

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра інженерії програмного забезпечення

ПОЯСНЮВАЛЬНА ЗАПИСКА

до бакалаврської роботи
на ступінь вищої освіти бакалавр

на тему: **«РОЗРОБКА WEB-СЕРВІСУ ДЛЯ СТВОРЕННЯ
ІНДИВІДУАЛЬНОГО ПЛАНУ ХАРЧУВАННЯ ТА ТРЕНУВАНЬ ЛЮДИНИ
НА ОСНОВІ БІБЛІОТЕКИ REACT.JS»**

Виконав: студент 4 курсу, групи ПД-42
спеціальності
121 Інженерія програмного забезпечення
(шифр і назва спеціальності/спеціалізації)

Малюк В.В.
(прізвище та ініціали)

Керівник Яскевич В.О.
(прізвище та ініціали)

Рецензент

(прізвище та ініціали)

Київ – 2023

ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра Інженерії програмного забезпечення

Ступінь вищої освіти - «Бакалавр»

Спеціальність підготовки – 121 «Інженерія програмного забезпечення»

ЗАТВЕРДЖУЮ

Завідувач кафедри

Інженерії програмного забезпечення

Негоденко О.В.

“ _____ ” _____ 2023 року

ЗАВДАННЯ
НА БАКАЛАВРСЬКУ РОБОТУ СТУДЕНТА

Малюку Вадиму Володимировичу

(прізвище, ім'я, по батькові)

1. Тема роботи: «Розробка web-сервісу для створення індивідуального плану харчування та тренувань людини на основі бібліотеки React.JS»

Керівник роботи: Яскевич Владислав Олександрович, к.т.н, доцент

(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом вищого навчального закладу від «24» лютого 2023 року №26.

2. Строк подання студентом роботи «01» червня 2023 року.
3. Вхідні дані до роботи:
- 3.1 Науково-технічна література з питань, пов'язаних з програмним забезпеченням щодо розробки.
- 3.2 Існуючі інструменти розробки web-сайтів
- 3.3 Науково-технічна література.
- 3.4 Положення побудови систем розробки web-сайтів.

4. Зміст розрахунково-пояснювальної записки.

- 4.1 Аналіз актуальності та проблематики розроблюваного web-сервісу.
- 4.2 Аналіз та вибір інструментів для реалізації web-сервісу.
- 4.3 Проектування web-сервісу.
- 4.4 Висновки.

5. Перелік демонстраційного матеріалу
 - 5.1 Титульний слайд.
 - 5.2 Мета, об'єкт, предмет, наукова новизна дослідження.
 - 5.3 Актуальність.
 - 5.4 Аналіз аналогів.
 - 5.5 Технічні завдання.
 - 5.4 Програмні засоби та інструменти реалізації.
 - 5.6. Розробка архітектури.
 - 5.7 Реалізація програми.
 - 5.8 Висновки.
 - 5.9 Апробація результатів дослідження.
 - 5.10 Кінцевий слайд.
6. Дата видачі завдання «25» лютого 2023р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів бакалаврської роботи	Строк виконання етапів роботи	Примітка
1	Підбір науково-технічної літератури	25.02.2023	Виконано
2	Дослідження аналогів та актуальності додатку	14.03.2023	Виконано
3	Аналіз та вибір інструментів для розробки додатку	26.03.2023	Виконано
4	Проектування та реалізація	09.04.2023	Виконано
5	Вступ, висновки, реферат	03.05.2023	Виконано
6	Розробка обов'язкових демонстраційних матеріалів	11.05.2023	Виконано
7	Попередній захист роботи	23.05.2023	Виконано
8	Здача роботи	01.06.2023	Виконано

Студент _____ Малюк В.В.
 (підпис) (прізвище та ініціали)

Керівник роботи _____ Яскевич В.О.
 (підпис) (прізвище та ініціали)

РЕФЕРАТ

Текстова частина бакалаврської роботи с. 53, табл. 2, рис. 24, джерел 6.

Об'єкт дослідження – розробка web-сервісу.

Предмет дослідження – web-сервіс для створення індивідуального плану харчування та тренування.

Мета роботи – підвищити рівень контролю харчування та тренування людини шляхом розробки web-сервісу за допомогою бібліотеки react.js.

У дипломному проекті проведено аналіз фітнес web-сервісів. Проаналізовано та виявлено переваги і недоліки програмних інструментів розробки. Розглянуті та проаналізовані особливості наявних фітнес web-сервісів.

Web-сервіс було створено за допомогою доповненої мови JSX та стилю сторінок CSS. Web-сервіс було створено у середовищі Visual Studio Code, алгоритми для розрахунку кількості калорій та потрібного плану тренування були написані на мові програмування JS. Для створення елементів графічного дизайну та прототипування було використано Figma.

Даний web-сервіс слугує для того, щоб люди, які зацікавленні в підтримці граного стану свого фізичного здоров'я, мали змогу самостійно створити під свої потреби план тренування чи харчування.

ЗМІСТ

ПОЯСНЮВАЛЬНА ЗАПИСКА	1
на тему: «РОЗРОБКА WEB-СЕРВІСУ ДЛЯ СТВОРЕННЯ	
ІНДИВІДУАЛЬНОГО ПЛАНУ ХАРЧУВАННЯ ТА ТРЕНУВАНЬ ЛЮДИНИ	
НА ОСНОВІ БІБЛІОТЕКИ REACT.JS»	1
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ	
ТЕХНОЛОГІЙ	2
ЗАТВЕРДЖУЮ	2
ЗАВДАННЯ	2
КАЛЕНДАРНИЙ ПЛАН	3
РЕФЕРАТ	6
ВСТУП	10
1. ОГЛЯД ІСНУЮЧИХ WEB-СЕРВІСІВ ТА АКТУАЛЬНІСТЬ ПРОЕКТУ	12
1.1 Аналіз та загальна характеристика web-сервісів(web-сайтів).....	12
1.2 Аналіз уже наявних аналогів	13
1.2.1 MyFitnessPal	14
1.2.2 FitSW	16
1.2.3 JEFIT	17
1.2.4 Порівняння	19
1.3 Актуальність проекту.....	21
2. ПРОЕКТУВАННЯ WEB-СЕРВІСУ ДЛЯ СТВОРЕННЯ ПЛАНУ	
ХАРЧУВАННЯ ТА ТРЕНУВАННЯ ЛЮДИНИ	24
2.1 Аналіз та обґрунтування вибору середовища та інструментів розробки	
проекту.....	24
2.1.1 Visual Studio Code	26
2.1.2 React	29
2.1.3 JSX.....	32
2.1.4 Firebase.....	35
2.2 Загальний алгоритм реалізації проекту.....	37

2.3 Функціонал продукту	42
3. ПРОГРАМНА РЕАЛІЗАЦІЯ ФІТНЕС WEB-SERVICES	43
3.1 Оцінка та планування розробки фітнес web-сервісу	43
3.2 Функціонал web-сервісу.	45
4. ПРИКЛАДИ ВИКОРИСТАННЯ ТА ТЕСТУВАННЯ WEB-SERVICES.....	55
4.1 Опис роботи web-сервісу.	55
4.2 Тестування роботи фітнес web-сервісу.	57
ВИСНОВКИ.....	59
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	61
ДОДАТОК А	62

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ

WHO - Всесвітня організація охорони здоров'я

VS Code - Visual studio code

DOM – Document object model

JSX – JavaScript XML

MVP – Мінімальна життєздатна версія

KFF – Key Features First

BDD – Behavioral Driven Development

ВСТУП

У сучасному світі люди все більше звертають увагу на своє здоров'я та фізичну форму. Одним з ключових аспектів здорового способу життя є правильне харчування та регулярна фізична активність.

За даними Всесвітньої організації охорони здоров'я, більше 1 мільярда дорослих людей у світі мають зайву вагу, а з них 300 мільйонів - офіційно вважаються ожирілими. Це свідчить про те, що проблема зайвої ваги є актуальною для багатьох людей та потребує вирішення. В той же час, здорове харчування та регулярна фізична активність можуть позитивно вплинути на стан здоров'я та допомогти зберегти фізичну форму.

У зв'язку з цим, розробка web-сервісу для створення індивідуального плану харчування та тренувань людини є актуальною темою для дослідження. Такий сервіс може допомогти людям досягати своїх фітнес-цілей, підтримувати здоровий спосіб життя та покращувати якість свого життя в цілому.

Існує достатня кількість фітнес web-сервісів, які мають свої технічні особливості, свою простоту використання, свої переваги та недоліки. Наявність потрібної інформації, зрозумілого інтерфейсу та потрібних інструментів визначає наскільки такий web-сервіс буде користуватися попитом серед користувачів.

У даній дипломній роботі було розглянуто процес розробки web-сервісу для створення індивідуального плану харчування та тренувань людини. Були визначені його ключові функціональні та нефункціональні вимоги, розроблено дизайн та реалізовано прототип сервісу.

У цілому, розробка web-сервісу для створення індивідуального плану харчування та тренувань є актуальною та захоплюючою темою, з потенціалом позитивно вплинути на здоров'я та благополуччя людей. Завдяки новітній технології та можливостям мережі Інтернет, люди можуть отримати персоналізований сервіс, який допоможе їм досягти своїх фітнес-цілей та зберегти здоров'я протягом усього життя.

Метою є створення web-сервісу для створення індивідуального плану

харчування та тренувань людини. Web-сервіс повинен мати низький поріг входу та простий і мінімалістичний інтерфейс для легкої адаптації нових користувачів. Для досягнення мети було складено такий список задач:

- а) Огляд архітектури існуючих web-сервісів та аналіз проекту.
- б) Порівняння та аналіз існуючих аналогів для виявлення їх переваг та недоліків.
- в) Розробка та реалізація web-сервісу.
- г) Тестування web-сервісу, щоб переконатися в його працездатності та відповідності вимогам.

1. ОГЛЯД ІСНУЮЧИХ WEB-СЕРВІСІВ ТА АКТУАЛЬНІСТЬ ПРОЕКТУ

1.1 Аналіз та загальна характеристика web-сервісів(web-сайтів)

Web-сайт - це електронна сторінка або набір сторінок, які зберігаються на сервері та можуть бути доступні через Інтернет. Web-сайти містять різноманітний контент, такий як текст, зображення, відео та інші медіа-елементи, які можна переглянути та взаємодіяти з ними за допомогою web-браузера. Web-сайти можуть бути використані для різних цілей: таких як реклама товарів та послуг, комунікація, розміщення інформації, продаж товарів та багато іншого.

Web-сайти можуть мати різну мету та функціональність. Деякі web-сайти призначені для розваг – такі, як ігрові платформи, або сайти зі зображеннями та відео. Інші web-сайти можуть бути призначені для комунікації – такі, як соціальні мережі та форуми. Є також web-сайти, що надають користувачам можливість здійснювати покупки онлайн, зокрема електронні комерційні платформи.

За допомогою web-сайтів можна забезпечити доступ до інформації та послуг у будь-якому місці та в будь-який час, за умов наявності Інтернет-підключення. Крім того, web-сайти можуть бути інструментом для збору даних, маркетингових досліджень та спілкування зі споживачами.

Web-сайти є невід’ємною частиною більшості сфер життя у сучасному світі. Вони забезпечують широкий спектр можливостей, які допомагають людям знаходити інформацію, здійснювати покупки, спілкуватися, навчатися та багато іншого.

У бізнесі web-сайти відіграють важливу роль в рекламі продуктів та послуг, просуванні бренду, підтримці клієнтів і співпраці з партнерами. Багато компаній мають свої web-сайти, де клієнти можуть ознайомитися з асортиментом, цінами, зробити замовлення та скористатися іншими сервісами.

У галузі освіти web-сайти допомагають студентам знаходити ресурси для навчання та підвищення кваліфікації, а також забезпечують можливість дистанційного навчання. Web-сайти також допомагають інституціям освіти

залучати студентів та забезпечувати їм зручний доступ до необхідної інформації.

У сфері здоров'я web-сайти надають корисну інформацію про захворювання, ліки та методи лікування. Також вони дозволяють замовляти медичні послуги, записуватися на консультації, спілкуватися з лікарями та отримувати консультації в онлайн режимі.

У сфері культури та мистецтва web-сайти надають можливість дізнатися про майстер-класи, виставки та інші події. Вони також допомагають широкому загалу мати доступ до творчості та культурних цінностей з усього світу. У підсумку, web-сайти є потужним інструментом для комунікації, спілкування та розваг. Вони дають можливість представити інформацію про продукти та послуги, надати можливість здійснення покупок в Інтернеті, допомагають в організації роботи і забезпеченні зв'язку з клієнтами. Web-сайти забезпечують доступ до великої кількості інформації, дозволяють легко та швидко отримати необхідну інформацію, а також забезпечують можливість спілкування з іншими користувачами з усього світу. Окрім цього, web-сайти зручні для користувачів, оскільки їх можна відкривати на будь-якому пристрої, що має доступ до Інтернету, що дозволяє отримувати інформацію в будь-який час та в будь-якому місці.

1.2 Аналіз уже наявних аналогів

Для того, щоб почати займатись спортом і правильно харчуватися важливо мати доступ до різноманітних web-ресурсів, що мають корисну і зрозумілу інформацію як це зробити в першу чергу правильно. Важливо не тільки отримувати текстову інформацію про тренування та інструкцію їх виконання, а й відеоуроки також.

Що стосується створення плану харчування, там достатньо буде і текстової інформації з однією картинкою, щоб зрозуміти, що саме собою представляє та чи інша страва чи напій. Порівняємо фітнес web-сервіси наступних платформ:

MyFintessPal, JeFit, FitSW.

1.2.1 MyFitnessPal

MyFitnessPal - це безкоштовний онлайн-сервіс, який допомагає користувачам стежити за їх харчуванням та фізичною активністю. За допомогою MyFitnessPal користувачі можуть вести щоденник їжі та тренувань, стежити за кількістю калорій, білків, жирів та вуглеводів, які вони споживають, і порівнювати їх зі своєю денною нормою.

MyFitnessPal має велику базу даних продуктів, що дозволяє легко знайти і додати до щоденника їжі будь-який продукт, який споживаєте. Крім того, сервіс дозволяє додавати власні страви та рецепти, що дуже зручно для тих, хто готує вдома.

Також у MyFitnessPal є функція стеження за фізичною активністю, яка дозволяє користувачам вести щоденник тренувань та враховувати витрачені калорії. Крім того, сервіс підтримує підключення до інших фітнес-додатків та пристроїв, що дозволяє автоматично синхронізувати дані про активність.

У MyFitnessPal є безкоштовна та платна версії. Платна версія має додатковий функціонал, наприклад, можливість відстежувати рівень макронутрієнтів, налаштування цілей та більш детальні аналітичні звіти.

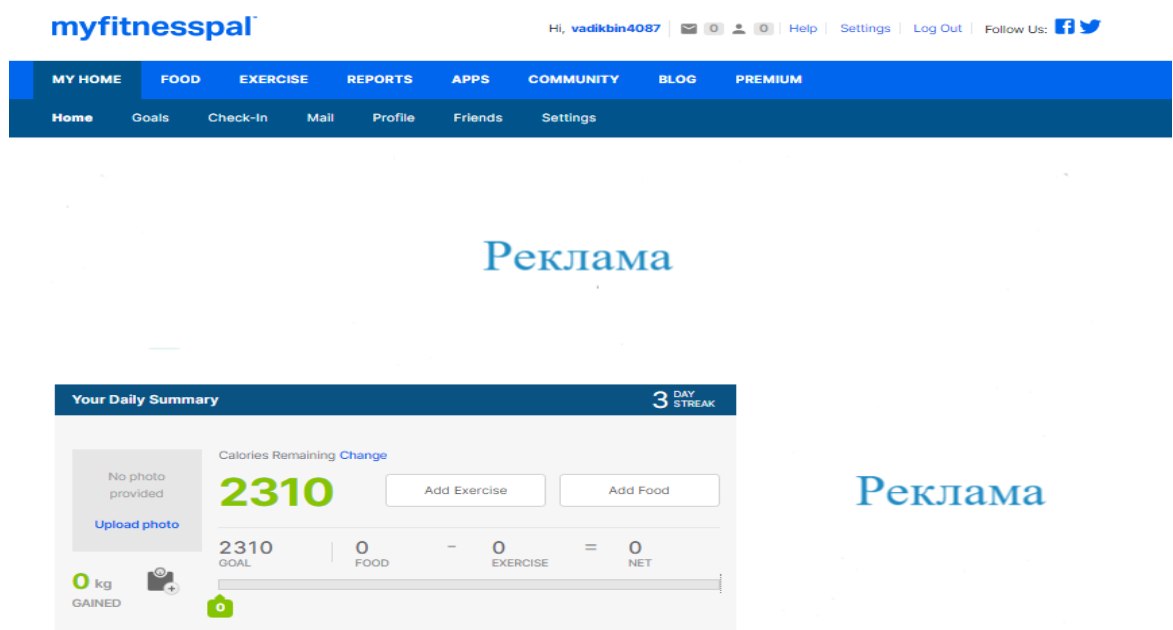


Рисунок 1.1 Сайт MyFitnessPal. Головна сторінка

Однією з переваг MyFitnessPal є те, що він є в більшій мірі безкоштовним. Це дає людям будь-якого фінансового статку користуватися основними функціями web-сервісу повністю безкоштовно. Також є і інші переваги, серед яких варто відмітити:

а) Можливість налаштувати персоналізований план: сервіс дозволяє вказати свої цілі та розрахувати необхідну кількість калорій, білків, жирів та вуглеводів, що допомагає досягати бажаних результатів.

