

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»

Факультет комп'ютерних наук і технологій

(повне найменування інституту, факультету)

Кафедра програмних засобів

(повне найменування кафедри)

Пояснювальна записка

до дипломного проєкту (роботи)

бакалавр

(ступінь вищої освіти)

на тему РОЗРОБКА ВЕБСИСТЕМИ ТРЕНАЖЕРНОГО ЗАЛУ
DEVELOPMENT OF A GYM MANAGEMENT WEB SYSTEM

Виконав: студент(ка) 4 курсу, групи КНТ-138
Спеціальності 121 Інженерія програмного
забезпечення

(код і найменування спеціальності)

Освітня програма (спеціалізація)

Інженерія програмного забезпечення

Самойленко Є.В.

(прізвище та ініціали)

Керівник Скрупський С.Ю.

(прізвище та ініціали)

Рецензент Голуб Т.В.

(прізвище та ініціали)

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Запорізька політехніка»
 (повне найменування закладу вищої освіти)

Інститут, факультет ФКНТ
 Кафедра програмних засобів
 Ступінь вищої освіти бакалавр
 Спеціальність 121 Інженерія програмного забезпечення
(код і найменування)
 Освітня програма (спеціалізація) Інженерія програмного забезпечення
(назва освітньої програми (спеціалізації))

ЗАТВЕРДЖУЮ

Завідувач кафедри ПЗ, д.т.н, проф.
С.О. Субботін
 “ ” 2022 року

З А В Д А Н Н Я

НА ДИПЛОМНИЙ ПРОЄКТ (РОБОТУ) СТУДЕНТА(КИ)

Самойленко Євгенія Віталійовича
(прізвище, ім'я, по батькові)

1. Тема проєкту (роботи) Розробка вебсистеми тренажерного залу. Development of a Gym Management Web System

керівник проєкту (роботи) Скрупський Степан Юрійович, к.т.н., доцент,
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом закладу вищої освіти від “ 27 ” квітня 2022 року № 102

2. Строк подання студентом проєкту (роботи) 07 червня 2022 року

3. Вихідні дані до проєкту (роботи) рекомендована література, технічне завдання

4. Зміст розрахунково–пояснювальної записки (перелік питань, які потрібно розробити) 1. Аналіз проблеми та постановка технічного завдання. 2. Методи розробки вебсистем. 3. Проєктування вебсистеми. 4. Розробка вебсайту тренажерного залу

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
Слайди презентації

6. Консультанти розділів проєкту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	прийняв виконане завдання
1–4 Основна частина	Скрупський С.Ю., доцент		
Нормоконтроль	Камінська Ж.К., асистент		

7. Дата видачі завдання « 04 » квітня 2022 року.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів дипломного проєкту (роботи)	Строк виконання етапів проєкту (роботи)	Примітка
1	Постановка завдання роботи.	1 тиждень	Завдання, ТЗ
2	Аналіз предметної області.	1 тиждень	Розділ 1
3	Вибір мови програмування та інших технологій	2 тиждень	Розділ 2
	розробки.		
4	Розробка структури вебсистеми.	2 тиждень	Розділ 3
5	Розробка бази даних.	3–4 тижні	Розділ 3, 4
6	Розробка вебсайту.	5 тиждень	Розділ 4
	програмного забезпечення.		
7	Оформлення пояснювальної записки та документів	6 тиждень	Додатки
	до неї.		
8	Нормоконтроль та рецензування.	7 тиждень	
9	Захист роботи.	8 тиждень	

Студент(ка)

(підпис)

Самойленко Є.В.

(прізвище та ініціали)

Керівник проєкту (роботи)

(підпис)

Скрупський С.Ю.

(прізвище та ініціали)

РЕФЕРАТ

Пояснювальна записка до дипломної кваліфікаційної роботи бакалавра: 87 с., 60 рис., 6 табл., 4 дод., 16 джерел.

АБОНЕМЕНТ, ВЕБСАЙТ, ВЕБСИСТЕМА, БАЗА ДАНИХ, СТОРІНКА, ТРЕНАЖЕРНИЙ ЗАЛ, ФОРМА, BOOTSTRAP, CSS, HTML, JAVA SCRIPT, JQUERY, PHP, MYSQL.

