.

Фізична модель описує реальну структуру таблиць у MySQL, включаючи типи даних, ключі, індекси та зв'язки між таблицями. Нижче наведено скріншоти основних таблиць, що використовуються у системі.

Таблиця *Topic*

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
id	BIGINT	☒	☒	☐	☐	☐	☐	☒	☐	
title	VARCHAR(255)	☐	☒	☐	☐	☐	☐	☐	☐	
lead	LONGTEXT	☐	☒	☐	☐	☐	☐	☐	☐	
cover_image	VARCHAR(100)	☐	☒	☐	☐	☐	☐	☐	☐	
reading_time	INT	☐	☒	☐	☐	☐	☐	☐	☐	
updated_at	DATE	☐	☒	☐	☐	☐	☐	☐	☐	
slug	VARCHAR(50)	☐	☒	☐	☐	☐	☐	☐	☐	
topic_id	BIGINT	☐	☒	☐	☐	☐	☐	☐	☐	

Рис. 2.10 – Структура таблиці Topic

Таблиця містить основні атрибути продукту: назву, короткий опис, категорію, часові мітки створення та оновлення. Поле id є первинним ключем, а зв'язок із таблицею TopicPage реалізовано через зовнішній ключ topic_id.

Таблиця TopicPage

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
id	BIGINT	☒	☒	☐	☐	☐	☐	☒	☐	
type	VARCHAR(50)	☐	☒	☐	☐	☐	☐	☐	☐	
content	JSON	☐	☒	☐	☐	☐	☐	☐	☐	
order	INT	☐	☒	☐	☐	☐	☐	☐	☐	
topic_page_id	BIGINT	☐	☒	☐	☐	☐	☐	☐	☐	

Рис. 2.11 – Структура таблиці TopicPage

Таблиця зберігає структуровані блоки контенту, що належать до конкретного Topic. Поле topic_id є зовнішнім ключем, який забезпечує зв'язок із таблицею Topic.

Таблиця Review

topic_topic - Table topic_topicpage - Table topic_topicblock - Table review_contenttype - Table review_review - Table x

<

Рис. 2.12 – Структура таблиці Review

Таблиця містить загальну інформацію про аналітичний матеріал: назву, тип (огляд, тренд, порівняння), дату створення та інші атрибути.

Таблиця ReviewBlock

topic_topicpage - Table

topic_topicblock - Table

review_contenttype - Table

review_review - Table

review_reviewpage - Table

review_reviewblock - Table

Table Name:

Schema: **diplomproject**

Charset/Collation:

Default Charset

Default Collation

Engine:

InnoDB

Comments:

Column Name	Datatype	PK	NN	UQ	B	UN	ZF	AI	G	Default/Expression
 id	BIGINT	☒	☒	☐	☐	☐	☐	☒	☐	
 type	VARCHAR(50)	☐	☒	☐	☐	☐	☐	☐	☐	
 content	JSON	☐	☒	☐	☐	☐	☐	☐	☐	
 order	INT	☐	☒	☐	☐	☐	☐	☐	☐	
 review_page_id	BIGINT	☐	☒	☐	☐	☐	☐	☐	☐	

Рис. 2.13 – Структура таблиці ReviewBlock

Таблиця містить текстові та графічні блоки, що формують повний зміст огляду. Зв'язок із Review реалізовано через поле review_id.