б) Широкий вибір продуктів: у сервісі є база даних з більш ніж 11 мільйонами продуктів, що дозволяє швидко знайти необхідну інформацію про калорійність і склад продуктів.

в) Функції соціальної мережі: сервіс дозволяє додавати друзів та ділитися своїми досягненнями, що надає мотивації для досягнення більшого успіху в тренуваннях та харчуванні.

Серед головних недоліків можна відмітити:

а) Застарілий інтерфейс: Дизайн та інтерфейс MyFitnessPal можуть здатися застарілими та складними для користувачів, особливо для тих, хто не має досвіду використання подібних web-сайтів.

б) Реклама: На деяких сторінках сервісу можуть з'являтися рекламні банери та інші рекламні матеріали, що можуть бути надокучливими для користувачів.

в) Платні функції: Щоб отримати доступ до деяких функцій, користувачам може доводитися платити підписку, що може становити додаткові витрати.

Не зважаючи на мінуси, MyFitnessPal є чудовим web-сервісом для створення особистого плану тренування, чи харчування. Завдяки великій кількості інструментів, користувач має змогу зробити свій аналітичний звіт ще більш зрозумілим та інформаційним.

1.2.2 FitSW

FitSW - це онлайн-сервіс для створення індивідуальних планів тренувань та програм харчування. Користувачі можуть використовувати вбудовані функції для створення своїх власних тренувальних програм та дієти, а також використовувати готові програми, що розроблені професіоналами.

Одна з головних переваг FitSW полягає в тому, що користувачі можуть настроїти свої плани тренувань та програми харчування відповідно до своїх цілей та індивідуальних потреб. Сервіс має також інтегрований календар, що дозволяє користувачам відстежувати свої тренування та їжу із зручним інтерфейсом.

Недоліком FitSW є те, що деякі функції та програми доступні тільки в платній версії, що може бути обмеженням для деяких користувачів з обмеженим бюджетом. Крім того, застосунок може виявитися складним у використанні для новачків, які не знають як правильно скласти тренувальні програми або плани харчування.

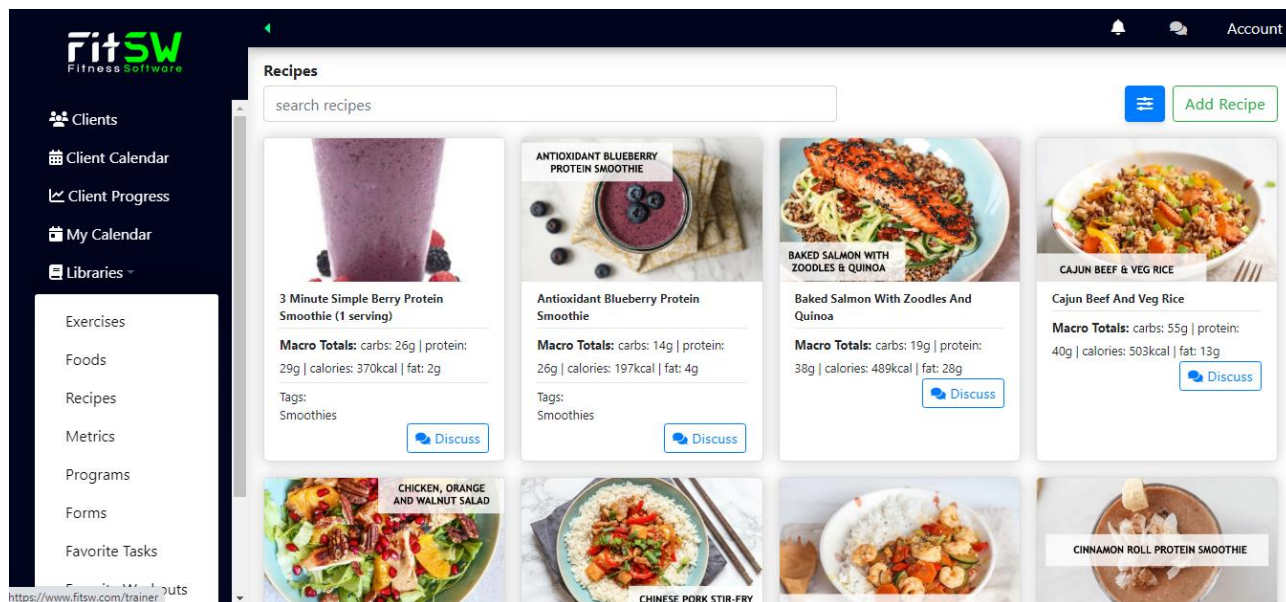


Рисунок 1.2 Сайт FitSW. Підбірка рецептів

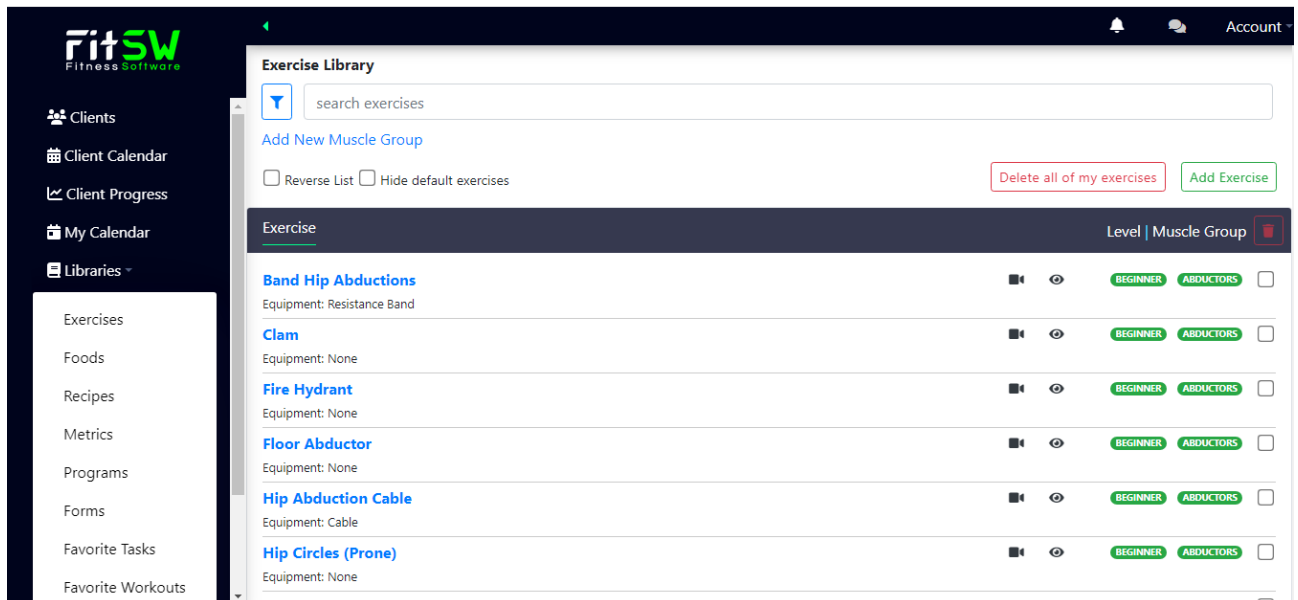


Рисунок 1.3 Сайт FitSW. Підбірка вправ для тренування

Нажаль, web-сервіс не підтримує жодну іншу мову, крім англійської, що може бути проблемою для деяких користувачів нашої країни. В цілому web-сервіс надає корисні інструменти для створення свого власного плану тренування чи харчування, що правда через доволі складний інтерфейс та знову ж таки – мову, скористатися ними буде доволі складно для користувачів з низьким рівнем навичок у користуванні телефоном.

1.2.3 JEFIT

JEFIT - це онлайн-сервіс та мобільний додаток для тренувань, який допомагає користувачам створювати та виконувати плани тренувань, вести журнал тренувань та відстежувати свій прогрес у фітнесі.

Сервіс має велику базу даних вправ та програм тренувань, включаючи як базові, так і вдосконалені, які можуть бути налаштовані для задоволення потреб користувачів. Крім того, в сервісі JEFIT є можливість побачити відеоінструкції до кожної вправи.

Однією з головних переваг JEFIT є те, що користувачі можуть створювати свої власні плани тренувань, враховуючи свої цілі та обмеження. Крім того, сервіс

надає можливість вести журнал тренувань, де користувачі можуть записувати виконані вправи, вагу, пульс та інші параметри, що допомагає відстежувати свій прогрес та налаштовувати програму тренувань.

Недоліком JEFIT є те, що під час використання безкоштовної версії сервісу, користувачі можуть зіткнутися з деякими обмеженнями, такими як обмежена кількість доступних програм тренувань та вправ. Крім того, інтерфейс додатку може бути складним для новачків, які тільки починають займатися фітнесом.

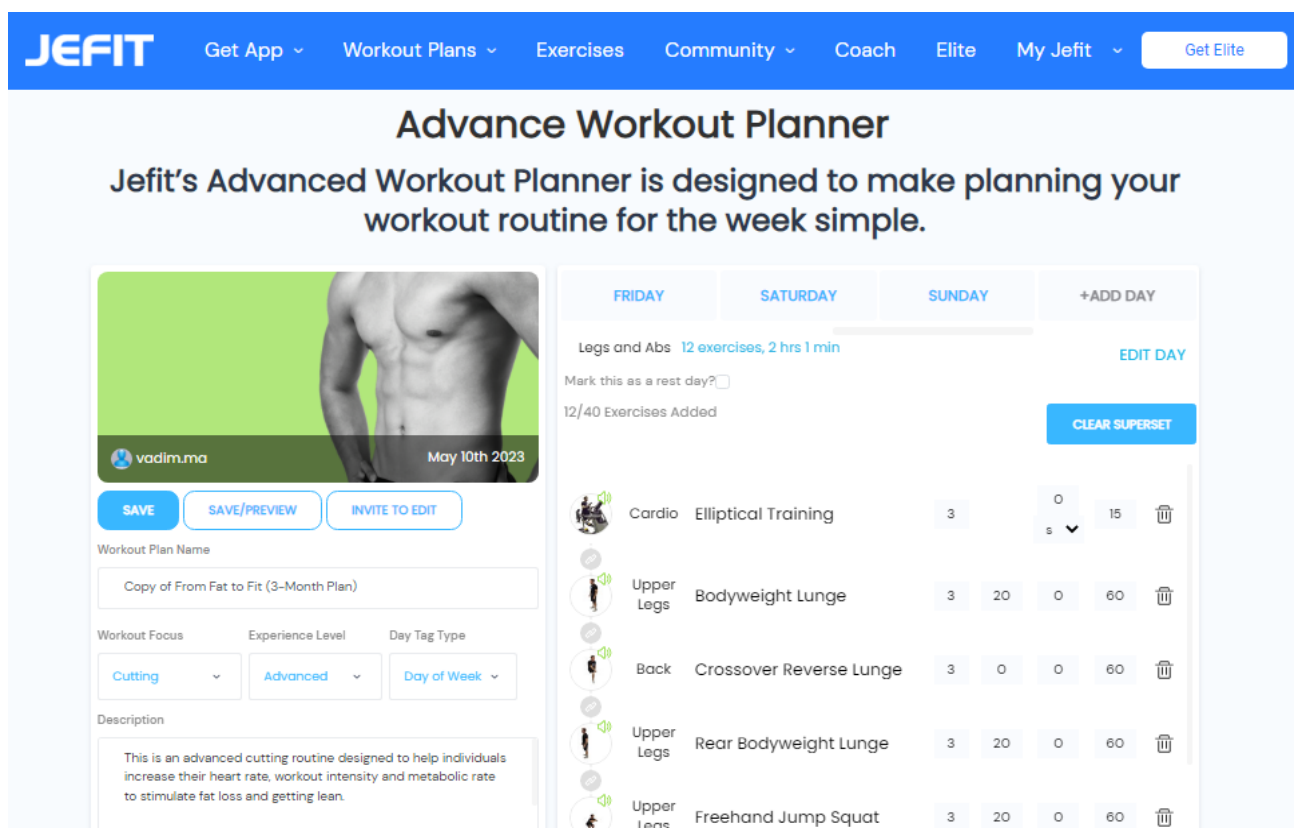


Рисунок 1.4 Сайт JEFIT. Особистий кабінет

Знову ж таки, доступна тільки англійська мова, що буде проблемою для користувачів, які не знають цю мову. Нажаль, web-сервіс не має інструменту для створення індивідуального плану харчування, але має досить гарний інструмент для створення індивідуального плану тренування різної складності під кожен людину.

1.2.4 Порівняння

Інструмент для створення плану харчування: Якщо користувачі web-сервісу бажають створити свій особистий план харчування під свої потреби, то вони можуть користуватися сервісом MyFitnessPal або FitSW. Нажаль, з вище перелічених web-сервісів, тільки вони мають таку функцію, оскільки решта web-сайтів направлені на створення плану тренування людини.

Інструмент для створення плану тренування: Якщо користувачі web-сервісу бажають створити свій особистий план тренування під свої потреби, то вони можуть користуватися всіма вище згаданими web-сайтами. Оскільки кожен сайт має свою базу вправ для тренування та свої налаштування особистого плану тренувань, користувачу доведеться обирати найбільш зручний для себе варіант, але нажаль тільки сервіс MyFitnessPal має змогу в налаштуваннях обрати російську мову.

Простота використання: У всіх трьох фітнес web-сервісах доволі складний для користувача-початківця інтерфейс, в деяких випадках додає складності те, що мову змінити не можливо. Загалом ця проблема вирішується доволі просто, але у випадку цих web-сайтів, буде набагато ліпше, якщо вони все ж дадуть можливість змінювати мову інтерфейсу. На даний момент найпростішим у користуванні залишається MyFitnessPal, далі JEFIT та FitSW.

Таблиця 1.1 – Переваги та недоліки існуючих web-сервісів

Платформа	Переваги	Недоліки	Простота користування	Створення плану харчування	Створення плану тренування
MyFitnessPal	Не занадто складний інтерфейс, є можливість зміни мови та зручні інструменти	Реклама всюди, платні деякі функції	Середньо	Так	Так
JEFIT	Є можливість додаткових безкоштовних функцій, які є платними в інших сервісах	Деякі функції платні, складний інтерфейс	Складно	Ні	Так
FitSW	Є можливість налаштувати під себе плани харчування та тренування	Складний у використанні, деякі функції платні	Складно	Так	Так

Підводячи підсумки, можна зробити висновок, що для заняття спортом існують різні web-сервіси, кожен з яких має свої переваги та недоліки.

Кожен web-сервіс має свої переваги та недоліки, які можуть бути важливими при виборі основного сервісу для відстеження своїх результатів, чи створення індивідуального плану тренування, чи харчування.

Перша платформа, яку розглядаємо, - MyFitnessPal. Вона є практично безкоштовною та містить багато корисної інформації, що робить її чудовим варіантом для тих, хто хоче розпочати свої тренування, чи почати правильно та збалансовано харчуватися. Не зовсім складна у використанні, крім того є можливість змінити мову інтерфейсу у налаштуваннях.

Наступна платформа - JEFIT. Вона вже є більш складною у використанні та не має можливості створити індивідуальний план харчування, що є мінусом, оскільки на харчуванні ґрунтується весь результат ваших тренувань, але доволі зрозумілий і простий інструмент для створення власного плану тренування.

FitSW - це ще один web-сервіс для тренувань. Нажаль, має доволі складний інтерфейс та не має можливості змінити мову, але є зручні та зрозумілі інструменти для створення індивідуального плану харчування та тренування людини.

Загалом, всі web-сервіси мають свої плюси та мінуси в певних аспектах. Але, якщо користувачу важливо мати під рукою і план харчування і план тренування, то кращим для цього є сервіс MyFitnessPal, решту сервісів також не варто відкидати, оскільки, якщо важливо підібрати саме комфортний план тренування, то кращим у цьому буде web-сервіс JEFIT, тому, якщо користувач буде мати можливість і бажання поєднати два кращих інструменти з різних сайтів, він може взяти план харчування з MyFitnessPal та план тренування з JEFIT.

1.3 Актуальність проекту

Створення фітнес web-сервісів є дуже актуальною темою в сучасному світі, оскільки здоровий спосіб життя та фітнес-індустрія знаходяться в центрі уваги багатьох людей. Web-сервіси забезпечують доступність та зручність для користувачів, дозволяючи їм з легкістю створювати індивідуальні плани харчування та тренувань, вести щоденники спортивних досягнень, отримувати корисні поради та рекомендації від професійних тренерів.

Фітнес web-сервіси можуть бути корисними як для особистого користування, так і для бізнесу. Багато фітнес-центрів, тренерів та спортивних товариств створюють власні web-сервіси, щоб збільшити кількість клієнтів та покращити їхнє задоволення від послуг. Також web-сервіси можуть бути корисними для спортивних заходів та змагань, надаючи користувачам зручний спосіб реєструватись та отримувати інформацію про події.