Об'єкт дослідження – програмні засоби для інформаційної системи тренажерного залу.

Предмет дослідження – вебсайт тренажерного залу.

Мета роботи – створення програмної системи для представлення інформації в мережі інтернет та автоматизації роботи тренажерного залу.

Матеріали, методи та технічні засоби: структурне та об'єктно. Для розробки системи було обрано найбільш підходящі програмні засоби. Як СУБД обрано MySQL. Мови програмування: для верстки HTML, JavaScript, CSS, JavaScript, бібліотека jQuery, для серверної розробки PHP. Як серверну мову програмування було обрано PHP.

Результати. Створено вебсистему тренажерного залу.

Висновки. Розроблений вебсайт та базу даних можна використовувати для автоматизації роботи клубу та зберігання даних.

Галузь використання – тренажерні зали та фітнес клуби.

ABSTRACT

Explanatory note to the diploma qualifying work of the bachelor: 87 pages, 60 figures, 6 tables, 4 appendixes, 16 sources.

SUBSCRIPTION, WEBSITE, WEBSITE, DATABASE, PAGE, GYM, FORM, BOOTSTRAP, CSS, HTML, JAVA SCRIPT, JQUERY, PHP, MYSQL.

Object of study is software for the gym information system.

Subject of study are the gym website.

The purpose of the work is to create a software system for the automation and acceleration of the rehabilitation of the child under the supervision of a physician or for home use under the parents supervision.

Materials, methods and technical means: structural and object. The most suitable software tools were selected for the development of the system. MySQL is selected as the DBMS. Programming languages: for HTML layout, JavaScript, CSS, JavaScript, jQuery library, for PHP server development. PHP was chosen as the server programming language.

Results. A gym website has been created.

Conclusions. The developed website and database can be used to automate the work of the club and data storage.

The field of use is a gyms and fitness clubs.

ЗМІСТ

С.

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК	7
ВСТУП	8
1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ.....	10
1.1 Сучасні вебсистеми.....	10
1.2 Вимоги до вебсистеми тренажерного залу.....	11
1.3 Висновки за розділом	13
2 МЕТОДИ РОЗРОБКИ ВЕБСИСТЕМ.....	14
2.1 Технології розробки вебсистем	14
2.2 Середовища розробки вебсистем	18
2.3 Висновки за розділом	22
3 ПРОЄКТУВАННЯ ВЕБСИСТЕМИ ТРЕНАЖЕРНОГО ЗАЛУ	24
3.1 Розробка структури вебсистеми	24
3.2 Розробка бази даних.....	27
3.3 Розробка дизайну вебсайту	29
3.4 Висновки за розділом	33
4 РОЗРОБКА ВЕБСАЙТУ ТРЕНАЖЕРНОГО ЗАЛУ	35
4.1 Керівництво програміста.....	35
4.2 Керівництво користувача	45
4.3 Висновки за розділом	55
ВИСНОВКИ.....	56
ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ	58
Додаток А Текст файлу main.js.....	60
Додаток Б Текст файлу main.php	64
Додаток В Текст файлу form.php.....	74
Додаток Г Слайди презентації	80

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАК

API – Application Programming Interface;
CSS – Cascading Style Sheets,
HTML – HyperText Markup Language,
IDE – Integrated Development Environment,
JS – JavaScript,
PHP – Hypertext Preprocessor,
БД – Бази Даних,
СУБД – Системи Управління Базами Даних.

ВСТУП

Розробка вебсистеми — одна з найбільш популярних послуг в галузі інтернет–технологій. Створення вебсайту – процедура складна і ретельна. Актуальність розробки вебсистем полягає в наступному:

- допомога та розвиток стилю фірми в мережі Інтернет (іміджева складова частина);
- надання більш детальних даних про пропозиції компанії (інформаційна складова частина);
- універсальність та актуальність інформації. Тобто в реальному часі показувати всі минулі зміни фірми, анонси компанії, зміни вартості, послуг, обставин тощо. У будь–яких обставинах споживач вільно здатний подивитись нові відомості на вебсайті;
- рекламне оголошення компанії у мережі Інтернет (рекламна складова частина). Усе найбільше фірм вкладають кошти в інтернетрекламу. Вона дає можливість охопити велику аудиторію, дає можливість повідомити великій кількості покупців відомості про товари та послуги фірми;
- залучення нових покупців та партнерів, реалізація товарів та послуг з сайту компанії (комерційна складова частина).