Таблиця Topic_Category

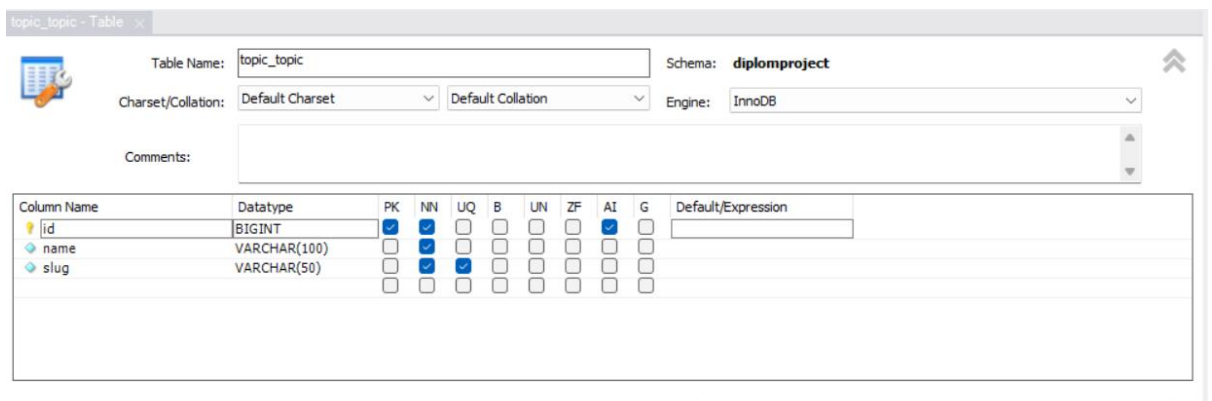


Рис. 2.14 – Структура таблиці Category

Категорії використовуються для групування Topic

Таблиця Review_contenttype

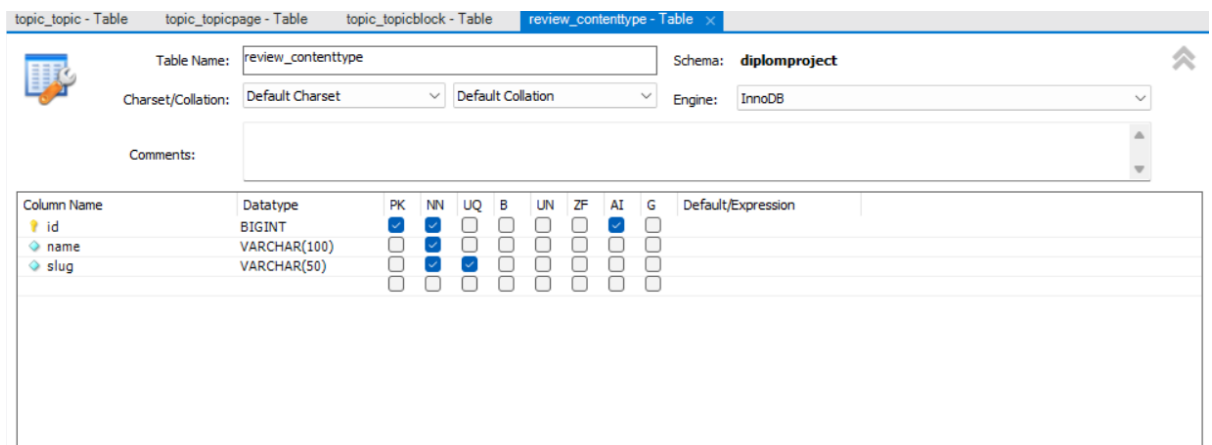


Рис. 2.15 – Структура таблиці Review_contenttype

Теги дозволяють розширити можливості пошуку та фільтрації контенту.

Таблиця	Первинний ключ (PK)	Зовнішній ключ (FK)	Тип зв'язку
Topic	id	contenttype_id → topic_contenttype(id)	1 ContentType → N Topic
TopicPage	id	topic_id → Topic(id)	1 Topic → N TopicPage

Review	id	contenttype_id → review_contenttype(id)	1 ContentType → N Review
ReviewPage	id	review_id → Review(id)	1 Review → N ReviewPage
topic_contenttype	id		Використовується у Topic
review_contenttype	id		Використовується у Review

Таблиця 2.16 – Фізична модель зв'язків між таблицями бази даних

Фізична модель бази даних відображає реальні зв'язки між таблицями у середовищі MySQL. Основними сутностями системи є Topic, TopicPage, Review та ReviewPage, які пов'язані між собою за принципом «один-до-багатьох». Це дозволяє зберігати структуровані блоки контенту для кожного Topic або Review у вигляді окремих записів у відповідних таблицях TopicPage та ReviewPage.

Для категоризації контенту використовуються дві окремі таблиці: topic_contenttype та review_contenttype. Перша відповідає за категорії у розділі Topic, друга — за категорії у розділі Review. Хоча ці сутності могли бути об'єднані в одну таблицю, їх розділення не впливає на логічну цілісність системи та дозволяє незалежно керувати типами контенту.

Усі зовнішні ключі реалізовані з використанням механізмів ON DELETE CASCADE та ON UPDATE CASCADE, що гарантує цілісність даних та автоматичне оновлення або видалення пов'язаних записів. Фізична модель забезпечує оптимальну структуру для виконання запитів, масштабування системи та підтримки високої продуктивності веб-порталу.

Висновки до розділу 2

У другому розділі було виконано комплексне проектування інформаційного веб-порталу, що включає визначення загальносистемних рішень, побудову архітектури, моделювання бази даних та опис ключових сценаріїв взаємодії між компонентами системи. На основі діаграм варіантів використання було визначено ролі користувачів та їх функціональні можливості, що забезпечило чітке розмежування доступу та логічну структуру роботи порталу.

Архітектурна модель, побудована за принципами MVT, дозволила формалізувати процес обробки запитів та взаємодію між клієнтською і серверною частинами. DFD-діаграми нульового та першого рівнів відобразили основні потоки даних у системі та деталізували ключові бізнес-процеси, що забезпечують роботу порталу.

Концептуальна та фізична моделі бази даних продемонстрували структуру зберігання інформації, взаємозв'язки між сутностями та механізми забезпечення цілісності даних. Побудовані діаграми послідовностей показали покрокову взаємодію між користувачем, сервером та базою даних під час перегляду Topic, Review.

Таким чином, результати проектування, представлені у цьому розділі, створюють цілісну основу для подальшої реалізації системи, забезпечують її логічну узгодженість, масштабованість та ефективність роботи.

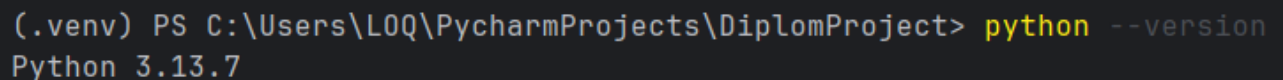
РОЗДІЛ 3. РОЗРОБКА ВЕБ-ЗАСТОСУНКУ ДЛЯ ПІДТРИМКИ ЗДОРОВОГО ТА АКТИВНОГО СПОСОБУ ЖИТТЯ

3.1 Вимоги до програмного забезпечення

Для розробки, тестування та експлуатації веб-порталу було використано сучасні інструменти та програмні засоби, які забезпечують стабільну роботу серверної частини, коректну взаємодію з базою даних та зручність у процесі розробки. Нижче наведено перелік основного програмного забезпечення, що застосовувалося під час створення системи.

Мова програмування та фреймворк:

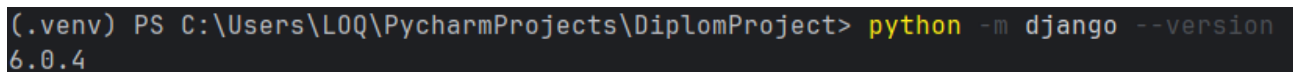
Python 3.13.7 - основна мова програмування, що забезпечує логіку роботи серверної частини.



```
(.venv) PS C:\Users\L0Q\PycharmProjects\DiplomProject> python --version
Python 3.13.7
```

На рисунку 3.1 наведено скріншот встановленої версії Python.