За даними дослідження, проведеного Міністерством охорони здоров'я України разом з Всесвітньою організацією охорони здоров'я (WHO) та Інститутом медико-соціальних проблем сім'ї НАН України, у 2021 році 63% дорослого населення України мають проблеми з надлишковою вагою, а 27% - страждають на ожиріння. Також, 14% дітей у віці 6-9 років та 17% підлітків у віці 13-15 років в Україні мають надмірну масу тіла. Це є серйозною проблемою для здоров'я людей, яка може викликати ризик розвитку різних захворювань, зокрема серцево-судинних, діабету та інших хронічних захворювань. Тому актуальність створення фітнес web-сервісів, які допоможуть людям підтримувати здоровий спосіб життя та контролювати свою вагу, велика.

Фітнес web-сервіси є корисним інструментом для тих, хто хоче підтримувати своє здоров'я та форму. Вони дозволяють створити індивідуальний план харчування та тренувань, вести облік калорій та іншої статистики, спілкуватися з іншими користувачами та отримувати корисні поради від професійних тренерів.

Крім того, фітнес web-сервіси можуть бути корисними для людей з різними рівнями фізичної підготовки. Вони дозволяють вибрати програму тренувань та харчування, яка відповідає вашому рівню підготовки та меті, яку людина ставить перед собою.

Крім того, фітнес web-сервіси можуть бути корисними для тренерів, які можуть вести облік своїх клієнтів та надавати їм індивідуальні рекомендації.

Таким чином, фітнес web-сервіси можуть бути корисними для широкого кола людей, які хочуть підтримувати своє здоров'я та форму.

Правильне харчування та регулярні тренування важливі для здоров'я та довголіття людини. Зокрема, правильна дієта може допомогти знизити ризик розвитку серцево-судинних захворювань, діабету, ожиріння та інших хронічних захворювань. Регулярні ж тренування допомагають підтримувати здоров'я серцево-судинної, м'язової та нервової систем, а також сприяють зниженню рівня стресу та поліпшенню настрою. В цілому, правильне харчування та тренування є важливими складовими здорового та активного способу життя.

Неправильне харчування та малоактивний спосіб життя можуть призвести до

численних проблем зі здоров'ям. Наприклад, зайва вага або ожиріння можуть збільшити ризик розвитку таких захворювань, як діабет, серцево-судинні захворювання, артеріальна гіпертензія та інші. Недостатня фізична активність може також призвести до проблем зі здоров'ям, таких як слабкість м'язів, поганий тонус, ризик розвитку депресії та інші. Окрім цього, неправильне харчування може призвести до нестачі необхідних живильних речовин та вітамінів, що може погіршити загальний стан здоров'я та знизити імунітет.

Фітнес web-сервіси можуть бути розраховані на широку аудиторію, яка зацікавлена в здоровому способі життя та покращенні свого фізичного стану. Зазвичай, такі сервіси користуються популярністю серед людей, які займаються спортом на різних рівнях - від початківців до професійних спортсменів. Також фітнес web-сервіси можуть бути корисні для людей, які мають проблеми зі здоров'ям, і хочуть покращити своє фізичне самопочуття. Оскільки такі сервіси можуть надавати індивідуальні рекомендації щодо харчування та тренувань, вони можуть бути корисними для людей будь-якого віку та рівня фізичної підготовки.

Загалом, створення фітнес web-сервісів є дуже актуальним в сучасному світі, так як вони можуть бути корисними для різних аудиторій, таких як спортсмени, початківці в фітнесі та люди, які прагнуть покращити своє здоров'я та фізичну форму. Фітнес web-сервіси можуть надавати користувачам цінну інформацію про правильне харчування та тренування, допомагати планувати та відстежувати свій прогрес, а також мотивувати їх до досягнення своїх цілей. З урахуванням зростаючої популярності фітнесу та здорового способу життя, можна очікувати, що попит на фітнес web-сервіси буде лише зростати в майбутньому.

2. ПРОЕКТУВАННЯ WEB-SERVISU ДЛЯ СТВОРЕННЯ ПЛАНУ ХАРЧУВАННЯ ТА ТРЕНУВАННЯ ЛЮДИНИ

2.1 Аналіз та обґрунтування вибору середовища та інструментів розробки проекту

В цьому розділі будуть розглянуті основні інструменти та середовище розробки, обрані для створення фітнес web-сервісу для створення плану харчування та тренування людини.

Для створення фітнес web-сервісу було обрано мову програмування JavaScript. Ця мова є однією з найпопулярніших мов програмування в світі, що забезпечує наявність великої кількості ресурсів для вивчення та підтримки. Крім того, JavaScript є мовою програмування, яка є платформи-незалежною, тобто код, написаний на мові JavaScript, можна запускати на будь-якій операційній системі, що є дуже зручним.

Основна частина коду web-сервісу написана за допомогою JSX. JSX - це розширення JavaScript, що дозволяє описувати структуру web-сторінок або інтерфейсів користувача у вигляді коду, який схожий на HTML. За допомогою JSX, можна описати компоненти, які будуть відображені на сторінці, та взаємодіяти з ними через JavaScript. JSX зручний у використанні, оскільки дозволяє писати код на одній мові, замість того, щоб використовувати окремі файли для HTML, CSS та JavaScript. JSX використовується в більшості фреймворків та бібліотек, які використовують React, таких як Next.js, Gatsby, а також React Native для розробки мобільних додатків.

React є потужним інструментом для розробки інтерфейсів користувача, який має ряд переваг, які роблять його популярним серед розробників. Використання React дозволяє ефективно відображати зміни в стані компонентів та автоматично оновлювати їх, що робить додатки швидкими та продуктивними. React дозволяє розбивати інтерфейс на невеликі, повторно використовувані компоненти, що забезпечує масштабованість проекту з часом. Крім того, React має декларативний підхід до розробки, що робить код більш зрозумілим та підтримуваним. Взагалі,

використання React дозволяє розробникам швидко та ефективно створювати інтерактивні інтерфейси з високою продуктивністю та широким спектром функцій.

Дизайн сторінок було оформлення за допомогою мови стилів CSS. CSS є незамінним інструментом для web-розробки, оскільки він дозволяє розробникам стильово оформлювати web-сторінки, що забезпечує кращу читабельність та зручність користування. CSS дозволяє налаштувати різноманітні елементи на сторінці, такі як кольори, фони, шрифти та розміри, щоб забезпечити гармонійний дизайн та єдиний стиль на всій web-сторінці. CSS дозволяє також зменшити розмір коду сторінки, що допомагає збільшити швидкість завантаження сторінки та зменшити обсяг даних переданих клієнту. Крім того, CSS дозволяє розробникам створювати адаптивні web-сторінки, які підлаштовуються під різні пристрої та екрани, що забезпечує більшу зручність користування сайтом. Взагалі, використання CSS є важливою складовою для створення сучасних, естетично привабливих та функціональних web-сторінок.

Visual Studio Code є популярним та потужним редактором коду, який має велику кількість корисних функцій для розробників, таких як підсвічування синтаксису, автодоповнення коду, інтеграція з Git та різними іншими інструментами розробки. VS Code також підтримує різні мови програмування та має велику кількість розширень та налаштувань, що дозволяє розробникам налаштовувати його під свої потреби. Використання VS Code дозволяє підвищити продуктивність та зручність розробки web-додатків. Додатково, Visual Studio Code підтримує відладку коду та має інтегровану консоль для виконання команд. Крім того, він має можливість розробляти розширення за допомогою мови програмування JavaScript, що дозволяє розширити можливості редактора та налаштувати його для конкретних задач розробки. VS Code має інтегровану підтримку засобів для роботи з контейнерами та хмарами, що дозволяє легко налаштувати середовище розробки та виконувати розробку web-додатків на різних платформах. Також він має вбудовану систему керування пакетами та залежностями, що дозволяє швидко та зручно додавати нові бібліотеки та інструменти для розробки.

Отже, вибір мови програмування JavaScript та бібліотеки React, а також середовища розробки Visual Studio Code є обґрунтованим та дозволить створити якісний web-сервіс для створення плану харчування та тренування.

2.1.1 Visual Studio Code

Visual Studio Code (VS Code) - це безкоштовний відкритий редактор коду від Microsoft, який надає потужні інструменти для розробки різноманітних web-додатків. VS Code підтримує різні мови програмування, включаючи JavaScript, HTML, CSS, Python, Java та багато інших.

Одна з ключових переваг VS Code - це його зручність у використанні та конфігурації. Редактор має простий та лаконічний інтерфейс, що дозволяє швидко зорієнтуватись у функціоналі та почати розробку.

VS Code надає безліч корисних функцій для розробників, таких як підсвічування синтаксису, автодоповнення коду, відступи та вирівнювання, зміна регістру, відладка, інтеграція з системою керування версіями Git та багато інших. Редактор дозволяє зручно налаштовувати та персоналізувати його інтерфейс, налаштування, теми та розширення.

Цей редактор має вбудовані засоби відлагодження коду та підтримує різноманітні розширення, такі як Debugger for Chrome, Debugger for Node.js тощо. Також в ньому є інтегрована консоль, що дозволяє виконувати команди безпосередньо. Такі можливості забезпечують зручний та швидкий процес відлагодження та дебагування коду, роблячи цей редактор більш привабливим для розробників порівняно з іншими текстовими редакторами.

VS Code підтримує багато інші розширення, що можна легко встановлювати та налаштовувати через магазин розширень. Розширення дозволяють розширити можливості редактора та зручно працювати з різними мовами програмування та інструментами.

Крім того, VS Code має інтегровану систему керування пакетами та залежностями, яка дозволяє швидко та зручно додавати нові бібліотеки та

інструменти для розробки.

У цьому редакторі можна працювати з декількома файлами одночасно, використовуючи спеціальні вкладки. Редактор також має вбудовані засоби для роботи з терміналом та інтегрованою консоллю, що значно полегшує взаємодію з командним рядком. Крім того, VS Code надає зручні інструменти для налаштування та керування проектами, використовуючи різноманітні пакети та розширення. Всі ці можливості дозволяють зручно та швидко працювати з проектами, роблячи VS Code відмінним вибором для розробників.

VS Code може бути використаний для розробки різноманітних web-додатків, від простих статичних сайтів до складних web-прикладань. Редактор надає безліч корисних інструментів та функцій, які полегшують та прискорюють процес розробки.

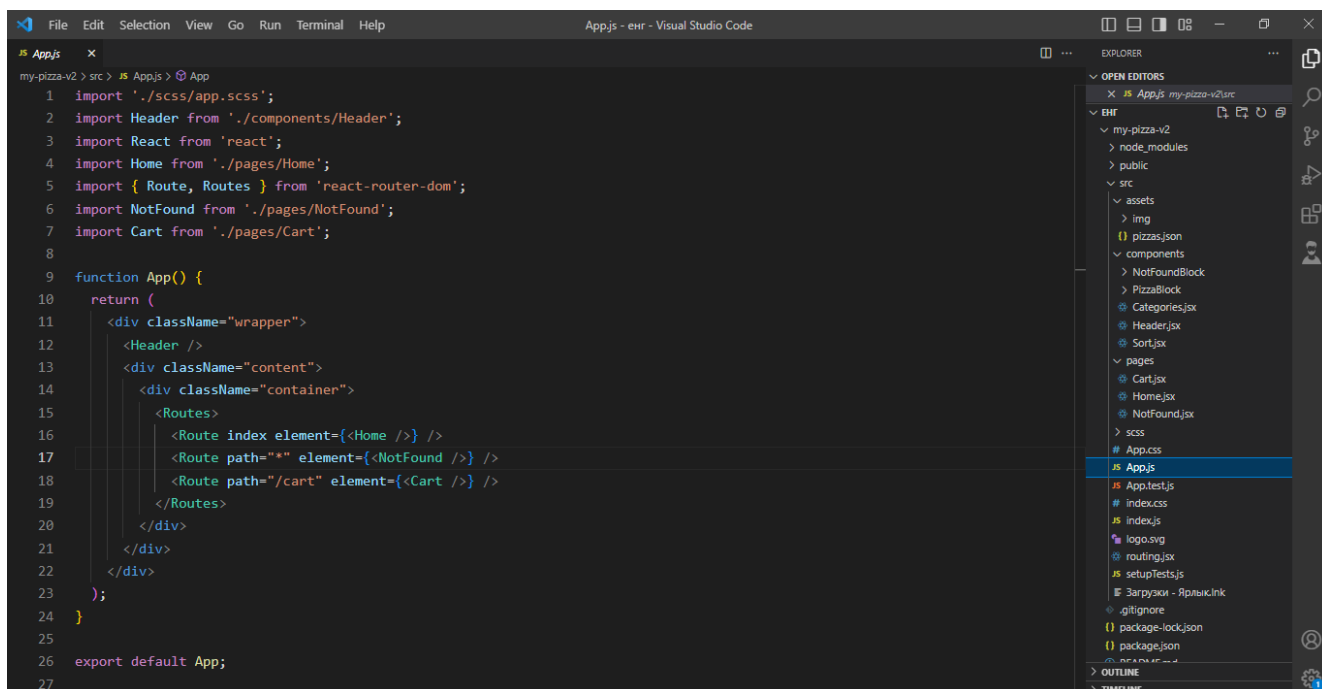


Рисунок 2.1 Середовище розробки Visual Studio Code

Visual Studio Code також має можливість розширення за допомогою пакетів та розширень, які можна встановлювати безпосередньо з редактора. Ці пакети та розширення дозволяють додавати нові функції та можливості до редактора, такі як

підтримка нових мов програмування, інтеграція зі сторонніми сервісами та бібліотеками, автоматичне завершення коду, рефакторинг та інші.

Крім того, VS Code має вбудований дебагер, який дозволяє відлагоджувати код безпосередньо з редактора. Дебагер підтримує різні мови програмування та забезпечує можливість пошуку та виправлення помилок в коді, що полегшує та прискорює процес розробки.

Додатково, цей редактор володіє вбудованим менеджером версій Git, що дозволяє легко керувати версіями проекту. Завдяки цьому інтегрованому рішенню, розробники можуть без зусиль здійснювати коміти, відкатувати зміни, створювати та злити гілки. Інструментарій Git у цьому редакторі надає зручні можливості для ефективного управління кодовою базою та збереження історії змін проекту.

У загальному, VS Code - це потужний та зручний редактор коду, який надає безліч корисних функцій та можливостей для розробки програмного забезпечення різного рівня складності. Дана інформація може бути корисною для тих, хто планує розпочати розробку програмного забезпечення та хоче знайти зручний та потужний інструмент для цього.

Одним з головних переваг VS Code є його гнучкість та можливості розширення. Редактор дозволяє легко додавати нові функції та можливості, які можуть бути корисними для розробників програмного забезпечення різного рівня складності. Це досягається за допомогою пакетів та розширень, які можна встановлювати безпосередньо з редактора.

Крім того, VS Code має широку підтримку різних мов програмування та фреймворків. Редактор підтримує багато мов програмування, включаючи JavaScript, Python, C#, Java, Ruby, PHP та інші. Також VS Code має спеціальні розширення для підтримки різних фреймворків, таких як React, Angular, Vue та інші.

Додатково, VS Code має вбудований дебагер, який дозволяє відлагоджувати код безпосередньо з редактора. Це дозволяє забезпечити більш швидкий та ефективний процес відлагодження коду.

Крім того, VS Code має вбудовану підтримку Git, що дозволяє легко керувати

версіями проекту та здійснювати коміти, відкати зміни, створювати та зливати гілки.

Загалом, VS Code є гнучким, потужним та зручним редактором коду з великою кількістю можливостей розширення та підтримки різних мов програмування та фреймворків. Він є відмінним вибором для розробників, які шукають зручний та потужний інструмент для розробки програмного забезпечення.

2.1.2 React

Для створення основного інтерфейсу користувача була використана бібліотека React.

React - це відкрита бібліотека JavaScript для розробки інтерфейсів користувача. Вона дозволяє створювати динамічні web-додатки, які реагують на зміни даних без перезавантаження сторінки. Основним принципом React є компонентний підхід, де весь інтерфейс розбивається на невеликі незалежні компоненти, які можуть бути повторно використані. React використовує віртуальний DOM для ефективного оновлення та маніпуляцій зі структурою сторінки. Це дозволяє розробникам швидко і зручно будувати складні інтерфейси, які є легкими для тестування та підтримки.