Актуальність розробки вебсистеми тренажерного залу полягає в інформаційному забезпеченні фірми в реальному часі. Вебсистема дозволяє надати інформацію про компанію та її товари або послуги стисло і водночас повноцінно. Також сайт повідомляє про новини фірми, про зміни в прайсі або режим роботи, містити мотиваційну інформацію для майбутніх клієнтів.

Задачі, які потрібно вирішити при розробці вебсистеми тренажерного залу:

- аналіз сучасних вебсистем та вимоги до системи тренажерного залу;
- аналіз методів розробки вебсистем, порівняння середовищ розробки;
- розробка структури вебсистеми;
- розробка бази даних тренажерного залу;

- розробка дизайну вебсайту тренажерного залу;
- верстка вебсайту тренажерного залу.

Практичну значущість роботи становить підвищення якості роботи з клієнтами, наданої їм інформації, підвищення якості процесів вирішення проблемних ситуацій у роботі з клієнтами. Атоматизація надання послуг тренажерного залу в онлайн режимі.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Сучасні вебсистеми

Подивитися сайт може будь-яка людина, яка має комп'ютер, смартфон, планшет і т.д., підключений до Інтернету.

За призначенням, цілями та функціями сайти можна класифікувати наступним чином [1]:

- пошукові системи – добре всім відомі сайти Google і т.д., які служать для пошуку інформації в мережі Інтернет, тим самим спрощують життя рядового користувача;

- сайт-візитка – невеликий за обсягом сайт, основним завданням якого є презентація компанії, її послуг та продукції. Як правило складається з кількох статичних сторінок, об'єднаних корпоративним стилем, інформація на яких оновлюється досить рідко;

- промо-сайт – створюються з метою разового просування якогось події, заходи тощо. Можуть бути як окремими самостійними сайтами, і частиною якогось основного сайту. Як правило, наявність у якогось заходу окремого промо-сайту вважається гарним тоном та високим рівнем організації просування;

- корпоративна вебсистема – один із найпоширеніших видів сайтів у світовому павутинні. Як правило, такі сайти мають складну розгалужену структуру і багатий функціонал, що дозволяє вирішувати безліч задач. Дизайн такого сайту найчастіше витримані у якомусь корпоративному стилі. Корпоративні сайти часто можуть обслуговуватися людиною, яка не знайома з програмуванням, оскільки вони вибудовані на основі двигуна будь-якої контент-менеджмент системи CMS [1];

- інтернет-магазини – створюються з метою реалізації товарів або послуг через Інтернет. Можуть бути як частиною будь-якого корпоративного сайту і по суті є його відгалуженням, так і окремим сайтом. Інтернет-магазин обов'язково містить у собі каталог товарів, кошик для покупок, різні варіанти

доставки та оплати товарів. Інтернет–магазин схожий на звичайний магазин, але має ряд переваг: велике охоплення аудиторії; зниження витрат на утримання магазину; зручність у пошуку товарів; простота у зміні асортименту у каталозі товарів; обробка замовлень займає менше часу;

– тематичні портали та новинні інтернет–видання – сайти, що містять на своїх сторінках безліч різноманітної інформації, присвяченої будь–якій темі або об'єднаній під шапкою якогось видання. Як правило, мають складну розгалужену структуру;

– форум — можливо, як розділом якогось основного сайту, так і окремо створеним сайтом, призначений для спілкування відвідувачів сайту між собою. Форум зручний для моніторингу тем відвідувачів сайту, що цікавлять, необхідних у проєкті.

Якщо говорити про останні тенденції в сайтобудуванні, можна виділити такі тенденції, що характеризують сучасну епоху веброзробки. Серед них: використання віртуальної та доповненої реальності, розподіленого реєстру, оплати за допомогою внутрішніх монет та інші цікаві ініціативи [2].