Django 6.0.4 - веб-фреймворк, який реалізує архітектуру MVT та забезпечує маршрутизацію, роботу з моделями, шаблонами та ORM.



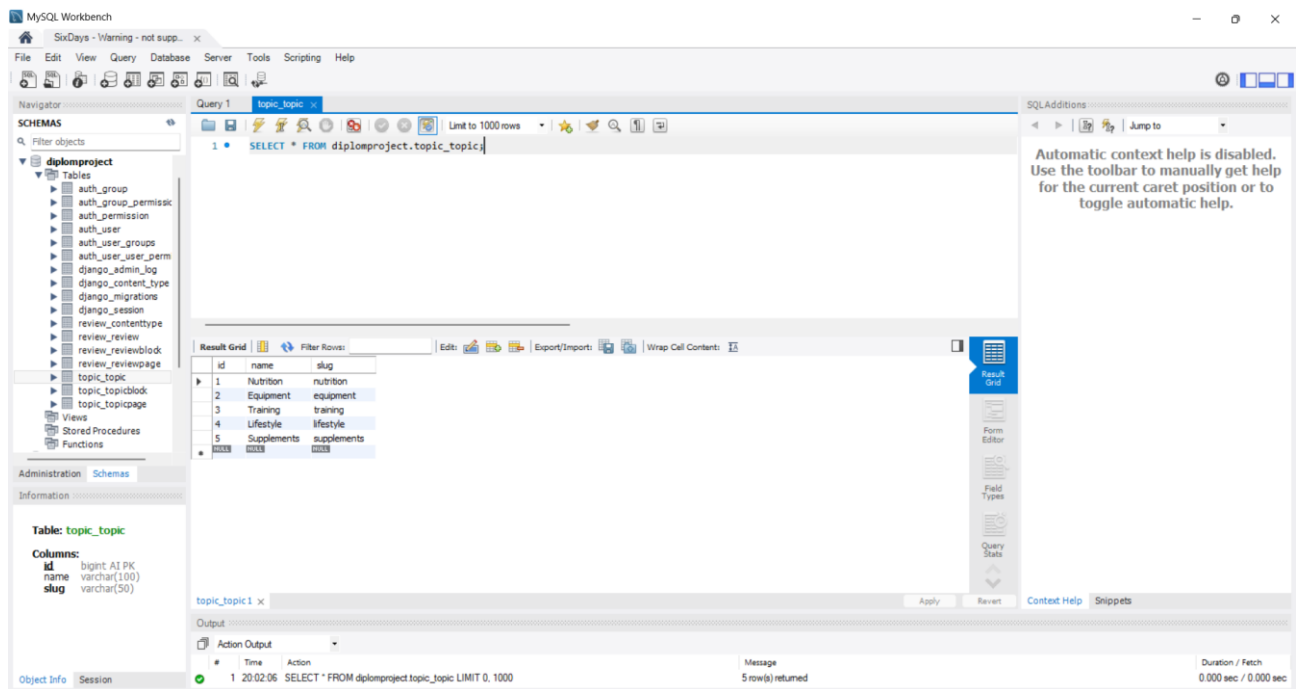
```
(.venv) PS C:\Users\L0Q\PycharmProjects\DiplomProject> python -m django --version
6.0.4
```

На рисунку 3.2 показано версію Django, встановлену у середовищі розробки.

Система керування базами даних:

MySQL Server — використовується для зберігання структурованих даних порталу.

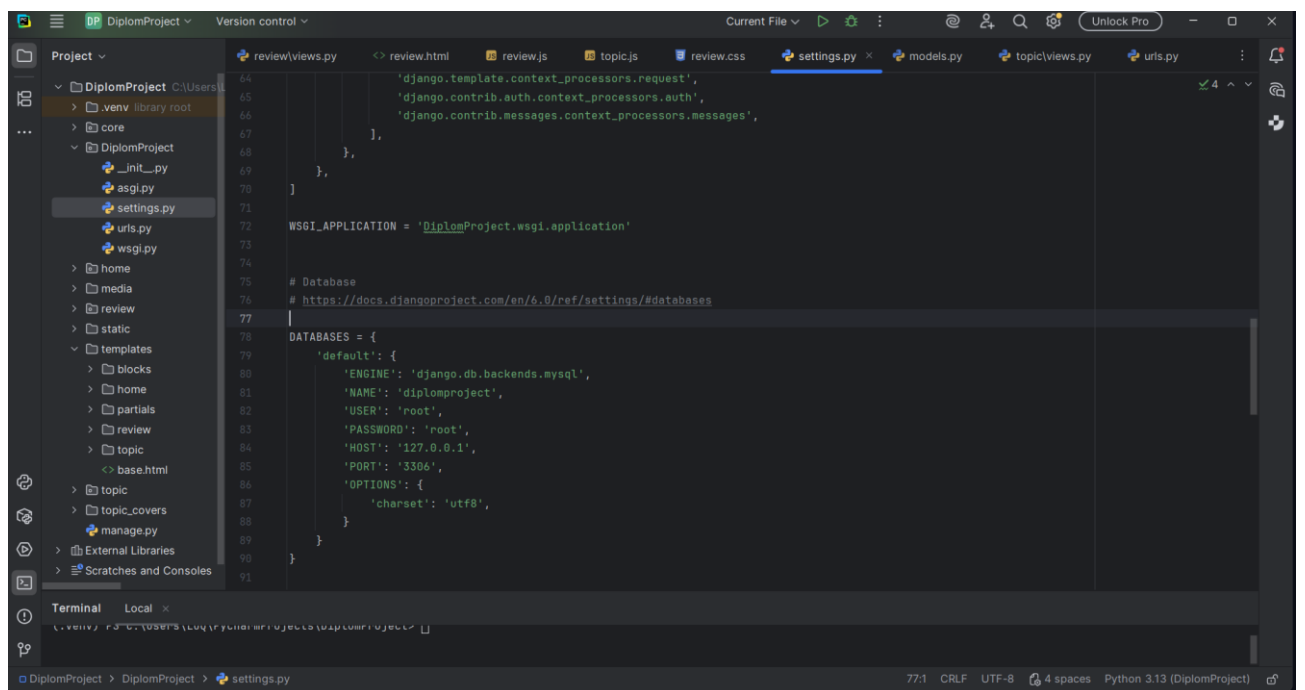
MySQL Workbench — застосовується для адміністрування бази даних, перегляду таблиць, виконання SQL-запитів та візуалізації структури БД.