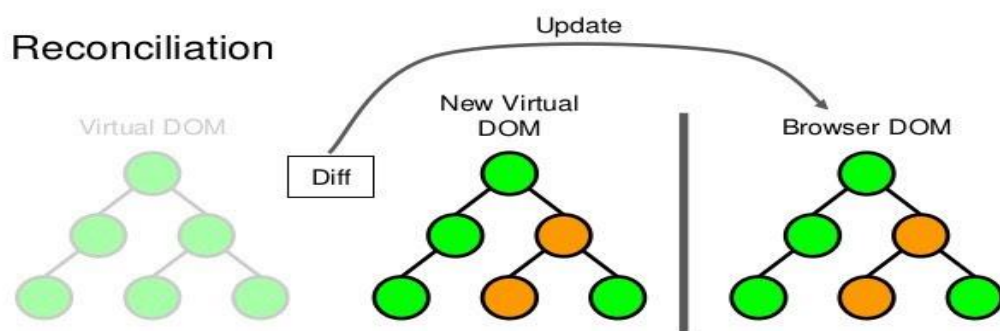


Рисунок 2.2. Принцип роботи Virtual DOM у React

React - це відкрита бібліотека JavaScript, розроблена командою Facebook, призначена для створення інтерфейсів користувача. Основною метою React є полегшення процесу розробки web-додатків, зокрема побудови динамічних інтерфейсів, які швидко реагують на зміни даних.

Одним з головних принципів React є компонентний підхід. Весь інтерфейс розбивається на незалежні компоненти, які можуть бути повторно використані і знаходитися в ієрархічних відношеннях. Кожен компонент відповідає за свою частину інтерфейсу і має власний стан (state) і властивості (props). Це спрощує організацію коду і дозволяє писати чистий, зрозумілий і легко супроводжуваний код.

Одна з головних переваг React - використання віртуального DOM (Document Object Model). Він представляє собою легку копію реального DOM, з яким React працює. Зміни в даних викликають оновлення віртуального DOM, а React знаходить оптимальний спосіб оновлення реального DOM на основі цих змін. Це дозволяє забезпечити швидкість та ефективність роботи додатків, особливо в умовах складних інтерфейсів з багатьма елементами.

Крім того, React має багатий екосистему з розширеннями та допоміжними бібліотеками, які допомагають в розробці. Наприклад, React Router дозволяє легко управляти маршрутизацією, Redux - для керування станом додатка, а Material-UI - для швидкого створення стильного інтерфейсу.

React активно використовується в розробці великих web-додатків, таких як Facebook, Instagram, Netflix, Airbnb, і багатьох інших. Його популярність зумовлена простотою використання, широкою спільнотою розробників та високою продуктивністю.

React має декілька переваг порівняно з іншими бібліотеками та фреймворками. Ось докладний огляд його переваг:

Компонентний підхід: React базується на компонентах, що дозволяє розбити інтерфейс на невеликі, самодостатні та повторно використовувані блоки. Це спрощує розробку, тестування та супроводження коду, а також дозволяє зберігати його логічну структуру.

Віртуальний DOM: React використовує віртуальний DOM, що дає йому високу продуктивність. Замість того, щоб безпосередньо оновлювати реальний DOM, React працює з його легкою копією у пам'яті, а потім ефективно оновлює лише необхідні елементи. Це забезпечує швидке реагування на зміни даних та оптимізацію продуктивності.

Односторінкові додатки: React добре підходить для розробки односторінкових додатків, де інтерфейс оновлюється без перезавантаження сторінки. Він дозволяє плавну навігацію, динамічні оновлення та керування станом додатка, що покращує користувацький досвід.

Розширюваність: React є дуже розширюваним. Він має велику кількість розширень (пакетів), які допомагають збільшити функціональність та розширити можливості React. Також, завдяки широкій спільноті розробників, завжди є відповідні ресурси, документація та підтримка для будь-яких потреб.

Спрощена робота зі станом: React має чіткий та простий підхід до керування станом додатка. З використанням стейтів (state) та властивостей (props), розробники можуть ефективно відстежувати та змінювати стан компонентів. Це робить управління даними більш прозорим та простим.

Широке сприйняття на ринку: React є однією з найпопулярніших бібліотек для розробки web-додатків. Його використовують великі компанії, такі як Facebook, Instagram, Netflix, Airbnb і багато інших. Це означає, що є велика спільнота розробників, яка сприяє обміну знаннями, підтримці та розвитку екосистеми React.

В цілому, React надає потужні інструменти для швидкого та ефективного розробки web-додатків. Він поєднує в собі простоту, швидкість та розширюваність, роблячи його одним з найпопулярніших виборів для сучасних web-розробок.

Встановлення та конфігурація React

Спочатку потрібно встановити Node.js, якщо він ще не встановлений на комп'ютері. Node.js включає в себе пакетний менеджер npm, який потрібен для управління залежностями проекту.

Потрібно створити новий проект React за допомогою Create React App. Щоб

це зробити потрібно відкрити командний рядок або термінал і ввести команду `prx create-react-app my-app`, де `my-app` - це назва вашого проекту.

Після успішного створення проекту перейдіть до його папки, використовуючи команду `cd my-app`.

Запустіть проект, введіть команду `npm start`. Це запустить розробний сервер, і ваш проект буде доступний за адресою <http://localhost:3000>.

Файли проекту можна редагувати в текстовому редакторі, такому як Visual Studio Code. Зміни будуть автоматично підтягуватися на локальний сервер і відображатися в браузері.

Додайте необхідні бібліотеки до проекту, використовуючи команду `npm install package-name`. Наприклад, щоб використовувати бібліотеку React Router, її потрібно встановити за допомогою команди `npm install react-router-dom`.

Використовуйте створений проект для створення компонентів, налаштування маршрутизації, додавання стилів та логіки за допомогою React. За допомогою цих кроків можна швидко встановити та налаштувати проект React і почати його розробку.

2.1.3 JSX

JSX (JavaScript XML) - це розширення синтаксису JavaScript, яке дозволяє писати HTML-подібний код прямо в JavaScript файлів, що використовуються в React-проектах. Він дозволяє зручно описувати структуру та вигляд компонентів.

Основні особливості JSX:

Синтаксис схожий на HTML: JSX виглядає дуже схожим на HTML, що робить код більш зрозумілим і зрозумілим для розробників, знайомих з HTML.

Використання компонентів: JSX дозволяє створювати та використовувати власні компоненти. Можна описати компонент як функцію або клас і використовувати його як будь-який інший HTML-елемент.

Вставка JavaScript-коду: JSX дозволяє вставляти JavaScript-код безпосередньо в середину розмітки. Можна використовувати фігурні дужки `{}` для

вставки змінних, виразів та функцій.

Робота з атрибутами: Атрибути HTML-елементів можуть бути використані в JSX так само, як у звичайному HTML, з деякими відмінностями. Використовуючи фігурні дужки, можна вставляти значення JavaScript в атрибути.

Рендеринг списків: JSX дозволяє легко рендерити списки елементів за допомогою методу `map`. Можна створити масив JSX-елементів, які представляють кожен елемент списку.

Умовний рендеринг: За допомогою JavaScript-виразів, можна визначати умови для рендерингу різних частин JSX залежно від певних умов або значень змінних.

Використання CSS-класів: JSX дозволяє використовувати CSS-класи для стилізації елементів. Можна використовувати ключове слово `className` замість `class` для встановлення CSS-класів.

Синтаксис JSX.

Основні правила синтаксису JSX:

Елементи: можна створювати елементи, що відповідають HTML-тегам, за допомогою тегового синтаксису, наприклад `<div>`, ``, `<h1>`, тощо.

Атрибути: можна встановлювати атрибути HTML-елементів, використовуючи синтаксис `атрибут="значення"`. Наприклад, `<input type="text" />` встановлює атрибут `type` зі значенням `"text"` для елемента `<input>`.

Вставка JavaScript-коду: можна вставляти JavaScript-код безпосередньо в середину розмітки JSX, використовуючи фігурні дужки `{}`. Наприклад, `<p>{message}</p>` вставляє значення змінної `message` внутрішньо HTML-елемента `<p>`.

Використання виразів: можна використовувати JavaScript-вирази в фігурних дужках `{}` для обчислення значень атрибутів, стилів та іншого. Наприклад, `<div className={isActive ? 'active' : 'inactive'}></div>` використовує тернарний оператор для визначення значення атрибуту `className`.

Компоненти: можна використовувати React-компоненти, що визначені як функції або класи, в JSX. Наприклад, `<Header />` використовує компонент

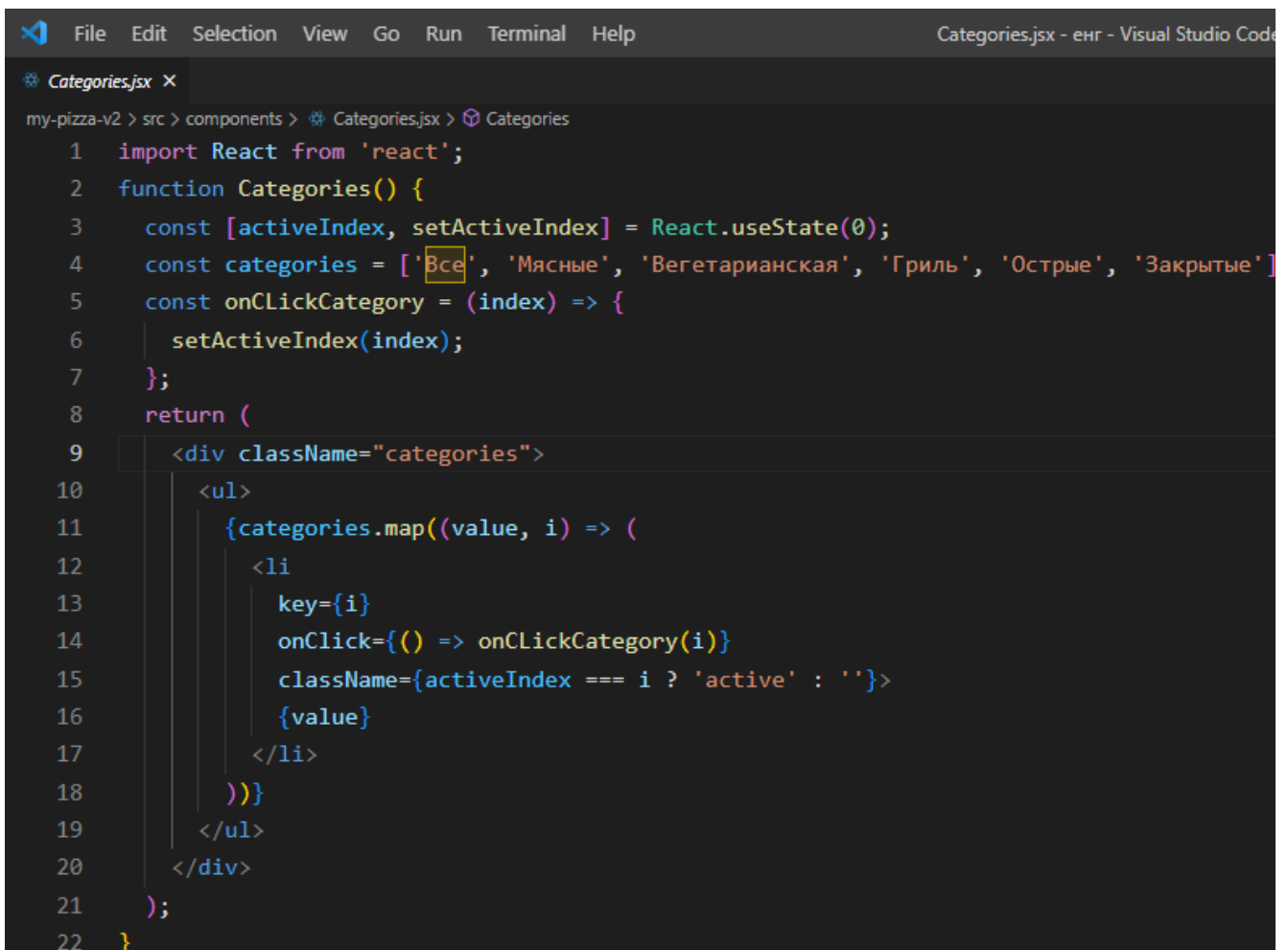
<Header>, що відображає заголовок.

Вкладеність: JSX підтримує вкладеність елементів, тобто можна розміщувати один елемент всередині іншого. Наприклад,

<div><h1>Title</h1><p>Paragraph</p></div> містить <h1> та <p> елементи всередині <div>.

Коментарі: можна додавати коментарі в JSX, використовуючи синтаксис `{/* коментар */}`. Коментарі не відображаються на вихідному HTML, вони використовуються лише для пояснення коду.

JSX дозволяє зробити код React більш зрозумілим та легким для читання, оскільки він поєднує логіку компонентів та розмітку в одному місці.



```

1  import React from 'react';
2  function Categories() {
3    const [activeIndex, setActiveIndex] = React.useState(0);
4    const categories = ['Все', 'Мясные', 'Вегетарианская', 'Гриль', 'Острые', 'Закрытые'];
5    const onClickCategory = (index) => {
6      setActiveIndex(index);
7    };
8    return (
9      <div className="categories">
10     <ul>
11       {categories.map((value, i) => (
12         <li
13           key={i}
14           onClick={() => onClickCategory(i)}
15           className={activeIndex === i ? 'active' : ''}>
16           {value}
17         </li>
18       ))}
19     </ul>
20   </div>
21 );
22 }

```

Рисунок 2.2. Приклад синтаксису JSX

Переваги використання JSX.

Основними перевагами використання JSX є:

Зрозумілість і читабельність коду: JSX дозволяє писати розмітку прямо в JavaScript-коді, що спрощує розуміння структури компонентів та їх взаємодії. Це дозволяє розробникам легше читати, редагувати та відлагоджувати код.

Єдність логіки та представлення: JSX дозволяє об'єднати логіку компонентів і їх представлення в одному файлі, що полегшує розробку і підтримку коду. Розмітка та динамічні дані можуть бути вбудовані безпосередньо в компоненти, що робить код більш структурованим і зрозумілим.

Висока розширюваність: JSX підтримує використання JavaScript-коду безпосередньо всередині розмітки, що робить його дуже гнучким та розширюваним. Розробники можуть використовувати всі можливості JavaScript, включаючи змінні, функції, умови та цикли, для створення динамічної та складної логіки.

Ефективна робота зі стилями: JSX дозволяє використовувати CSS-класи для стилізації елементів, що спрощує роботу зі стилями та додає читабельності до коду. Використання CSS-класів дозволяє легко змінювати стилі компонентів без втручання в логіку.

Переваги React екосистеми: JSX є нативним для React, тому використання його дозволяє використовувати всю потужність та переваги React екосистеми, включаючи популярні бібліотеки, розширення та інструменти для розробки. В цілому, використання JSX допомагає зробити розробку.

React додатків більш простою, зрозумілою та ефективною, забезпечуючи зручний спосіб поєднання логіки та представлення в одному місці.

2.1.4 Firebase

Firebase - це платформа для розробки мобільних та web-додатків, розроблена компанією Google. Вона надає набір інструментів та сервісів для реалізації функціональності додатків, зокрема зберігання даних в хмарі, аутентифікацію

користувачів, реал-тайм синхронізацію даних, розгортання та багато іншого. Firebase може бути дуже корисним у створенні web-сайтів та web-додатків. Ось кілька способів, які допомагають використовувати Firebase у цьому проєкті:

а) Зберігання та синхронізація даних: Firebase надає базу даних Firestore і реального часу базу даних, що дозволяють зберігати та синхронізувати дані між клієнтами та сервером. Це дозволяє створювати web-додатки, які оновлюються в реальному часі, а також дозволяє працювати з даними офлайн.

б) Аутентифікація користувачів: Firebase має вбудовану систему аутентифікації, яка дозволяє легко додати реєстрацію та вхід в систему для користувачів вашого web-сайту. можна налаштувати аутентифікацію через електронну пошту, соціальні мережі, номер телефону та інші методи.

в) Хостинг та розгортання: Firebase має сервіс хостингу, який дозволяє легко розгорнути ваш web-сайт або web-додаток. можна швидко публікувати свій проєкт та забезпечити швидкий доступ до нього з будь-якого пристрою.

г) Хмарне сховище: Firebase надає сервіс зберігання файлів в хмарі. можна завантажувати та зберігати файли, такі як зображення, відео, документи тощо, і отримувати до них доступ з вашого web-сайту або web-додатку.

г) Аналітика та звіти: Firebase має вбудований інструмент аналітики, який надає статистику про використання вашого web-сайту або web-додатку. можна отримувати дані про активність користувачів, конверсії, розподіл платформ та багато іншого.

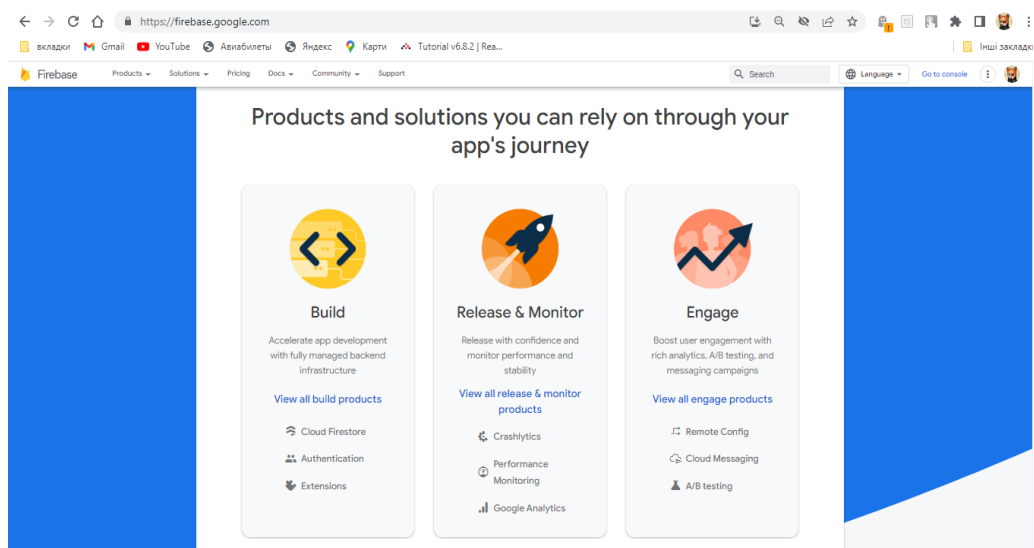


Рисунок 2.3 Головна сторінка сервісу Firebase.