1.2 Вимоги до вебсистеми тренажерного залу

Створити якісний спортивний сайт або фітнес–лендінг важливо для залучення нових клієнтів, зручності нинішніх користувачів. Сайти потрібні для підтримки довіри, спрощення взаємодії між користувачем та клубом (наприклад, через особистий кабінет), надання інформації про послуги, а також акредитацію спортивних тренерів.

Щоб створити конверсійний сайт тренажерного клубу, важливо врахувати його функціональність, корисність. Потрібна зручна структура, набір посадкових сторінок, інтеграція додаткових сервісів. Наприклад, особистий кабінет дасть можливість клієнту стежити за своїм розкладом, акціями клубу, консультуватись із інструкторами онлайн. Для спортивних

лендингів характерною є система купонів, а також повідомлень через соціальні мережі, щоб підтримувати лояльність споживачів.

Щоб вебсайт мав високу конверсію та приносив прибуток, потрібно проаналізувати цільову аудиторію тренажерного клубу, прикладом якої можуть бути: професійні спортсмени, молоді мами, студенти. Це різні демографічні групи, кожній із яких потрібен індивідуальний підхід [2].

Наприклад, якщо ваша цільова група є студентами, підкресліть доступність цін, близькість до університетського корпусу. Якщо молоді мами — покажіть, що всередині фітнес-центру є дитячий куточок або спортивні секції, де вона зможе залишити дитину. Професійним спортсменам важливо знати, що у вашому спортзалі працюють професійні тренери зі стажем, є сучасні тренажери.

Якщо сайт не зможе підкреслити вигоди клубу для кожного сегмента аудиторії, він залишиться просто красивою картинкою, а не інструментом продажів. Щоб цього уникнути, необхідно правильно оформити блок переваг – описати, які проблеми клієнта допоможе вирішити центр, що саме він отримає, займаючись у вас.

Щоб вибрати потрібний спортзал, користувачі порівнюють їхні соціальні мережі, сайти. Найкращі сайти тренажерних клубів надають корисну інформацію про компанію, пропонують онлайнкову допомогу.

Обов'язково треба вказати послуги фітнес клубу та відомості про тренерів та їх досягнення. Ця інформація допоможе показати експертність, завоювати довіру.

Щоб зацікавити користувача недостатньо просто написати, які в клубі тренажери. Потрібно розмістити зображення на сторінці, розповісти про застосування різних типів обладнання.

Розклад тренувань з можливістю онлайн-запису або придбанням абонементу – ця функція дозволить клієнту відразу вибрати зручний час, без необхідності залишати заявку або телефонувати операторам.

Розробка блогу із рекомендаціями інструкторів. Інформативні, корисні статті, а також відео для кожного сегмента цільової аудиторії допоможуть виділитися серед інших залів, допоможуть SEO–просування сайту.

1.3 Висновки за розділом

Створити якісний спортивний сайт або фітнес–лендінг важливо для залучення нових клієнтів, зручності нинішніх користувачів. Сайти потрібні для підтримки довіри, спрощення взаємодії між користувачем та клубом (наприклад, через особистий кабінет), надання інформації про послуги, а також акредитацію спортивних тренерів. Щоб створити конверсійний сайт тренажерного клубу, важливо врахувати його функціональність, корисність. Потрібна зручна структура, набір посадкових сторінок, інтеграція додаткових сервісів.

2 МЕТОДИ РОЗРОБКИ ВЕБСИСТЕМ

2.1 Технології розробки вебсистем

Вебсистема – це сайт з елементами інтерактиву. Вони дозволяють користувачам взаємодіяти: натискати кнопки, заповнювати форми, вимагати прайс, робити покупки. Поштові клієнти, соцмережі, пошукові системи, інтернет–магазини, програми для управління проектами — це приклади таких додатків [3].

З погляду архітектури вебапрограми складаються з двох частин: клієнтської та серверної. Клієнтська частина також називається фронтендом. По суті, це те, що користувачі бачать на екрані пристрою. Серверна частина або бекенд — програмно–апаратна частина сервісу. Це набір коштів, що реалізують логіку програми.