На рисунку 3.3 наведено скріншот відкритої схеми бази даних із розгорнутими таблицями.

Середовище розробки

PyCharm - IDE, що забезпечує зручну роботу з файлами проєкту, підсвічування синтаксису, інтеграцію з Git та запуск локального сервера.



На рисунку 3.4 наведено структуру Django-проєкту з розгорнутими папками.

Додаткові інструменти:

- `pip` — менеджер пакетів Python.
- `Git` — система контролю версій.
- Веб-браузер (Chrome) — для тестування клієнтської частини.

Використання зазначеного програмного забезпечення забезпечує стабільність роботи системи, зручність розробки та можливість подальшого масштабування.

3.2 Вимоги до апаратного забезпечення

Для коректної роботи веб-порталу та забезпечення стабільного функціонування серверної частини необхідно, щоб комп'ютер або сервер, на якому розгортається система, відповідав певним апаратним вимогам. Оскільки система побудована на основі фреймворку Django та використовує СУБД MySQL, вимоги до обладнання залишаються помірними та не потребують спеціалізованих ресурсів.

Нижче наведено мінімальні та рекомендовані характеристики, достатні для розробки, тестування та експлуатації системи.

Параметр	Мінімальні вимоги	Рекомендовані вимоги
Процесор (CPU)	2 ядра, 1.6–2.0 GHz	4 ядра, 2.5+ GHz
Оперативна пам'ять (RAM)	4 GB	8–16 GB
Накопичувач (HDD/SSD)	2 GB для проєкту та БД	10+ GB SSD
Операційна система	Windows 10 / Linux Ubuntu 20.04	Windows 11 / Linux Ubuntu 22.04

Мережеві вимоги	Підключення до Інтернету	Стабільне підключення 10–100 Mbps
Браузер	Chrome / Firefox	Chrome

Таблиця 3.5 – Апаратні вимоги до системи

Мінімальні характеристики забезпечують можливість запуску локального сервера Django, роботи з MySQL та виконання базових операцій розробки. Рекомендовані характеристики дозволяють комфортно працювати з великими обсягами даних, швидко виконувати міграції, тестування та обробку запитів.

Система не потребує спеціалізованого обладнання та може бути розгорнута як на персональному комп'ютері розробника, так і на віддаленому сервері або хмарній платформі.

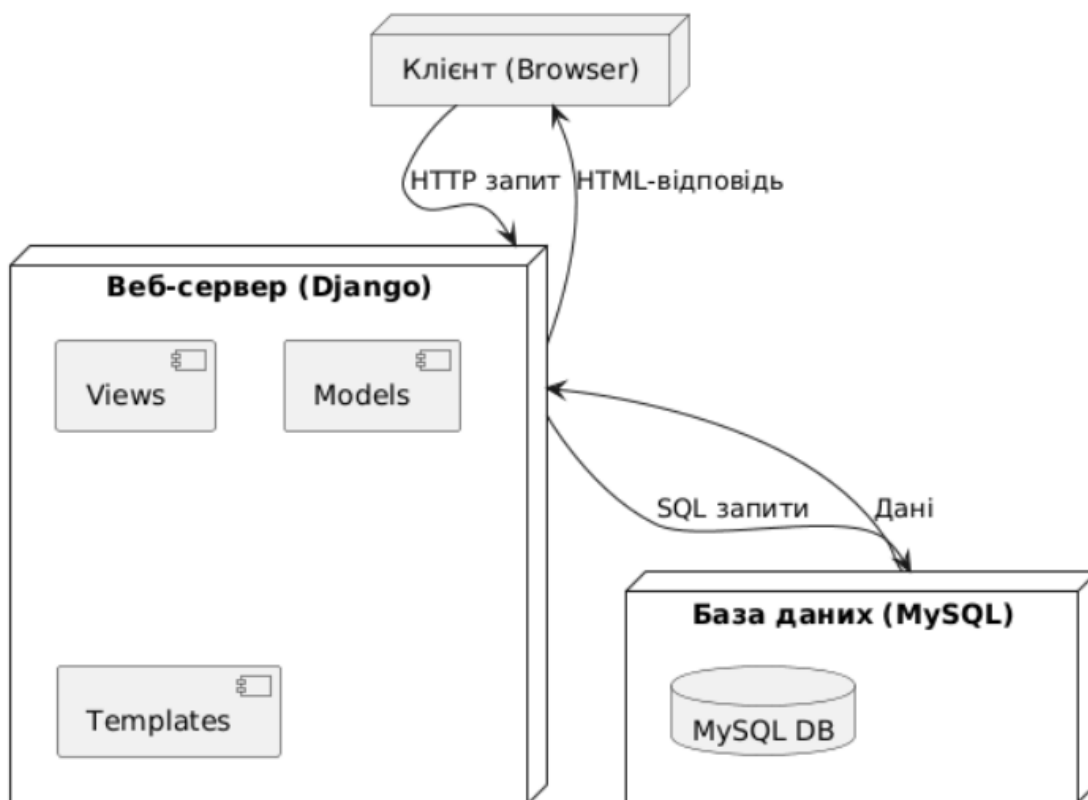
3.3 Діаграма розгортання

Діаграма розгортання відображає фізичну структуру системи та показує, на яких апаратних або програмних вузлах розміщені основні компоненти веб-порталу. Вона демонструє взаємодію між клієнтом, веб-сервером та сервером бази даних, а також описує, як саме відбувається обробка запитів у реальному середовищі.