Firebase має можливості інтеграції з іншими сервісами Google, що дозволяє розширити функціональність вашого web-сайту або web-додатку. Наприклад, можна використовувати Google Cloud Functions для написання та розгортання власних серверних функцій, які можуть взаємодіяти з Firebase. Це дозволяє реалізувати додаткову бізнес-логіку або обробку даних на стороні сервера. Крім того, можна інтегрувати Firebase з Google Analytics для отримання додаткових аналітичних можливостей і детального відстеження поведінки користувачів. Також Firebase надає інтеграцію з Google Ads, що дозволяє впроваджувати рекламні кампанії та відстежувати їх ефективність безпосередньо з Firebase Console. Інтеграція з іншими сервісами Google допомагає використовувати різноманітні інструменти та можливості, що пропонуються екосистемою Google, для покращення вашого web-проекту.

Firebase надає багато інструментів та сервісів, які спрощують розробку web-сайтів та web-додатків, забезпечуючи готові рішення для багатьох типових задач. Він підтримує роботу з різними технологіями, такими як JavaScript, React, Vue.js, Angular та інші, що робить його популярним вибором серед розробників.

Firebase є потужною та привабливою платформою для розробки web-сайтів та web-додатків. Вона відзначається простотою використання, гнучкістю в підтримці різних технологій та надає готові рішення для багатьох задач. Це робить Firebase прекрасним вибором для розробників, які шукають швидку і ефективну платформу для розробки та розгортання своїх проектів.

2.2 Загальний алгоритм реалізації проекту

Створення фітнес web-сервісу для створення плану харчування та тренування людини - це комплексна задача, яка потребує великої кількості роботи та підходів. Фітнес web-сервіс має допомогти людям котрі бажають займатися спортом, та привести своє фізичне здоров'я та тіло в порядок. У цьому пункті описано кроки, які були виконані при створенні web сервісу, включаючи розробку макету у Figma та розробку в середовищі Visual Studio Code.

Визначення мети та цілей створення web-сервісу

Перший крок у створенні web-сервісу – визначення мети та ціле цього самого web-сервісу. Метою створення такого web-сервісу було допомогти людям, що бажають почати або уже доглядають за своїм фізичним здоров'ям, створити для себе план харчування та тренування. Цілі створення встановлювалися з урахуванням вимог та потреб сучасного ринку web-сервісів. Визначення мети та цілей створення, дозволило зосередитися на головних завданнях та визначити стратегію створення web-сервісу.

Розробка змісту web-сервісу

Другий крок - розробка змісту web-сервісу. Зміст web-сервісу було створено з урахуванням різноманітних особливостей використання подібних web-сайтів, що дозволить знизити поріг мінімальних знань для початку користування web-сервісом та сприяє ефективному користуванню з ціллю створити план харчування та тренування.

Розробка макету у Figma

Третій крок - розробка макету у Figma. Це дозволило створити візуальну концепцію web-сайту, побачити його у вигляді макету та внести необхідні зміни перед тим, як розробити фінальний продукт. Застосування Figma дозволило більш ефективно організувати роботу над фітнес web-сервісом та зробити його більш доступним та зрозумілим для користувачів. Розробка макету у Figma дозволила покращити візуальну привабливість та зрозумілість web-сервісу.

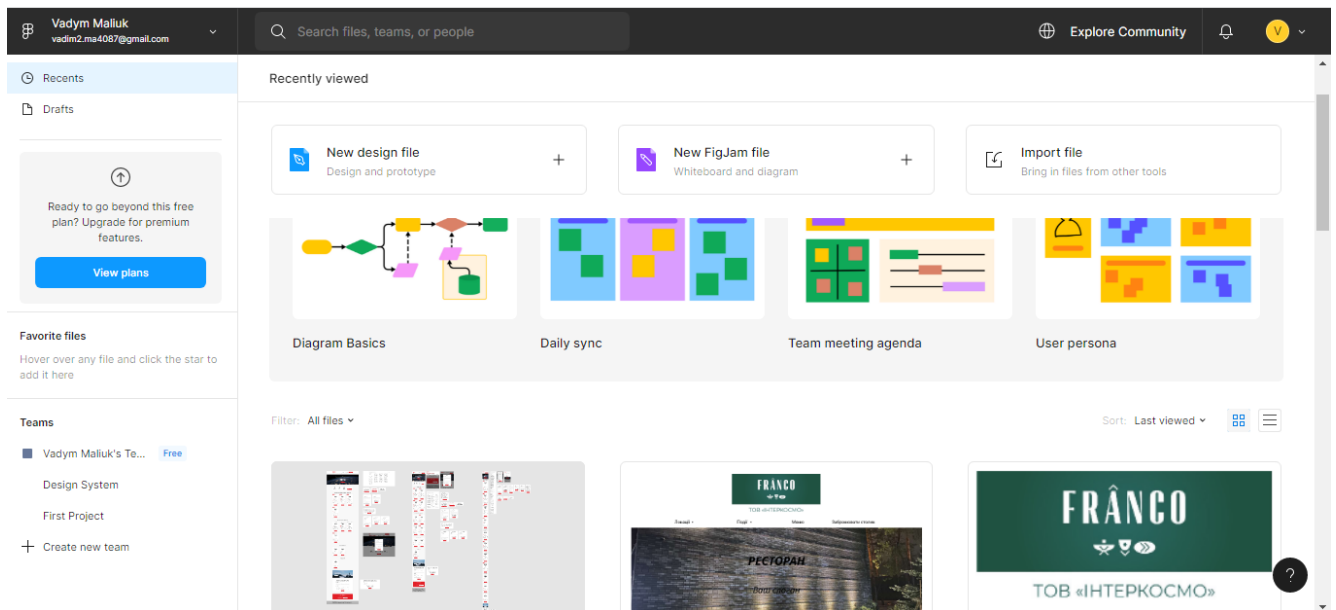


Рисунок 3.3. Макетування у Figma.

Проектування системи

Четвертий крок - проектування системи. Для створення web-сервісу, був використаний підхід user-centered design, що дозволив зосередитися на потребах та очікуваннях користувачів та забезпечити їх задоволеність від користування web-сервісом.

Розроблена діаграма User flow дозволила проаналізувати та покращити процес взаємодії користувачів з web-сервісом, забезпечивши максимальну зручність та ефективність використання. В результаті, користувачі мають можливість легко пересуватися по сторінкам, ознайомлюватися з цікавою та корисною інформацією і звісно ж перейти до створення індивідуального плану харчування та тренування. Діаграма є важливою складовою в розробці будь-якого продукту, оскільки вона дозволяє оцінити ефективність взаємодії з користувачами та зробити відповідні покращення, що забезпечують кращу якість взаємодії з продуктом. Крім того, це дозволило ідентифікувати можливі проблеми з користуванням та вирішити їх ще на етапі проектування web-сервісу.

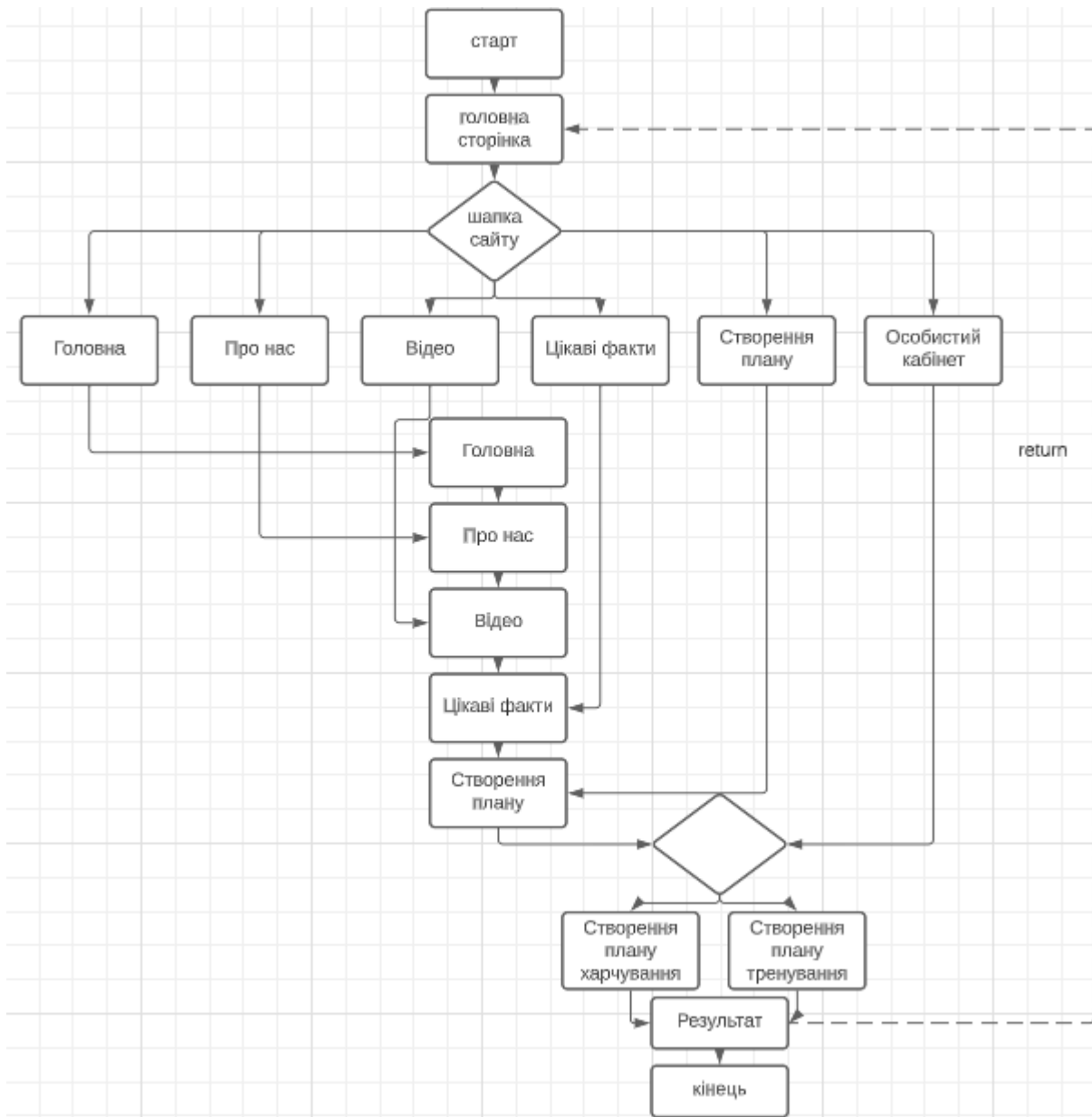


Рисунок 3.4. Діаграма «User flow».

Розробка в середовищі Visual Studio Code

П'ятий крок - розробка в середовищі VS Code. Для цього були використані JavaScript, бібліотека React та JSX і CSS. Ця мова програмування є однією з найпоширеніших у світі, яка використовується для розробки програмного забезпечення, web-сайтів та web-додатків різної складності. Використання JavaScript для створення фітнес web-сервісу для створення плану харчування та тренування має кілька переваг.

По-перше, JavaScript є мовою програмування, яка має високу стабільність та надійність, що важливо для розробки будь-якого web-сайту чи web-додатку. Вона дозволяє розробникам створювати різну логіку використання web-сайту чи використання конкретної його функції.

По-друге, JS підтримується всіма основними web-браузерами, що робить його доступним для широкого кола користувачів.

По-третє, JS дозволяє створювати динамічні елементи, які можуть змінюватись залежно від дій користувача, такі як ефекти наведення курсора, анімації, графіки та багато іншого.

Загалом, виконання цих дій допомогло створити якісний web-сервіс, який дозволить користувачам почати доглядати за своїм фізичним станом та створювати для себе план харчування або тренування. За допомогою таких планів, будь-яка людина може значно покращити своє фізичне здоров'я, та відкрити для себе щось нове в плані духовного розвитку. Крім того, такий web-сервіс буде корисним не тільки для користувачів-початківців, але й для досвідчених, які хочуть покращити свій фізичний стан.

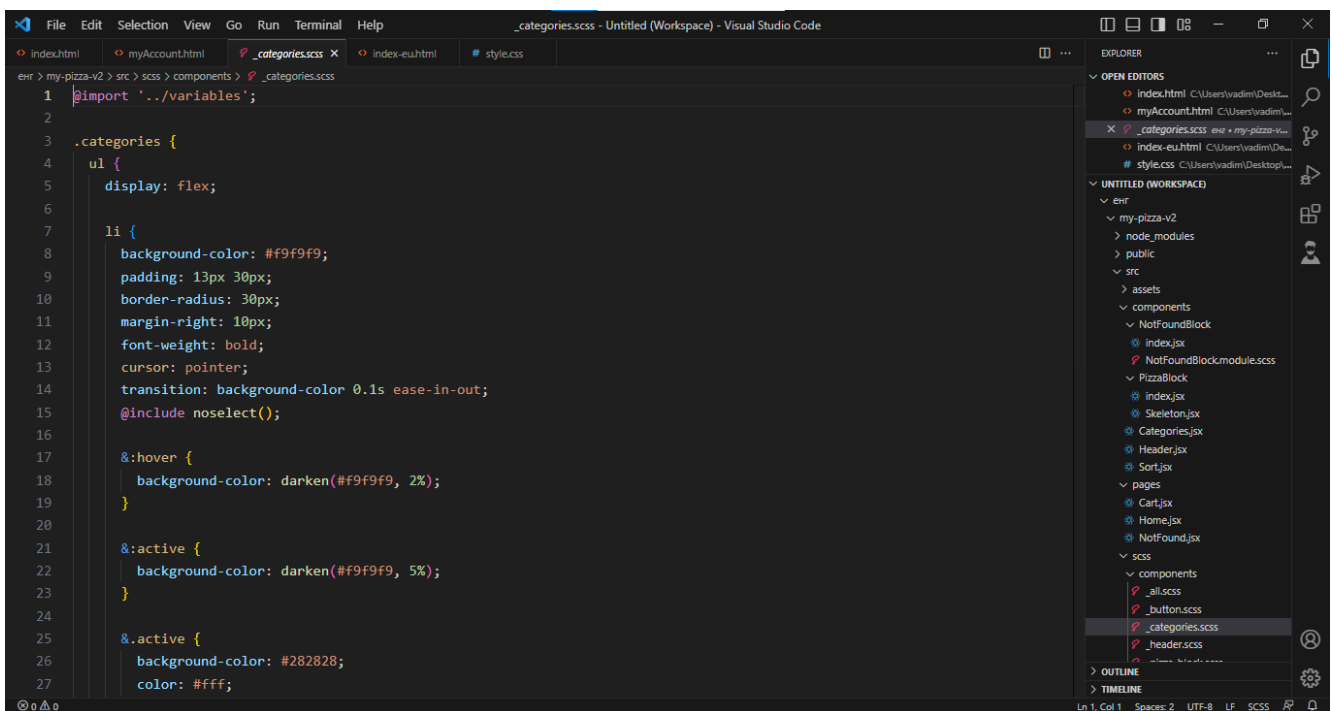


Рисунок 3.5. Процес розробки в середовищі Visual Studio Code

2.3 Функціонал продукту

Продукт розробки являє собою фітнес web-сервіс. Основна його мета – надати можливість для користувачів створювати для себе план тренування або харчування, для успішної підтримки або покращення свого здоров'я.

Отже, основні вимоги до продукту є такими:

- основна мета web-сервісу – створення плану тренування та харчування людини відповідно до його характеристик та цілей;
- збереження результатів розрахунку добової кількості калорій, жирів, білків та вуглеводів
- простий у користуванні інтерфейс, який буде зрозумілим для користувачів різних вікових категорій;
- теоретичні матеріали у вигляді цікавих фактів та тематичних відеороликів на тему фітнес тренувань та підтримки фізичного здоров'я.

В першу чергу реалізуються саме ці функції продукту, так як є основними реалізаціями критеріїв створення фітнес web-сервісу.

3. ПРОГРАМНА РЕАЛІЗАЦІЯ ФІТНЕС WEB-SERVICES

3.1 Оцінка та планування розробки фітнес web-сервісу

У процесі розробки продукту, після визначення основних вимог та архітектури системи, необхідно було скласти список задач для реалізації Мінімальної життєздатної версії (MVP). Важливим фактором при розробці MVP є обмежений час, який потрібно використовувати максимально ефективно та задовольнити найбільшу частину вимог продукту.