Виділяють три види вебсистем в сучасній літературі, які визначають підхід до розробки [3]-[4]:

- SPA або Single Page Application — це односторінкова вебпрограма, яка завантажується на одну сторінку HTML. Користувачам не потрібно перезавантажувати або підвантажувати додаткові сторінки завдяки динамічному оновленню за допомогою JavaScript. Тобто, користувачі бачать у браузері весь основний контент, а коли прокручують сторінку або переходять на інші, потрібні елементи просто підвантажуються;

- MPA або Multi Page Application — це багатосторінкові програми, де при кожній зміні даних або завантаженні нової інформації оновлюється сторінка;

- PWA або Progressive Web App — це гібрид сайту з мобільним додатком. Вони можуть встановлюватися на головний екран смартфона, надсилати push–повідомлення та працювати в офлайн режимі.

Клієнтська частина програми (фронтенд), тобто клієнт є інтерактивною частиною програми, яка виконується у веббраузері на

комп'ютері, смартфоні або планшеті. Клієнтська частина реалізує інтерфейс вебсистеми і завантажується у вигляді динамічних вебсторінок [4].

HTML — це стандартна мова розмітки, яку використовують для створення вебпроектів. Його елементи дозволяють відображати стандартні блоки сторінок, а також являють собою форматований текст, зображення, таблиці, форми введення даних і т.д. Коли браузер надсилає запит URL-адреси, щоб отримати сторінку або програму, спочатку повертається HTML-документ. HTML-документ містить додаткові відомості про оформлення або макет у вигляді CSS, а також про поведінку у вигляді JavaScript [5].

Каскадні таблиці стилів (CSS) — це мова розмітки, яка визначає оформлення та макет елементів HTML. Таким чином, HTML визначає структуру, а CSS — стиль. За допомогою CSS задаються шрифти, кольори, стилі, розташування окремих елементів та відображення сторінок на різних пристроях [6].

JavaScript — це мова програмування, яка допомагає реалізувати складну поведінку вебсторінки. За допомогою JavaScript можна створювати меню, анімацію, відеоплеєри, інтерактивні карти та навіть прості ігри у браузері [7].

Щоб ефективніше розробляти JavaScript, розробники використовують фреймворки, які також є важливою складовою технологічного стека. Фреймворк — це робоче середовище, свого роду каркас, який допомагає ефективніше створювати та підтримувати програмні продукти. Далі ми перерахуємо деякі з фреймворків та бібліотек, які ми зазвичай використовуємо у проєктах.

Популярні фреймворки та бібліотеки JavaScript.

React — бібліотека з відкритим кодом для створення інтерфейсів користувача. За допомогою React розробники створюють об'єкти веб, які працюють без перезавантаження основної сторінки. Одна з ключових особливостей React — універсальність. Цю бібліотеку можна використовувати на сервері та на мобільних платформах за допомогою React Native [8].

Ще одна важлива особливість бібліотеки – декларативність, це скорочує код та робить його зрозумілим.

Angular – це фреймворк від компанії Google. Насамперед він націлений на розробку SPA-рішень. Це середовище розробки відоме насамперед тому, що воно надає розробникам найкращі умови для об'єднання JavaScript з HTML та CSS. До вебдодатків на Angular.js відносяться Google та Youtube [9].

Vue – це розробка для реалізації інтерфейсів користувача. Він легко інтегрується вбагатофайловий та багатотехнологічний проєкт.

Під серверною частиною розуміють набір апаратно-програмних засобів, за допомогою яких реалізовано логіку роботи програми. Це те, що відбувається поза браузером та комп'ютером користувача. До бекенд відноситься панель адміністрування, управління даними, логіка їх передачі за запитами фронтенда [10].

Завдання серверної розробки — зробити так, щоб відповідь від сервера сягала клієнта і спроектовані блоки функціонували належним чином. А також створити для замовника зручне та безпечне середовище для наповнення та оновлення контенту на сайті.