У межах даного проєкту система складається з трьох основних вузлів:

- Клієнтський пристрій (Browser) — забезпечує доступ користувача до веб-порталу через HTTP/HTTPS.
- Веб-сервер (Django) — обробляє запити, виконує бізнес-логіку, формує HTML-сторінки та взаємодіє з базою даних.

- Сервер бази даних (MySQL) — зберігає структуровані дані, забезпечує виконання SQL-запитів та підтримує цілісність інформації.



На рисунку 3.6 наведено графічне представлення діаграми розгортання, створене у вигляді UML-схеми.

Опис діаграми:

- Клієнт - Користувач взаємодіє із системою через веб-браузер, надсилаючи HTTP-запити до веб-сервера. Браузер отримує HTML-відповідь, яка формується на основі шаблонів Django.
- Веб-сервер - Складається з таких компонентів:
 - Views — обробляють запити та формують відповіді.
 - Models — реалізують логіку роботи з даними.

- Templates — відповідають за відображення інформації на стороні клієнта.
- URL-маршрутизація — визначає, який view обробляє конкретний запит.

с) Сервер бази даних - Зберігає всі сутності системи: Topic, TopicPage, Review, ReviewPage, а також службові таблиці. Django ORM формує SQL-запити та забезпечує доступ до даних.

Структура серверної частини

Для підтвердження реалізації серверної логіки на рисунку 3.7 наведено структуру Django-проєкту, що включає:

- головний конфігураційний модуль (settings.py, urls.py)
- додаток із бізнес-логікою (models.py, views.py)
- шаблони інтерфейсу (templates/)
- статичні ресурси (static/)
- міграції (migrations/)

Ця структура відповідає стандартам Django та забезпечує модульність, масштабованість і зручність підтримки системи.

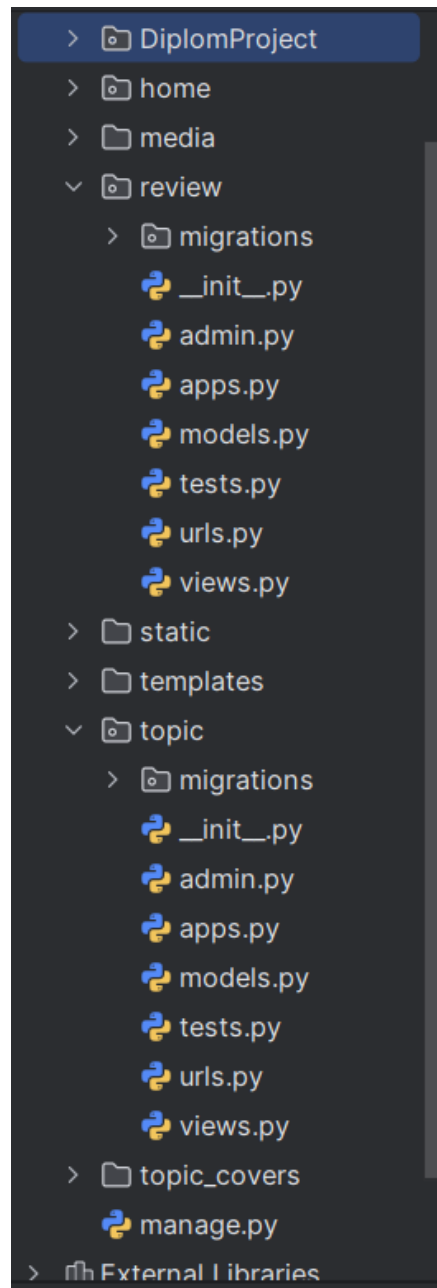


Рис. 3.7 - Структура Django-проекту

Висновки до розділу 3

У даному розділі було проведено детальний аналіз програмного та апаратного забезпечення, необхідного для розробки та функціонування веб-порталу. Визначено основні інструменти, середовище розробки та технології, що

забезпечують стабільну роботу системи, а також наведено мінімальні та рекомендовані апаратні характеристики для її розгортання.

Було побудовано діаграму розгортання, яка відображає фізичну структуру системи та взаємодію між клієнтською частиною, веб-сервером та сервером бази даних. Діаграма демонструє логічну організацію компонентів та підтверджує коректність обраної архітектури.

Також представлено структуру серверної частини проєкту, що включає основні модулі Django-додатку: моделі, представлення, шаблони, статичні файли та конфігураційні файли. Це підтверджує відповідність реалізації стандартам фреймворку та забезпечує модульність, масштабованість і зручність подальшої підтримки системи.

РОЗДІЛ 4. ОХОРОНА ПРАЦІ

Охорона праці під час розробки веб-порталу передбачає створення безпечних та комфортних умов праці для працівника, який виконує роботи за персональним комп'ютером. Робота програміста належить до категорії робіт з підвищеним навантаженням на зір, опорно-руховий апарат та нервову систему, тому важливо забезпечити відповідність робочого місця чинним нормам та стандартам.