Тому, для забезпечення ефективної та успішної розробки MVP, було вирішено провести оцінку розробки з акцентом на принцип "головні функції спочатку" (KFF - Key Features First), який є одним з ключових положень підходу до розробки MVP. Використання техніки Experience Based для оцінки часу на розробку, що базується на минулому досвіді розробника та його рівні знань, також допомагає забезпечити більш точну та реалістичну оцінку розробки.

	Функціональні вимоги
1	Розробити алгоритм, за допомогою якого буде розраховуватись необхідна кількість добових калорій, білків, жирів та вуглеводів.
2	Надати можливість авторизації користувача в особистому кабінеті за допомогою гугл-сервісу.
3	Надати можливість збереження результатів обчислення добової кількості калорій, білків, жирів та вуглеводів.

Рисунок 3.1. Формування функціональних вимог.

Кожна вимога була описана в стилі use-case, що дозволяє застосовувати підхід BDD (Behavioral Driven Development), орієнтований на розробку функціоналу з фокусом на сценаріях використання та поведінці системи. Такий опис дозволяє збільшити ефективність розробки та уникнути неправильного розуміння вимог.

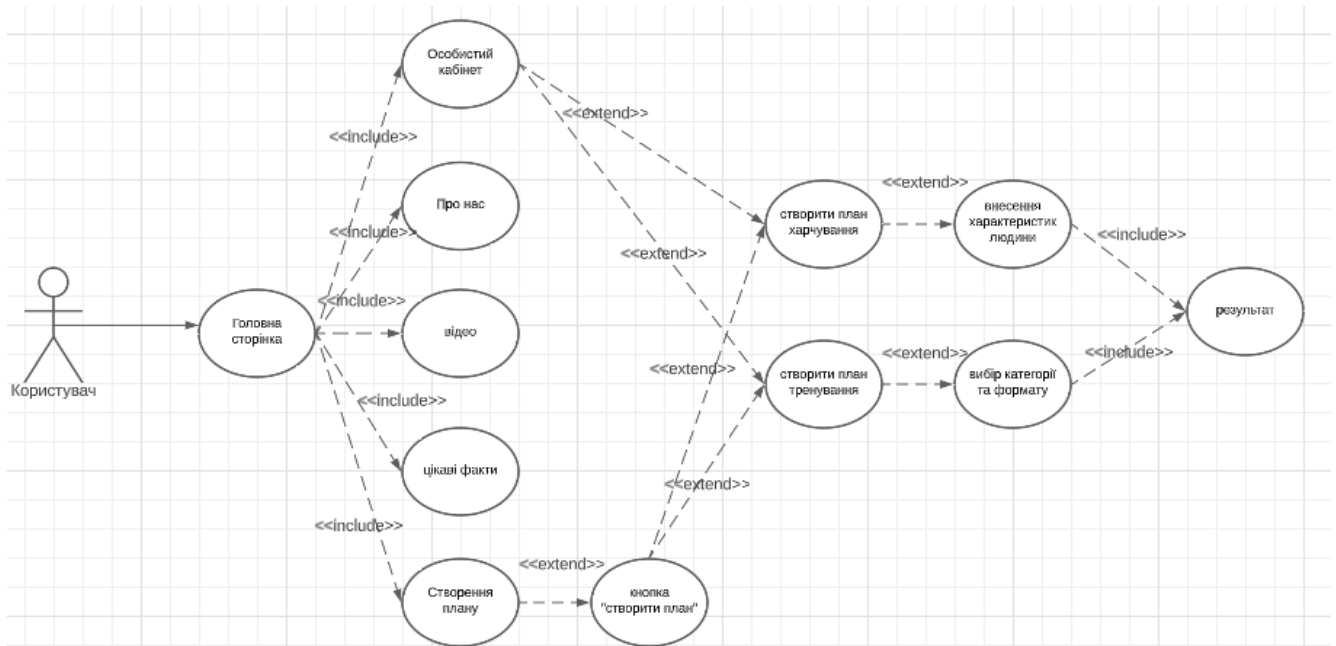


Рисунок 3.2. Use case діаграма.

У цілому, використання технік KFF, Experience Based, ABC-аналізу, а також стилів use-case та BDD допомогли забезпечити ефективну розробку MVP, відповідно до вимог продукту та умов, які були на той момент. Такий підхід дозволяє забезпечити максимальну ефективність розробки, уникнути помилок та забезпечити високу якість продукту.

При безпосередньому переміщенні задач до спринту вони розбивалися на окремі технічні підзадачі, які оцінювалися та реалізовувалися поступово.

3.2 Функціонал web-сервісу.

Основним функціоналом web-сервісу є наступні елементи:

- Створення плану харчування шляхом підбору потрібного меню за кількістю калорій.
- Створення плану тренування шляхом вибору конкретної категорії вправ
- Збереження результатів розрахунку в особистому кабінеті

Для того, щоб детальніше ознайомитися із web-сервісом, потрібно пройти повний шлях користування ним. При відкритті головної сторінки, відображаються панель навігації по сайту. Завдяки ним здійснюється уся навігація по сторінці та між ними. На головній сторінці користувач має змогу ознайомитися із загальною інформацією про фітнес, цікавими фактами та рекомендаціями по заняттю фітнесом.

Головна сторінка представляє собою header де є логотип сервісу, навігація по сторінці та кнопка переходу в особистий кабінет, далі вже у першому блоці сторінки буде інформація про назву сервісу та його короткий опис. Другий блок тримає в собі інформацію про web-сервіс. Третій блок це ознайомчі відео з рекомендаціями для користувачів які прагнуть займатися фітнесом для покращення свого фізичного стану. Четвертий блок містить цікаві факти про фітнес. П'ятий блок це кнопка переходу до створення плану харчування або тренування. У футері є посилання на соціальні мережі.

Інша сторінка, тобто сторінка особистого кабінету представляє собою header, footer та дві кнопки, створення плану харчування або створення плану тренування.

Якщо користувач вибирає створення плану харчування, у нього з'являються додаткові поля які потрібно заповнити.

The image shows a dark-themed user interface for a calorie calculator. It is divided into several sections:

- Gender Selection:** A section titled "ОБЕРІТЬ СТАТЬ" (Choose Gender) with two options: "ЧОЛОВІК" (Male) in red and "ЖІНКА" (Female) in white.
- Data Entry:** A section titled "ЗАПОВНІТЬ ДАНІ" (Fill in Data) with three input fields: "ВІК" (Age), "РІСТ СМ" (Height in cm), and "ВАГА КГ" (Weight in kg).
- Activity Level:** Three radio button options: "МАЛА АКТИВНІСТЬ" (Low activity), "СЕРЕДНЯ АКТИВНІСТЬ" (Medium activity), and "ВИСОКА АКТИВНІСТЬ" (High activity).
- Weight Goal:** Three radio button options: "СКИНУТИ ВАГУ" (Lose weight), "НАБРАТИ ВАГУ" (Gain weight), and "ВТРИМАТИ ВАГУ" (Maintain weight).
- Action:** A large red button at the bottom labeled "РОЗРАХУВАТИ" (Calculate).

Рисунок 3.6. Характеристики для розрахунку потрібної кількості калорій

Після того як користувач ввів свої характеристики, та натиснув кнопку “розрахувати” йому буде показано необхідну кількість його добових калорій, враховуючи його особисті потреби та запропоновано інструмент, за допомогою якого він може створити власне меню харчування просто ввівши потрібну йому кількість калорій, яку він тільки що дізнався. Розрахунок потрібної кількості калорій, білків, жирів та вуглеводів відбувається за допомогою цього коду:

```
const [trenChecked, setTrenChecked] = useState(false);
const [trenChecked2, setTrenChecked2] = useState(false);
const [category1Checked, setCategory1Checked] = useState(false);
const [category2Checked, setCategory2Checked] = useState(false);
const [category3Checked, setCategory3Checked] = useState(false);
const [category4Checked, setCategory4Checked] = useState(false);
```

```
const [videoChecked, setVideoChecked] = useState(false);
const [textChecked, setTextChecked] = useState(false);
const [sexFemaleChecked, setSexFemaleChecked] = useState(false);
const [sexMaleChecked, setSexMaleChecked] = useState(false);
const [ageValue, setAgeValue] = useState(NaN);
const [heightValue, setHeightValue] = useState(NaN);
const [weightValue, setWeightValue] = useState(NaN);
const [actSChecked, setActSChecked] = useState(false);
const [actMChecked, setActMChecked] = useState(false);
const [actLChecked, setActLChecked] = useState(false);
const [lostWChecked, setLostWChecked] = useState(false);
const [plusWChecked, setPlusWChecked] = useState(false);
const [stopWChecked, setStopWChecked] = useState(false);
```

```
const [resu, setResu] = useState();
const [prot, setProt] = useState();
const [fat, setFat] = useState();
const [carb, setCarb] = useState();
```

```
const [result, setResult] = useState(false);
const [resetData, setResetData] = useState(false);
const [tempUuid, setTempUuid] = useState([]);
```

```
const resetResult = () => {
  setSexFemaleChecked(false);
  setSexMaleChecked(false);
  setHeightValue(NaN);
  setAgeValue(NaN);
  setWeightValue(NaN);
  setActSChecked(false);
  setActMChecked(false);
```

```

setActLChecked(false);
setLostWChecked(false);
setPlusWChecked(false);
setStopWChecked(false);

setResetData(false);
};
const changeCheckbox = () => {
  if (!trenChecked2) {
    setTrenChecked(false);
    setTrenChecked2(true);
  }
};
const changeCheckbox2 = () => {
  if (!trenChecked) {
    setTrenChecked(true);
    setTrenChecked2(false);
  }
};

const resultDiet = () => {
  setResu(false);
  setResult(true);
  sexFemaleChecked ? setSexFemaleChecked(447.6) : setSexMaleChecked(88.36);
  if (actSChecked) {
    setActSChecked(1.2);
  }
  if (actMChecked) {
    setActSChecked(1.5);
  }
  if (actLChecked) {

```



```

    setActSChecked(1.8);
}

let weightMain, res;

if (lostWChecked) {
    weightMain = -500;
} else if (plusWChecked) {
    weightMain = 500;
} else {
    weightMain = 0;
}

if (sexFemaleChecked) {
    res =
        (sexFemaleChecked + heightValue * 3.1 + weightValue * 9.2 - ageValue * 4.3) *
        (actSChecked + actMChecked + actLChecked) +
        weightMain;
} else {
    res =
        (sexMaleChecked + heightValue * 4.8 + weightValue * 13.4 - ageValue * 5.7) *
        (actSChecked + actMChecked + actLChecked) +
        weightMain;
}

let protG = (Math.round(res) * 0.3) / 4;
let fatG = (Math.round(res) * 0.3) / 9;
let carbG = (Math.round(res) * 0.4) / 4;

setResu(Math.round(res));
setProt(Math.round(protG));
setFat(Math.round(fatG));
setCarb(Math.round(carbG));

```

```
setActSChecked(0);
setResetData(true);

const uuid = uid();
set(ref(database, `/${uuid}`), {
  res,
  protG,
  fatG,
  carbG,
  uuid,
});
```

ПОТРІБНА КІЛЬКІСТЬ КАЛОРІЙ:	3363
ПОТРІБНА КІЛЬКІСТЬ БІЛКІВ:	252
ПОТРІБНА КІЛЬКІСТЬ ЖИРІВ:	112
ПОТРІБНА КІЛЬКІСТЬ ВУГЛЕВОДІВ:	336
РЕКОМЕНДОВАНИЙ ІНСТРУМЕНТ ДЛЯ СТВОРЕННЯ МЕНЮ ХАРЧУВАННЯ:	СТВОРИТИ МЕНЮ

Рисунок 3.6. Результат розрахунку потрібної кількості добових калорій

Після того як користувач дізнався потрібну кількість добових калорій, він може перейти до створення меню харчування на день за цією кількістю калорій

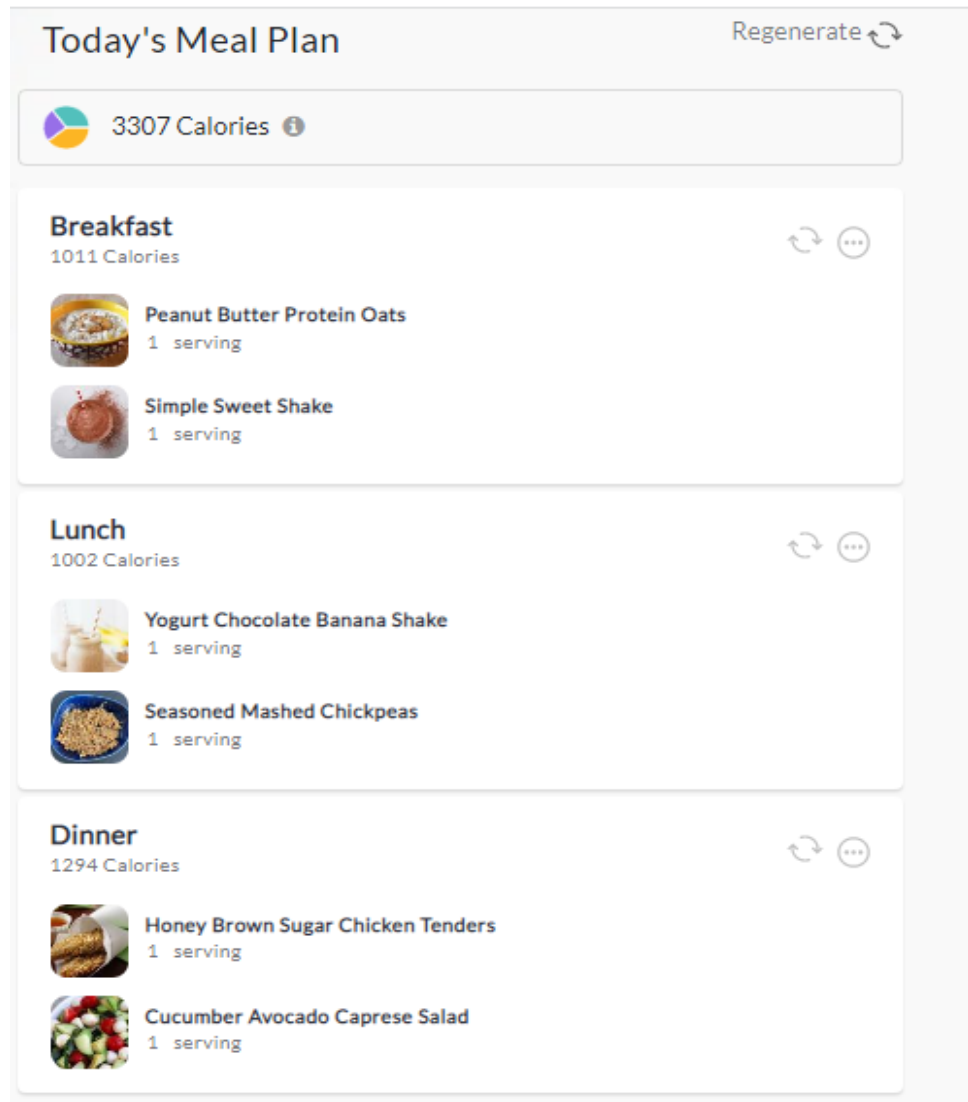


Рисунок 3.7. Результат розрахунку меню харчування за к-стю калорій

Якщо користувач хоче підібрати вправи для тренування, то в такому випадку він натискає кнопку “для тренування”. Там буде короткий опис чотирьох основних категорій вправ для різних типів м’язів та цілей.