Ось деякі мови програмування та їх основних фреймворків, які ми використовуємо на вебпроєктах.

PHP – одна з найпоширеніших мов веброзробки. PHP переважно застосовується, щоб «оживити» статичні HTML сторінки. Майже завжди користувачі приходять на сайт за інформацією, яка постійно змінюється, і потрібно відображати її актуальний стан. Наприклад, показати прогноз погоди. Якщо використовувати лише HTML, то вирішити такі завдання не вдасться. Тут може допомогти PHP: він приймає запит від вебсервера, виконує сценарій і повертає результат у вигляді готового HTML-коду. Сервер надсилає цей результат у браузер користувачеві, який, своєю чергою, відображає її користувачеві. Після цього бачить актуальний прогноз погоди [10].

Java – це універсальна мова програмування, яка часто використовується для веброзробки, а також для розробки під Android. Серед переваг вебдодатків на Java – кросплатформність, мультифункціональність, надійність та гнучкість [9].

Для розробки вебпрограм на Java ми використовуємо фреймворк Spring. Він забезпечує комплексну модель розробки та конфігурації для сучасних бізнес-додатків: від ecommerce-проектів до великих порталів, від освітніх платформ до урядових ресурсів. Ключовий елемент Spring – підтримка інфраструктури на рівні програми, тому розробники можуть зосередитись на бізнес-логіці без зайвих налаштувань залежно від середовища виконання.

Також із фреймворків Java застосовуємо Hibernate. Він спрощує розробку Java програми для взаємодії з базою даних. Node.js — кросплатформове середовище, що дозволяє розробникам використовувати JavaScript на стороні сервера перш ніж вона буде доступна в веббраузері користувача.

Також вебсистеми використовують бази даних для зберігання інформації. Існує два типи баз даних: реляційні та нереляційні. Відмінності з-поміж них у тому, як вони спроектовані, які типи даних підтримують, як зберігають інформацію. Найбільш поширені бази даних розробки вебпроектів: MySQL, PostgreSQL, MongoDB [11].

В таблиці 2.1 нижче ми зібрали основні технології розробки вебсистем, які були описані.

Таблиця 2.1 – Технології розробки вебсистем

Фронтенд	Бекенд
HTML – її елементи дозволяють відображати стандартні блоки сторінок, форматований текст, зображення, таблиці і т.д.	PHP – переважно застосовується, щоб зробити інтерактивними статичні HTML сторінки

Продовження таблиці 2.1

Фронтенд	Бекенд
CSS – за допомогою CSS задаються шрифти, кольори, стилі, розташування окремих елементів та відображення сторінок на різних пристроях.	JavaEE – кросплатформність, мультифункціональність, надійність та гнучкість. Фреймворки: - Spring; - Hibernate.
JavaScript – використовується для розробки інтерактивних об'єктів вебсторінок, які покращують взаємодію користувача. Популярні фреймворки: - React; - Angular; - Vue.	Node.js – кросплатформове середовище, яке виконує код JavaScript поза браузером. MySQL – реляційна БД зберігає структуровані дані, які зазвичай становлять об'єкти реального світу.

2.2 Середовища розробки вебсистем

Інтегроване середовище розробки (IDE) – це програмне забезпечення, яке можна завантажити на свій комп'ютер. Як говорилося вище, призначене для спрощення процесу веброзробки [12].

Ось основні функції середовища:

- текстовий редактор – це текстовий редактор у тому числі, але з деякими специфічними функціями;
- компілятор або інтерпретатор – це інструмент, що транслює вашу програму на машинний код. Завдяки цьому відбувається виконання скрипту. Інтерпретатор – вбудований інструмент, мета якого – виконання коду без попередньої компіляції;

- CI/CD – це вбудовані способи автоматизації необхідних процесів;
- залагоджувач — вбудований інструмент, який перевіряє код на наявність помилок. За допомогою нього можна виправити проблемні місця у коді;
- підсвічування синтаксису — інструмент, який підсвічує код. Кольори підсвічування відповідають певним елементам: властивостям, тегам, атрибутам тощо;
- графічний інтерфейс користувача (GUI) – це зручний сприйняття інтерфейс, у якому людині простіше орієнтуватися і виконувати дрібні взаємодії з кодом. У ньому можуть бути кнопки, меню та інші елементи. Це набагато простіше, ніж вручну вводити команди.