Основними факторами, що впливають на безпеку праці під час розробки програмного забезпечення, є:

- тривала робота за комп'ютером;
- статичне навантаження на м'язи спини та ший;
- підвищене навантаження на органи зору;
- можливий вплив електромагнітного випромінювання;
- ризики, пов'язані з електробезпекою;
- пожежна небезпека електронного обладнання.

Для забезпечення безпечних умов праці необхідно дотримуватися вимог щодо організації робочого місця, мікроклімату, освітлення, ергономіки та правил експлуатації комп'ютерної техніки.

4.1 Умови які повинні бути забезпечені

1. Вимоги до робочого місця програміста Робочий стіл повинен мати висоту 72-75 см.
 - Крісла мають бути регульованими по висоті та углу нахилу спинки.

- Монітор розташовується на відстані 50–70 см від очей.
 - Верхня межа екрана повинна бути на рівні очей або трохи нижче.
 - Клавіатура розташовується на відстані 10–30 см від краю столу.
2. Вимоги до освітлення Освітленість робочої зони має становити 300–500 лк.
- Джерела світла не повинні створювати відблиски на екрані.
 - Бажано використовувати комбіноване освітлення: природне + штучне.
3. Вимоги до мікроклімату Температура: 20-24 °С.
- Вологість: 40-60%.
 - Регулярне провітрювання приміщення.
 - Відсутність протягів.
4. Вимоги до електробезпеки Все обладнання має бути заземлене.
- Забороняється використовувати пошкоджені кабелі та розетки.
 - Розетки повинні мати захисні автомати.
 - У приміщенні має бути вогнегасник.
5. Пожежна безпека Заборонено залишати увімкнене обладнання без нагляду.
- Не допускається використання несправних підовжувачів.
 - У приміщенні повинні бути засоби пожежогасіння та план евакуації.
6. Організація режиму праці Перерви кожні 60 хвилин по 5-10 хвилин.
- Гімнастика для очей.

- Зміна положення тіла для зменшення статичного навантаження.

Висновки до розділу 4

У даному розділі було розглянуто основні вимоги охорони праці, що стосуються організації робочого місця програміста під час розробки веб-порталу. Дотримання норм щодо освітлення, мікроклімату, ергономіки, електробезпеки та пожежної безпеки дозволяє мінімізувати ризики для здоров'я працівника та забезпечити комфортні умови праці. Виконання цих вимог сприяє підвищенню продуктивності, зниженню втоми та запобіганню професійним захворюванням, що є важливим аспектом у процесі створення програмного забезпечення.

ВИСНОВКИ

У ході виконання дипломної роботи було проведено комплексне дослідження предметної області, визначено основні проблеми та недоліки існуючих рішень, а також сформовано концепцію створення веб-порталу, спрямованого на надання користувачам структурованої та доступної інформації щодо здорового способу життя.

На основі проведеного аналізу були поставлені такі основні завдання:

- дослідити предметну область та визначити актуальність теми;
- проаналізувати існуючі аналоги та виявити їхні недоліки;
- сформулювати концепцію власної системи та визначити її переваги;
- розробити архітектуру веб-порталу та побудувати необхідні діаграми (Use Case, DFD, ER-діаграму, діаграму послідовностей);
- створити концептуальну та фізичну модель бази даних;
- реалізувати серверну частину системи на основі фреймворку Django;
- розробити інтерфейс користувача та забезпечити взаємодію з базою даних;
- виконати тестування основних функціональних модулів;
- визначити вимоги до програмного та апаратного забезпечення;
- розглянути питання охорони праці під час роботи з комп'ютером.

У процесі виконання роботи всі поставлені завдання були успішно реалізовані. Було створено повноцінний веб-портал, що включає:

- структуровану базу даних;

- модулі для перегляду тематичних матеріалів;
- механізм відображення оглядів та порівнянь;
- інтуїтивний інтерфейс користувача;
- серверну логіку, що забезпечує стабільну роботу системи.

Розроблена система відповідає вимогам сучасних веб-технологій, є масштабованою, зручною у використанні та може бути розширена в майбутньому шляхом додавання нових функціональних модулів.

Таким чином, мета дипломної роботи досягнута, а всі поставлені завдання — виконані. Результати роботи підтверджують ефективність обраних методів проєктування та реалізації веб-порталу, а також демонструють практичну цінність створеної системи.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Vincent, W. S. (2020). Django for Professionals.
2. Winter, A., Ammenwerth, E., Haux, R., Marschollek, M., Steiner, B., Jahn, F. (2023). Health Information Systems: Technological and Management Perspectives. Springer.
3. Mollborn, S., Lawrence, E. M., Saint Onge, J. M. (2021). “Contributions and Challenges in Health Lifestyles Research”. Journal of Health and Social Behavior.
4. Nichter, D. (2020+). Efficient MySQL Performance: Best Practices and Techniques.
5. Krogh, J. W. (2020+). MySQL 8 Query Performance Tuning.
6. Shaw, B., Badhwar, S., Guest, C., Chandra, B. (2024). Web Development with Django (2nd Edition). Packt Publishing.

Додаток А

ІНСТРУКЦІЯ КОРИСТУВАЧА ВЕБ-ПОРТАЛУ

1. Загальна інформація - даний веб-портал призначений для перегляду тематичних матеріалів, пов'язаних зі здоровим способом життя. Система забезпечує зручний доступ до інформації, оглядів, рекомендацій та структурованих матеріалів.

Інтерфейс portalу є інтуїтивним і не вимагає спеціальних технічних знань від користувача.

2. Вимоги до користувача

Для роботи з веб-порталом необхідно:

- сучасний веб-браузер (Google Chrome, Mozilla Firefox, Microsoft Edge);
- стабільне підключення до Інтернету;
- роздільна здатність екрана від 1280×720;
- увімкнена підтримка JavaScript.