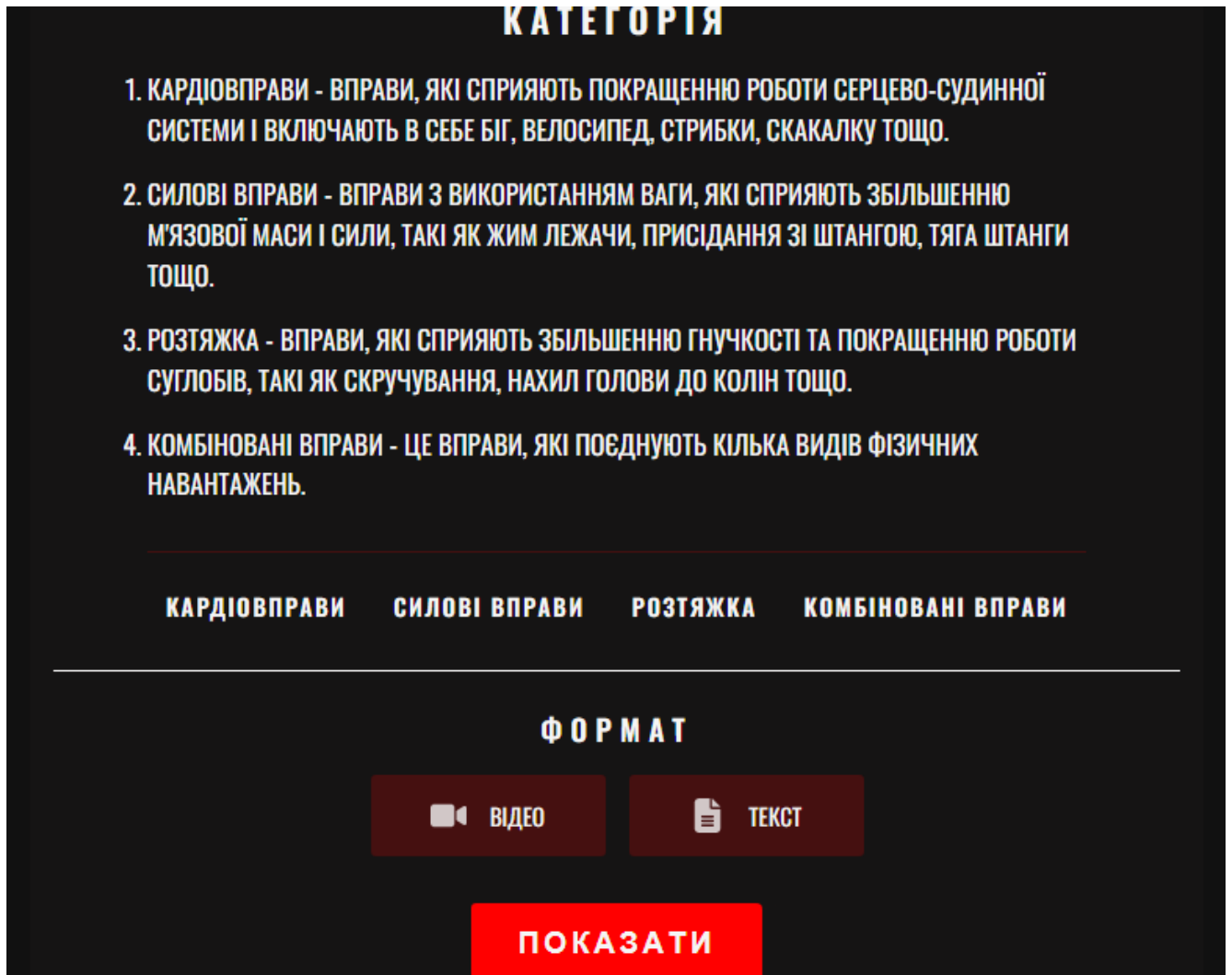


Рисунок 3.8. Категорії вправ які доступні на вибір

Після того, як користувач обрав потрібну йому категорію та формат у якому він бажає отримати вправи, він натискає кнопку "Показати" та отримує результат в залежності від того, що він обрав.

7. Бокові скручування: лежачи на боці, піднімаємо корпус вгору, тримаючи руки вздовж тіла. Вправу потрібно виконувати по 10-15 разів на кожному боці. Вправа "кішка-корова": стаємо на коліна і руки, при цьому долоні встановлюємо на підлогу відповідно до плечей. Починаємо виконувати вправу: спочатку вигинаємо спину, намагаючись дотягнутися головою до стегон, потім повільно опускаємо голову і піднімаємо хребет вгору, вигинаючи його в напрямку стелі. Вправу потрібно виконувати по 10-15 разів.
8. Випади: Станьте на одну ногу, в другій тримайте гантель або водну пляшку. Зробіть крок вперед і зігніть коліно, так щоб кут між стегном і гомілкою склався приблизно під кутом 90 градусів. Потім поверніться в стартову позицію і повторіть вправу на іншу ногу.
9. Згинання рук на турніку. Взятись за турник з надхватом, руки розташувати на ширині плечей. Витягти ноги вперед та підняти їх, зігнути руки в ліктях, опустити тулуб і розтягнутися.
10. Однією з ефективних вправ для силових тренувань дома є штанга з рюкзаком. Для цього потрібен рюкзак з книжками або іншими важкими предметами, який можна зафіксувати на штангу. Піднімаючи штангу з рюкзаком, можна тренувати різні групи м'язів, залежно від вибраної ваги та кількості повторень. Важливо слідувати за правильним виконанням вправи та не перевантажувати своє тіло, особливо якщо немає досвіду в роботі зі штангою.
Ці вправи можуть бути корисні для тих, хто не має можливості відвідувати спортивні зали або фітнес-клуби, але хоче тренувати своє тіло вдома. При виконанні вправ важливо дотримуватися правильної техніки і не надмірно навантажувати себе, щоб уникнути травм та погіршення здоров'я. Перед початком будь-яких фізичних навантажень важливо проконсультуватися з лікарем або тренером.

Рисунок 3.11. результат при виборі категорії "Силові вправи" та формат "текст"
частина 2

Таким чином користувач має змогу розрахувати необхідну йому кількість добових калорій, білків, жирів та вуглеводів у залежності від його потреб та створити меню добового харчування. Якщо користувач хоче зайнятися тренуванням, він може це зробити вибравши для себе підходящу категорію вправ та зручний формат, або текстові вправи, або відео з YouTube.

4. ПРИКЛАДИ ВИКОРИСТАННЯ ТА ТЕСТУВАННЯ WEB-SERVISY

4.1 Опис роботи web-сервісу.

Цей розділ демонструє приклад роботи web-сервісу для створення плану харчування чи тренування людини. У цьому розділі буде продемонстровано як саме виглядає web-сервіс так як він працює.

Робота з web-сервісом починається із основної сторінки. Після ознайомлення із інтерфейсом, користувач вирішує що йому потрібно, якщо користувач просто хоче ознайомитися з сервісом, на головній сторінці є доволі багато цікавої інформації. Якщо користувач хоче створити план харчування або тренування він може перейти відразу до його створення через навігаційну кнопку "Мій кабінет" в header або внизу сторінки через кнопку "Створити план".

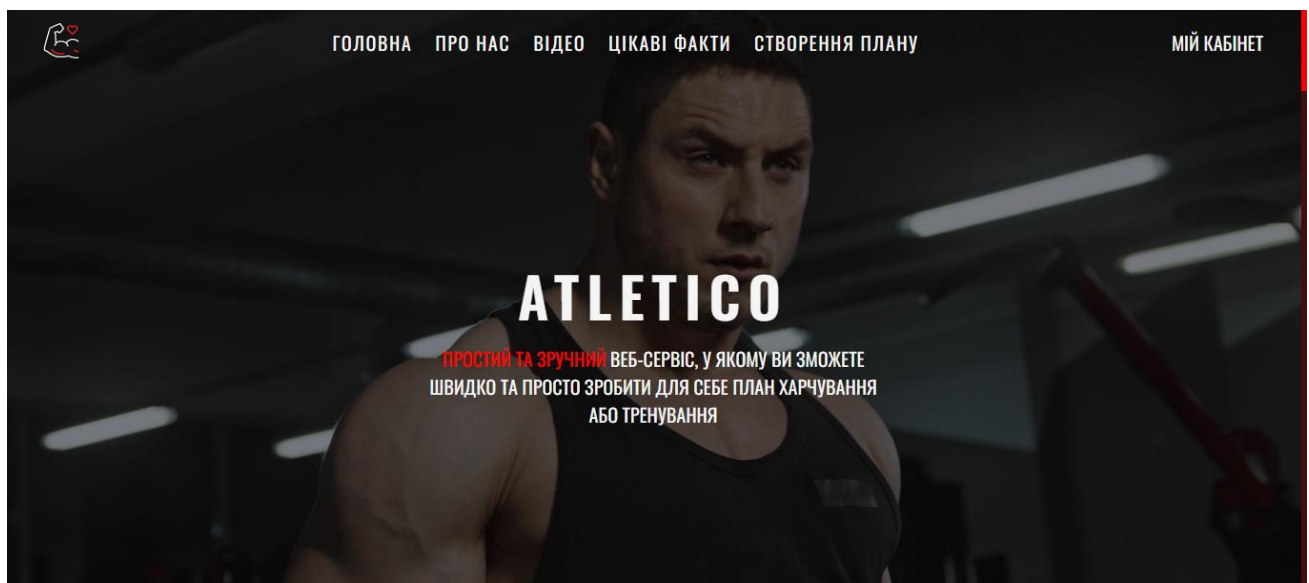


Рисунок 4.1. Головне меню.

При розробці інтерфейсу було спортивний стиль дизайну. Інтерфейс максимально простий та зручний у використанні, доступна навігація за допомогою панелі header, що робить орієнтування по web-сайту надпростим.

Однією з переваг такого рішення – покращення користувацького досвіду та полегшення навігації по сайту. Наповнення тематичною інформацією головну

сторінку web-сервісу новим користувачам дасть змогу зорієнтуватися, що таке фітнес в цілому та ознайомить з базовою інформацією та цікавими фактами про спорт та фітнес.



Рисунок 4.2. Розділ “Цікаві факти”

У розділі “Створити план” користувач натиснувши на кнопку буде перенаправлений в особистий кабінет де він зможе створити свій власний план тренування або харчування.

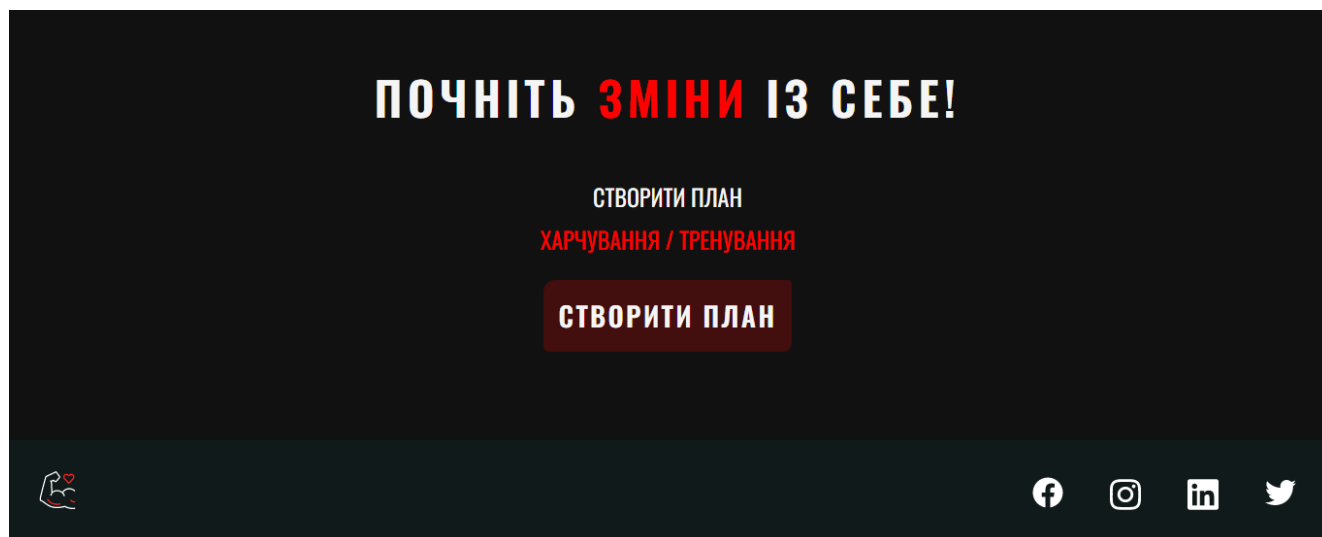


Рисунок 4.3. Розділ “Створити план”.

В особистому кабінеті користувач обирає що саме йому потрібно, або створити план харчування, або створити план тренування, в залежності від обраного варіанту йому будуть відкриті інші критерії підбору відповідно якщо буде обрано для харчування то буде показано критерії для розрахунку калорій, якщо для тренування то категорії вправ і формат.

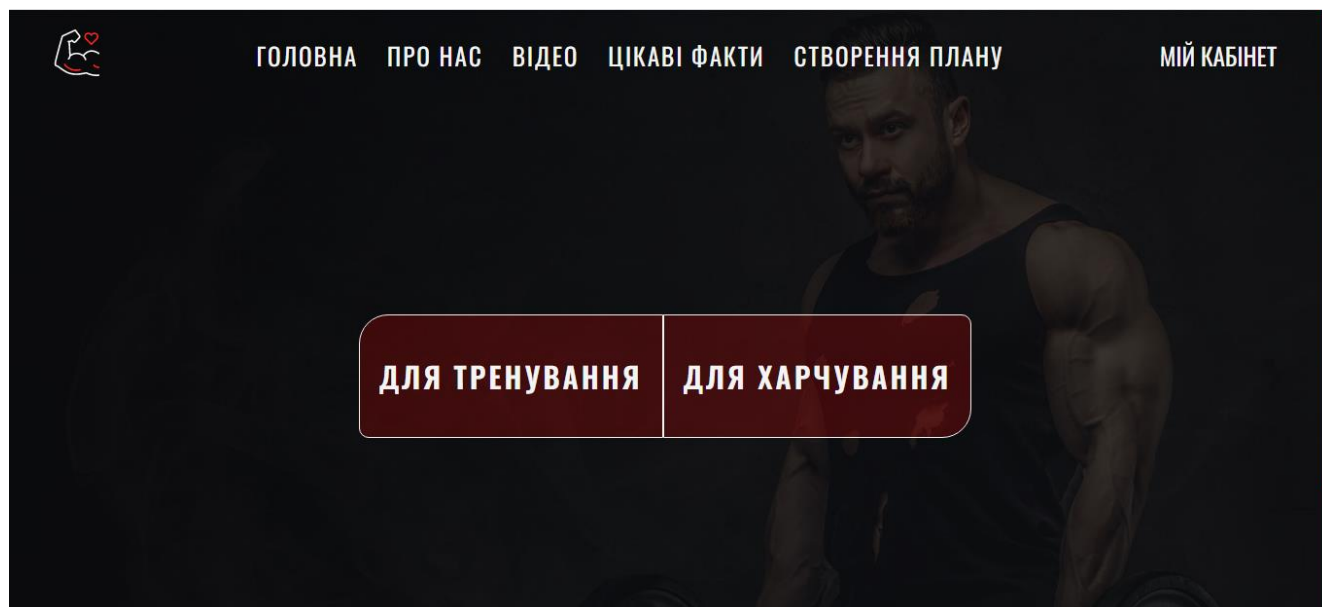


Рисунок 4.4. Особистий кабінет

4.2 Тестування роботи фітнес web-сервісу.

У процесі розробки фітнес web-сервісу, для забезпечення якості його роботи, потрібно перевірити як саме поводить себе web-сервіс при неправильних умовах, саме тому при створенні будь-якого програмного продукту проводиться його тестування на виявлення помилок, що можуть виникати при неправильному користуванні програмним продуктом.

Фітнес web-сервіс в першу чергу проходив перевірку як саме буде себе поводити при введенні неправильних даних. Для того, щоб web-сервіс працював без помилок, що виникали б при введенні некоректних даних, у форми для введення даних були додані додаткові умови, що перевіряють правильність введених даних. Таким чином, якщо при розрахунку добової кількості калорій, жирів, білків та вуглеводів, будуть введені некоректні данні, з'явиться повідомлення про помилку.

Так само і з планом тренування, щоб отримати очікуваний результат, користувач повинен обрати категорію вправ та формат результату, в іншому разі також з'явиться повідомлення про помилку.

Таблиця 4.1. Приклад виконуваних перевірок.

Номер кейсу	Заголовок	Пріоритет	Кроки виконання	Очікуваний результат
1	Введення від'ємних чисел у формі для розрахунку калорій	високий	1.Відкрити сторінку web-сервісу. 2.Перейти в особистий кабінет. 3.Вибрати 'План харчування'. 4.Ввести від'ємні данні. 5. Натиснути кнопку 'розрахувати'	З'являється повідомлення про те що введені данні не вірні
2	Введення реальних чисел у формі для розрахунку	високий	1.Відкрити сторінку web-сервісу. 2.Перейти в особистий кабінет. 3.Вибрати 'План харчування'. 4.Ввести правильні данні. 5. натиснути кнопку 'розрахувати'	Користувач побачить результат розрахунку
3	Спроба переглянути результати не вибравши нічого при створенні плану тренування	високий	1.Відкрити сторінку web-сервісу. 2.Перейти в особистий кабінет. 3.Вибрати 'План тренування'. 4. Натиснути кнопку 'показати'	З'являється повідомлення про те що не вибрана категорія вправ або формат результату
4	Спроба переглянути результати вибравши коректні варіанти при створенні плану тренування	високий	1.Відкрити сторінку web-сервісу. 2.Перейти в особистий кабінет. 3.Вибрати 'План тренування'. 4.Вибрати категорію і формат. 5.Натиснути кнопку 'показати'	Користувач побачить результат розрахунку

ВИСНОВКИ

Дипломна робота присвячена розробці фітнес web-сервісу з використанням технологій JavaScript, бібліотеки React та мови стилів CSS. У роботі було проведено аналіз існуючих фітнес web-сервісів та вирішено розробити новий web-сервіс, що дозволить користувачам почати займатися фітнес тренуванням для покращення свого фізичного стану, та створити власний план тренування або харчування.

Фітнес web-сервіс для створення плану харчування та тренування, що було розроблено в рамках дипломної роботи, є актуальним і має великий практичний потенціал для вивчення цієї теми.