При виборі середовища розробки важливо спочатку визначитися із завданнями, які вона виконуватиме. Якщо забути про основні компоненти середовища, то ось список можливих функцій, на які можна звернути увагу [13]:

- підтримка кількох мов програмування – тобто, повинна вміти працювати з усіма мовами, які потрібні для вашого проєкту;
- можливість кастомізації – деякі IDE дають користувачеві можливість підлаштовувати графічний інтерфейс під себе;
- юніт-тестування – це функція, що дозволяє вбудовувати мок-об'єкти у секції коду. Це звільняє вас від необхідності дописувати цю секцію, адже цей інструмент дозволяє її відразу протестувати;
- бібліотека вихідного коду — мають ресурси, що містять готові скрипти і код, які ви можете використовувати у своєму проєкті;
- засоби діагностики помилок та ведення їх звітності;
- автодоповнення коду;
- інтеграції та плагіни – можливість інтегрувати проєкт з GitHub або SVN.
- пошук за кодом – за допомогою нього ви зможете швидко знайти у коді необхідний елемент;

– діаграми ієрархії – можливість налаштування та відображення ваших файлів та скриптів у великих проєктах у зручному для вас порядку. Відображення ієрархічної діаграми може показати порядок виконання скриптів, їх зв'язок з іншими файлами і скриптами.

Visual Studio – це одна з найпопулярніших і найдоступніших середовищ для веброзробки. Visual Studio підтримує мови ASP.NET, Python, Node.js, C++. Можна створювати середовища розробки прямо в хмарі [14].

IntelliJ IDEA – одна з найкращих середовищ для Java. Основна увага тут приділяється функціям, що підвищують продуктивність. Всі вони компактні і не заважають графічний інтерфейс. IntelliJ IDEA індексує код, може дати кілька порад або навіть дописати код за вас. Ця платформа може автоматизувати деякі завдання, постійне повторення яких могло викликати роздратування [15].

Aptana Studio 3 – одна з найкращих операційних платформ для веброзробки. В ній можна настроїти графічний інтерфейс під себе: вбудований термінал, інтеграція з Git, відладчик, а також майстер розгортання. Автодоповнення доступне для HTML, CSS та JS [11].

PyCharm – це одна з найкращих середовищ для розробки на Python. PyCharm перевіряє код на наявність помилок прямо під час написання програми та має функцію автодоповнення [12].

PhpStorm – розроблена під PHP, але повністю підтримує інші фронтед-мови: HTML 5, CSS, JavaScript та інші. У ній є підтримка базового програмного забезпечення, необхідного для створення сайтів WordPress, Drupal і т.д. В інтерфейсі цієї платформи легко орієнтуватися, в ній є автодоповнення коду, інструменти для налагодження та тестування. Ще вона дозволяє проводити швидкий (і безпечний) рефакторинг [11].

WebStorm входить до списку кращих IDE для веброзробки, завдяки підтримки JavaScript. Вона включає функцію автодоповнення, виявляє помилки прямо під час написання коду і має вбудований налагодчик. Також підтримує модульне тестування та інтеграцію з такими сервісами: Git,

GitHub, Mercurial тощо. WebStorm має високу швидкодію і може автоматизувати багато процесів, що сприятливо позначиться на вашій продуктивності [16].

NetBeans – це одна з найпопулярніших платформ для розробки, тому що вона підтримує багато різних мов: Java, JavaScript, PHP, HTML 5, CSS та інші. NetBeans допомагає оптимізувати процес розробки за допомогою синтаксичного та семантичного підсвічування коду. Вона має безліч потужних інструментів для рефакторингу і при цьому є опенсорс проєктом.

Komodo IDE – одна з найкращих платформ преміум-класу, використовується під веброзробку. Вона має функцій: автозаповнення, візуальний відладчик, підтримку модульного тестування, доповнення, спільну розробку з командою, інструментарій для управління проєктами та багато іншого. Komodo IDE підтримує роботу з Git, SVN та може працювати з усіма мовами програмування [14].