3. Опис інтерфейсу та функціональних можливостей

Нижче наведено основні елементи інтерфейсу веб-порталу та їх призначення.

3.1 Головна сторінка

Головна сторінка є стартовою точкою роботи користувача з веб-порталом. На ній відображаються:

- список доступних тематичних розділів;
- короткі описи матеріалів;
- навігаційне меню;
- основні функції portalу;
- опис сайту;

Smart Reviews for Smarter Choices

This platform provides structured, evidence-based evaluations of fitness equipment, nutritional products, and related tools. All materials are prepared using a consistent analytical approach to support clear and well-reasoned product selection.

Evaluating Equipment Through Practical Use

Training equipment, accessories, and fitness tools are assessed through practical application and professional knowledge in physical conditioning. The goal is to provide users with reliable information that reflects real-world performance and functional value.

- ✓ Recommendations based on practical training experience
- ✓ Assessment of equipment in real usage conditions
- ✓ Analysis supported by formal education in physical training



Structured Analysis of Nutrition and Supplementation

Food products, supplements, and their combinations are examined using scientific sources and principles of human physiology. The aim is to provide clear, evidence-based insights that support long-term health and informed dietary choices.

- ✓ Nutrition guidance grounded in scientific literature
- ✓ Evaluation of effective product and supplement combinations
- ✓ Support for developing sustainable and health-oriented habits

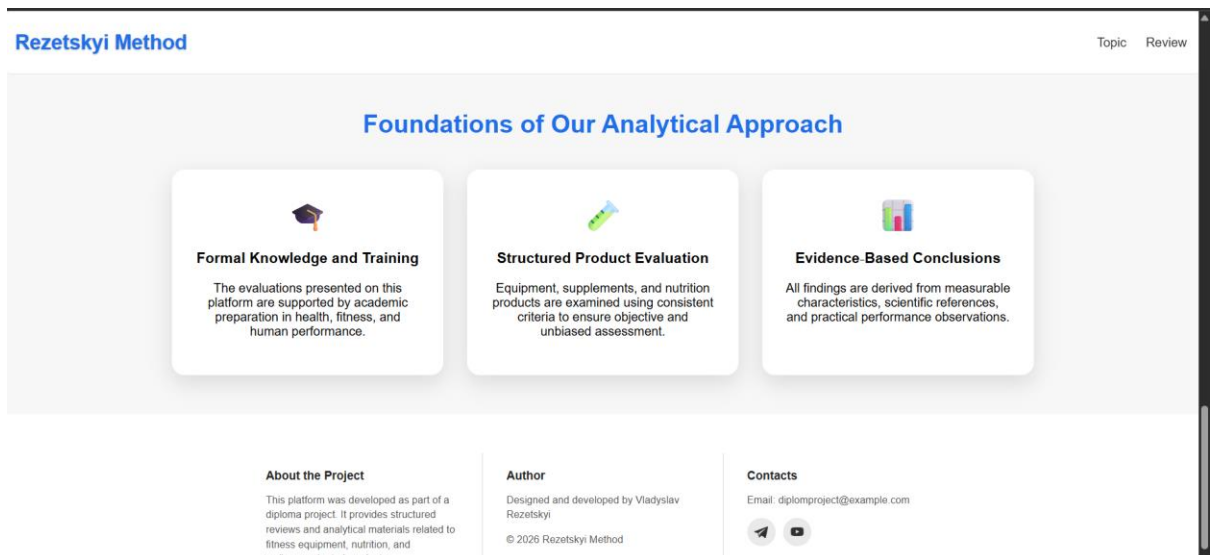


Рисунок 3.1 - Головна сторінка веб-порталу

3.2 Сторінка теми (Topic)

Після вибору теми користувач переходить на сторінку з детальним описом. На цій сторінці відображаються:

- **зображення (обкладинку);**
- **назву матеріалу (title);**
- **короткий опис;**
- **орієнтовний час читання;**

Таке представлення контенту забезпечує швидке ознайомлення з доступними матеріалами та дозволяє користувачу обрати найбільш релевантну тему.