У першому розділі "Огляд архітектури існуючих web-сервісів та аналіз проекту" було проведено аналіз та загальна характеристика наявних фітнес web-сервісів. Було розглянуто наявні аналоги, такі як MyFitnessPal, FitSW, JEFIT. Було проведено порівняння цих web-сервісів за різними критеріями, такими як: простота використання, наявність потрібних інструментів, переваги та недоліки. В результаті було виявлено, що на ринку існує певна кількість фітнес web-сервісів, але вони не задовольняють потреби тих користувачів, котрі прагнуть займатися фітнесом.

У другому розділі "Проектування web-сервісу для створення плану харчування та тренування людини" було проведено обґрунтування вибору середовища та інструментів розробки, таких як Visual Studio Code, React та технологію JSX та CSS. Виявлено, що ці засоби є найбільш оптимальними для створення фітнес web-сервісу. Також було розроблено загальний алгоритм реалізації проекту та описано функціонал продукту.

Третій розділ "Програмна реалізація фітнес web-сервісу" містить оцінку та планування розробки фітнес web-сервісу, опис набору інструментів, що використовувалися для розробки web-сервісу, та опис його функціоналу.

У четвертому розділі "Приклади використання web-сервісу" було надано опис роботи web-сервісу, наведено приклади його використання.

В результаті виконання дипломної роботи було розроблено фітнес web-сервіс, який містить функціонал, що дозволяє людям, які прагнуть займатися фітнесом, робити це в разі ефективніше, створюючи власні плани тренування або харчування, а також реалізовувати ці плани на практиці. Під час розробки було використано набір інструментів, які допомогли ефективно реалізувати потрібний функціонал. Для демонстрації роботи web-сервісу були надані приклади використання.

Отже, розробка такого фітнес web-сервісу для створення плану харчування та тренування є актуальною, оскільки, як показує практика, нове покоління геть не переймається за свій зовнішній вигляд, та і дорослі також. Не бажання людей займатися фітнесом викликане складністю користування наявних інструментів для створення плану тренування та харчування. Саме тому такий web-сервіс буде користуватись попитом, адже буде над простим у використанні.

У подальшому планується розробляти нові функції для сайту та покращувати дизайн, це допоможе втримувати старих користувачів та приманювати нових. Загалом такий web-сервіс допоможе людям досягати своїх цілей у напрямку свого фізичного здоров'я.

Щодо можливості подальшого розвитку web-сервісу, можна зазначити, що такий web-сервіс можна буде розвивати і надалі, шляхом оновлення вправ для тренування, розширення не тільки в сторону фітнесу, а і в сторону легкої атлетики, та інших видів спорту. Для цього можна використовувати нові технології, які з'являються на ринку, та використовувати їх для розвитку web-сервісу.

Підсумовуючи, слід зауважити, що розроблений фітнес web-сервіс є актуальним і має великий практичний потенціал для розвитку. Варто розвивати web-сервіс, додавати нові функції та інтегрувати його з іншими системами. Це дозволить збільшити практичну корисність та зробити його більш доступним для користувачів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Visual Studio Code [Електронний ресурс] // Microsoft. – 2023. – Режим доступу до ресурсу: <https://code.visualstudio.com/>
2. React Documentation [Електронний ресурс] – Режим доступу до ресурсу: <https://uk.legacy.reactjs.org/docs/getting-started.html>.
3. Офіційний web-сайт JavaScript. [Електронний ресурс]: [Web-сайт]. –: <https://www.javascript.com/>.
4. MyFitnessPal [Електронний ресурс] – Режим доступу до ресурсу: <https://www.myfitnesspal.com/>.
5. JEFIT [Електронний ресурс] – Режим доступу до ресурсу: <https://www.jefit.com/>.
6. FitSW [Електронний ресурс] – Режим доступу до ресурсу: <https://www.fitsw.com/trainer>.

ДОДАТОК А

ДЕМОНСТРАЦІЙНІ МАТЕРІАЛИ



ДЕРЖАВНИЙ УНІВЕРСИТЕТ ТЕЛЕКОМУНІКАЦІЙ
НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ
КАФЕДРА ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ



Розробка web-сервісу для створення індивідуального плану харчування та тренувань людини на основі бібліотеки React.JS

Виконав студент 4 курсу
групи ПД - 42
Малюк Вадим Володимирович
Керівник роботи
к.т.н. доц. Яскевич Владислав Олександрович

Київ – 2023

МЕТА, ОБ'ЄКТ ТА ПРЕДМЕТ ДОСЛІДЖЕННЯ

- **Мета роботи** – підвищити рівень контролю харчування та тренування людини шляхом розробки web-сервісу за допомогою бібліотеки react.js.
- **Об'єкт дослідження** - розробка web-сервісу.
- **Предмет дослідження** - web-сервіс для створення індивідуального плану харчування та тренування.

ЗАДАЧІ ДИПЛОМНОЇ РОБОТИ

1. Проаналізувати web-сервіси у цій галузі.
2. Розглянути та проаналізувати особливості наявних фітнес web-сервісів з інструментами для створення плану харчування або тренування.
3. Розробити фітнес web-сервіс за допомогою мови програмування JavaScript, її розширення JSX, в середовищі розробки Visual studio code та за допомогою бібліотеки React.
4. Проаналізувати переваги та недоліки програмних інструментів розробки.

3

АНАЛІЗ АНАЛОГІВ

Платформа	Переваги	Недоліки	Простота користування	Створення плану тренування	Створення плану харчування
MyFitnessPal	Не надто складний інтерфейс, є можливість зміни мови та зручні інструменти	Реклама всюди, платні деякі функції	Середньо	Так	Так
JEFIT	Є можливість додаткових безкоштовних функцій, які є платними в інших сервісах	Реклама всюди, платні деякі функції	Складно	Так	Ні
FitSW	Є можливість налаштувати під себе плани харчування та тренування	Складний у використанні, деякі функції платні	Складно	Так	Ні
Atletico	Простий у користуванні, є інструменти для створення плану харчування або тренування	Мала кількість вправ для тренування	Просто	Так	Так



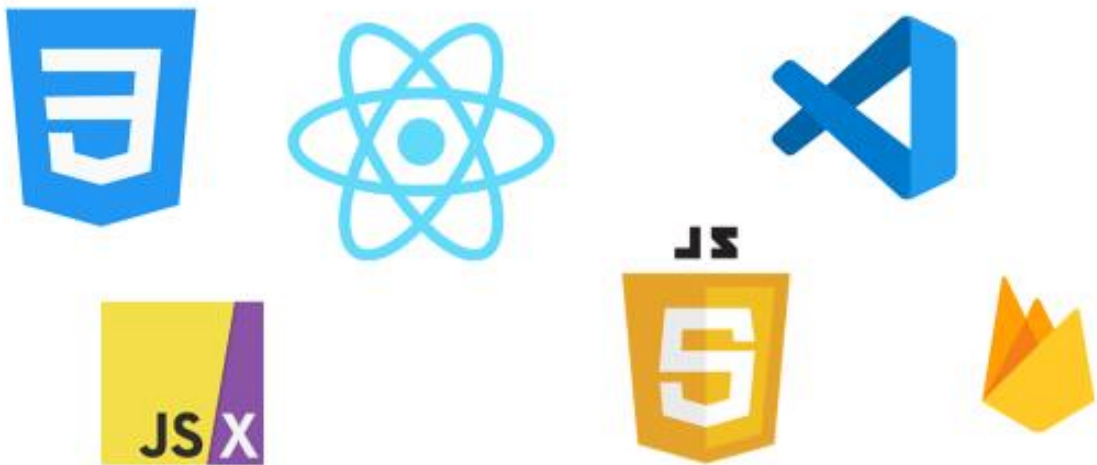
4

ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

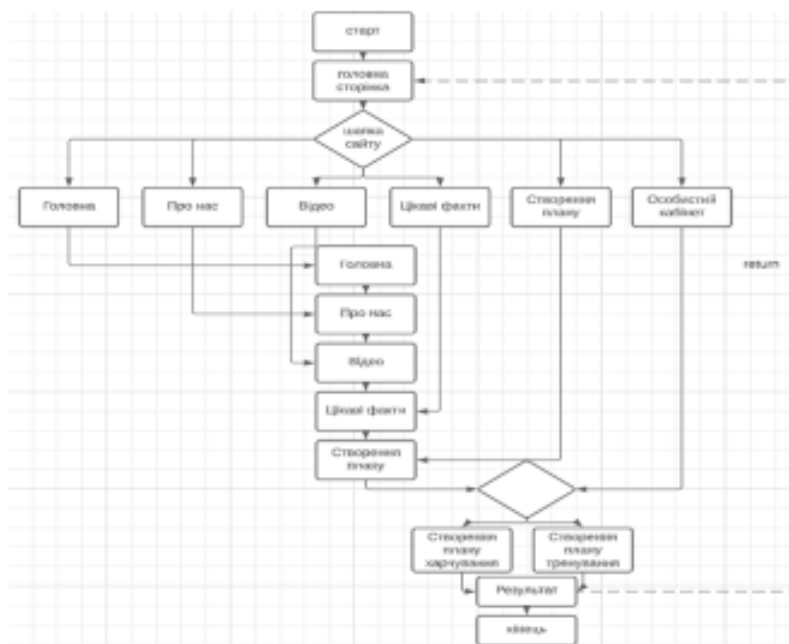
1. Розробити алгоритм, за допомогою якого буде розраховуватись необхідна кількість добових калорій, білків, жирів та вуглеводів.
2. Надати можливість авторизації користувача в особистому кабінеті за допомогою гугл-сервісу.
3. Надати можливість збереження результатів обчислення добової кількості калорій, білків, жирів та вуглеводів.
4. Розробити зручний у користуванні інтерфейс.

5

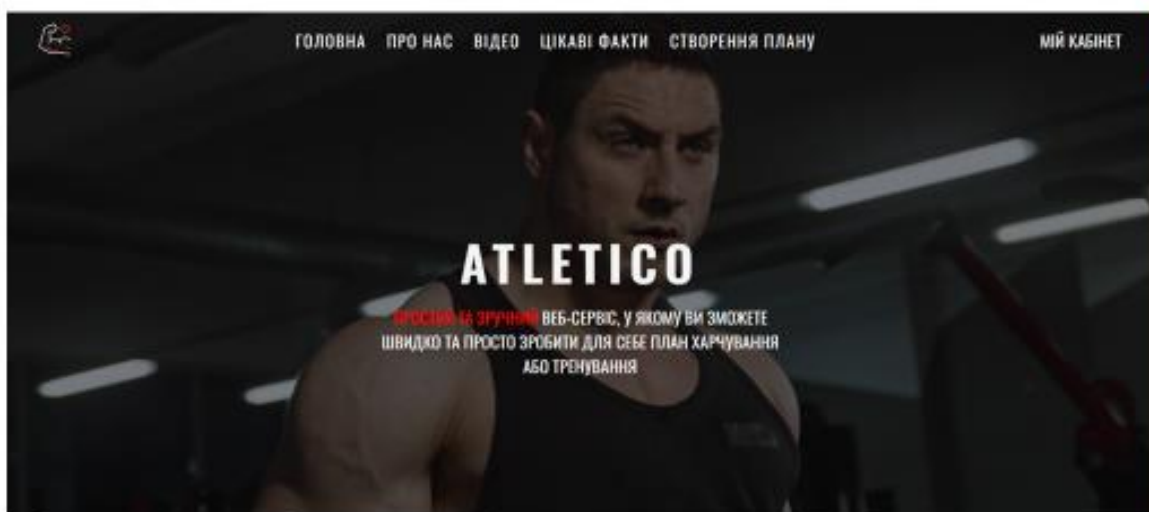
ПРОГРАМНІ ЗАСОБИ РЕАЛІЗАЦІЇ



ДІАГРАМА ДІЯЛЬНОСТІ



9

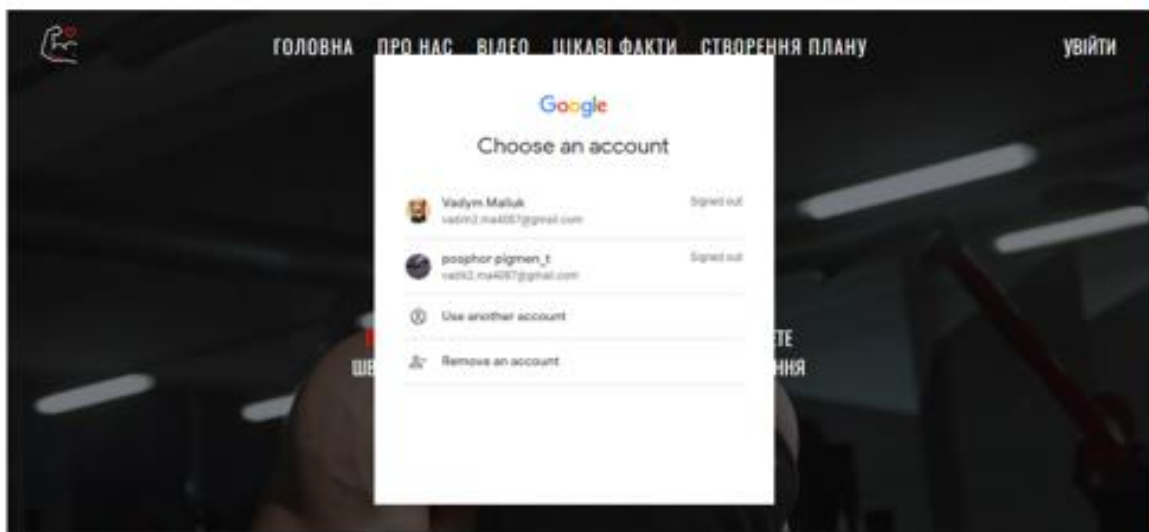


Головна сторінка web-сайту

10



Приклад контенту з головної сторінки



Авторизація в особистий кабінет

ВБЕРІТЬ СТАТЬ

ЧОЛОВІК ЖІНКА

ЗАПОВНІТЬ ДАНІ

ВІК РІСТ СМ ВАГА КГ

МАЛА АКТИВНІСТЬ СЕРЕДНЯ АКТИВНІСТЬ ВИСОКА АКТИВНІСТЬ

СКИНУТИ ВАГУ НАБРАТИ ВАГУ ОТРИМАТИ ВАГУ

РОЗРАХУВАТИ

ПОТРІБНА КІЛЬКІСТЬ КАЛОРІЙ: **3163**

ПОТРІБНА КІЛЬКІСТЬ БІЛКІВ: **152**

ПОТРІБНА КІЛЬКІСТЬ ЖИРІВ: **112**

ПОТРІБНА КІЛЬКІСТЬ ВУГЛЕВОДІВ: **336**

РЕКОМЕНДОВАНИЙ ІНСТРУМЕНТ ДЛЯ СТВОРЕННЯ МЕНЮ ХАРЧУВАННЯ: **СТВОРИТИ МЕНЮ**

Розрахунок добової кількості калорій, жирів, білків та вуглеводів для людини

13

ПОТРІБНА КІЛЬКІСТЬ КАЛОРІЙ:	2031
ПОТРІБНА КІЛЬКІСТЬ БІЛКІВ:	102
ПОТРІБНА КІЛЬКІСТЬ ЖИРІВ:	68
ПОТРІБНА КІЛЬКІСТЬ ВУГЛЕВОДІВ:	203
РЕКОМЕНДОВАНИЙ ІНСТРУМЕНТ ДЛЯ СТВОРЕННЯ МЕНЮ ХАРЧУВАННЯ:	СТВОРИТИ МЕНЮ

Історія			
1.		Калорій:	2000
		Білки:	157
		Жири:	70
		Вуглеводи:	210
2.		Калорій:	2031
		Білки:	152
		Жири:	68
		Вуглеводи:	203

Збереження історії розрахунку кількості калорій, білків, жирів та вуглеводів

14



Результат вибору типу вправ та формату користувачем

15

АПРОБАЦІЯ РЕЗУЛЬТАТІВ ДОСЛІДЖЕННЯ

Малюк В.В. Аналіз сучасних інструментів та методів створення веб-сайтів/Яскевич В.О, Малюк В.В// Всеукраїнська науково-технічна конференція «Застосування програмного забезпечення в ІКТ». Збірник тез. 20.04.2023, ДУТ, м. Київ — К.: ДУТ, 2023. — С. 19.

Малюк В.В. Розробка веб-сервісу для створення індивідуального плану харчування та тренування людини/Яскевич В.О, Малюк В.В// Всеукраїнська науково-технічна конференція «Застосування програмного забезпечення в ІКТ». Збірник тез. 20.04.2023, ДУТ, м. Київ — К.: ДУТ, 2023. — С. 21.

16

ВИСНОВКИ

1. Проведено аналіз предметної галузі, за результатами якої було виявлено, що більшість web-сервісів надають лише частковий безкоштовний доступ до своїх інструментів, в деяких випадках сервіси є повністю платними.
2. Проведено дослідження засобів створення програмного забезпечення.
3. Зроблено концепцію web-сервісу та вимоги, UML-діаграми використання та діяльності.
4. Розроблено web-сервіс для створення плану харчування та тренування.

17

ДЯКУЮ ЗА УВАГУ!