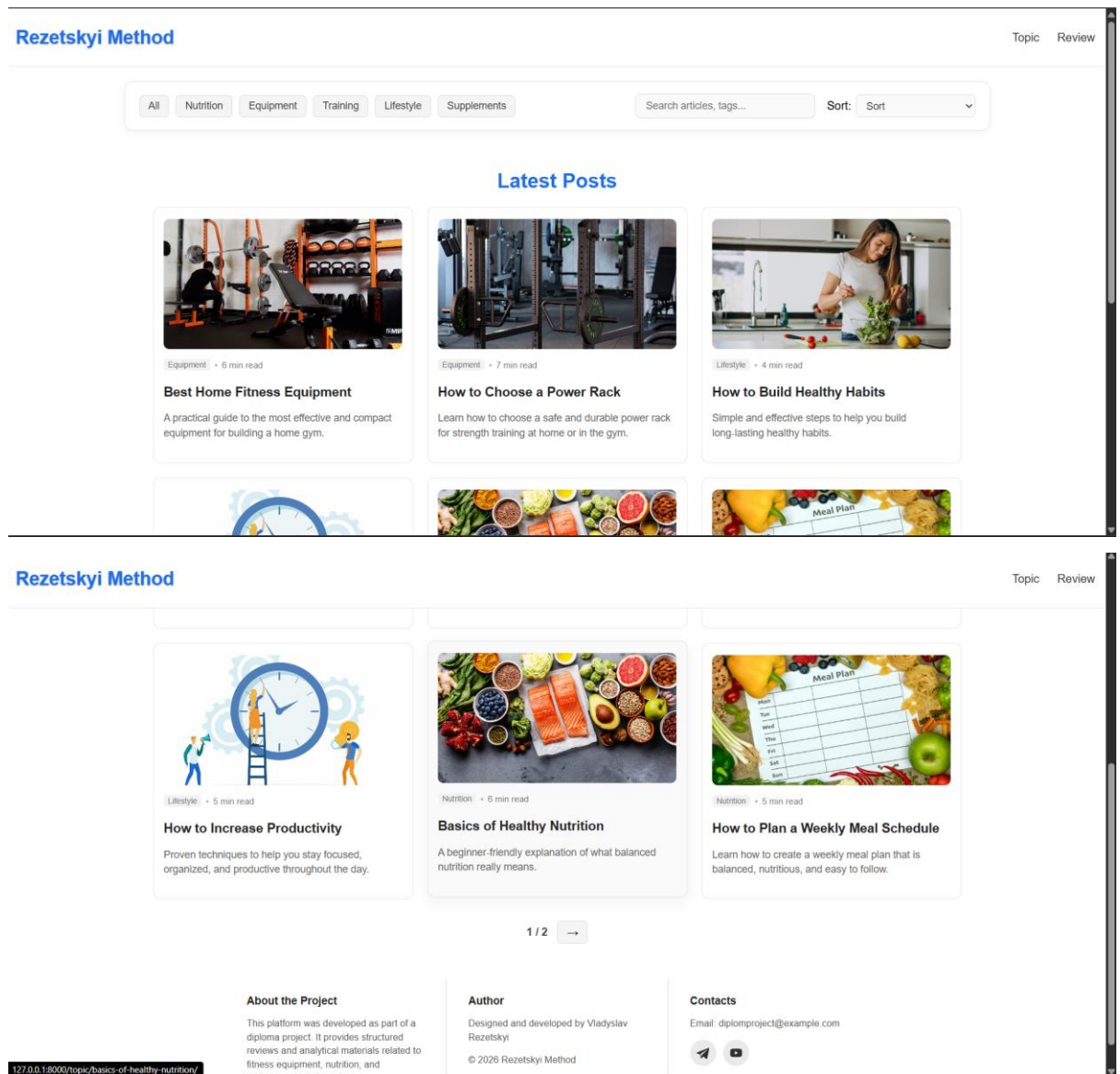


Рисунок 3.2 - Сторінка Топіс розділу

3.3 Сторінка огляду (Review)

Усі матеріали веб-порталу структуровані у вигляді карток, що забезпечує зручність перегляду та швидку навігацію.

Сторінка огляду містить:

- **зображення** — візуальне представлення теми;
- **title** — назва матеріалу;
- **короткий опис** — 1–2 речення;
- **reading time** — орієнтовний час читання;

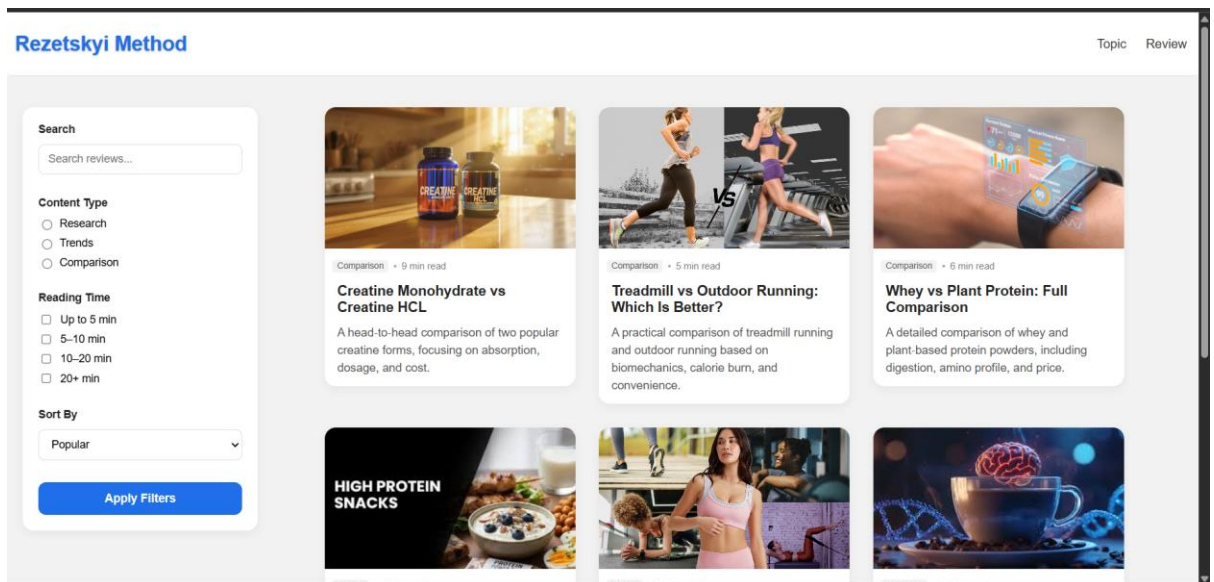


Рисунок 3.3 - Сторінка Review розділу

3.4 Пошук інформації

Пошукова система дозволяє швидко знаходити потрібні матеріали за ключовими словами. Функціонал включає:

- введення пошукового запиту;
- миттєве відображення результатів;
- перехід до відповідної теми або огляду.

3.5 Меню навігації

Меню навігації доступне на всіх сторінках порталу та дозволяє:

- повернутися на головну сторінку;
- переглянути список тем;
- перейти до оглядів;

Меню забезпечує швидкий доступ до всіх основних розділів.

4. Завершення роботи

Для завершення роботи з порталом достатньо закрити вкладку браузера.
Система не потребує авторизації та не зберігає персональних даних користувача.