В таблиці 2.2 наведено переваги кожного розглянутого інтегрованого середовища.

Таблиця 2.2 – Аналіз IDE

IDE	Переваги
Visual Studio	Підтримує мови ASP.NET, Python, Node.js, C++. Можна створювати середовища розробки прямо в хмарі.
IntelliJ IDEA	Основна увага тут приділяється функціям, що підвищують продуктивність (індексація коду, автодоповнення, рефакторинг).
Aptana Studio 3	В ній можна настроїти графічний інтерфейс під себе: вбудований термінал, інтеграція з Git, відладчик, а також майстер розгортання.
PyCharm	Середовище для розробки на Python. PyCharm перевіряє код на наявність помилок прямо під час написання програми та має функцію автодоповнення.

Продовження таблиці 2.2

IDE	Переваги
PhpStorm	Розроблена під PHP, але повністю підтримує інші фронтенд-мови: HTML 5, CSS, JavaScript та інші.
WebStorm	Вона включає функцію автодоповнення, виявляє помилки прямо під час написання коду і має вбудований налагодчик. Також підтримує модульне тестування та інтеграцію з такими сервісами: Git, SVN
NetBeans	Допомагає оптимізувати процес розробки за допомогою синтаксичного та семантичного підсвічування коду. Вона має безліч потужних інструментів для рефакторингу і при цьому є опенсорс проєктом
Komodo IDE	Вона має функцій: автозаповнення, візуальний відладчик, підтримку модульного тестування, доповнення, спільну розробку з командою, інструментарій для управління проєктами та багато іншого. Komodo IDE підтримує роботу з Git, SVN та може працювати з усіма мовами програмування.

2.3 Висновки за розділом

У веброзробці технологічний стек складається з дуже специфічного набору компонентів, які працюють разом для створення сайту. Він будується з двох основних частинах: зовнішньому (на стороні клієнта) та внутрішньому інтерфейсі (на стороні сервера). Кожна з двох частин виконує певні функції і включає відповідний набір інструментів. До клієнтської частини відносяться технології CSS, React, Angular, Vue.js, JavaScript. Бекенд базується на технологіях JavaEE, Spring, Node.js, Hibernate, MySQL.

При виборі платформи для розробки важливо спочатку визначитися із завданнями, які вона виконуватиме. Всі IDE відрізняються кількістю

вбудованих у них інструментів. Основні критерії вибору середовища: підтримка кількох мов програмування; можливість кастомізації; юніт-тестування; бібліотека вихідного коду; засоби діагностики помилок та ведення їх звітності; автодоповнення коду; інтеграції та плагіни; пошук за кодом; діаграми ієрархії.

Для розробки проєкту була обрана Aptana Studio 3. У ній можна настроїти графічний інтерфейс під себе. Є й такі функції: вбудований термінал, інтеграція з Git, відладчик, а також майстер розгортання. Автодоповнення в ній доступне для HTML, CSS та JavaScript.

Для розробки системи було обрано найбільш підходящі програмні засоби. Як СУБД обрано MySQL. Мови програмування: для верстки HTML, JavaScript, CSS, JavaScript, бібліотека jQuery, для серверної розробки PHP. Як серверну мову програмування було обрано PHP.

3 ПРОЄКТУВАННЯ ВЕБСИСТЕМИ ТРЕНАЖЕРНОГО ЗАЛУ

3.1 Розробка структури вебсистеми

В даній роботі вебсистема складається з трьох основних компонентів (рис. 3.1):

– клієнтська частина вебпрограми – це графічний інтерфейс. Це те, що відображається на сторінці. Графічний інтерфейс відображається у браузері. Користувач взаємодіє з вебсистемою саме через браузер, натискаючи за посиланнями та кнопками;

– серверна частина вебсистеми – це програма або скрипт на сервері, що обробляє запити користувача (точніше, запити браузера);

– база даних. У випадку форуму або блогу, дані, що зберігаються в БД, – це пости, коментарі, новини, і так далі. База даних розміщується на сервері.

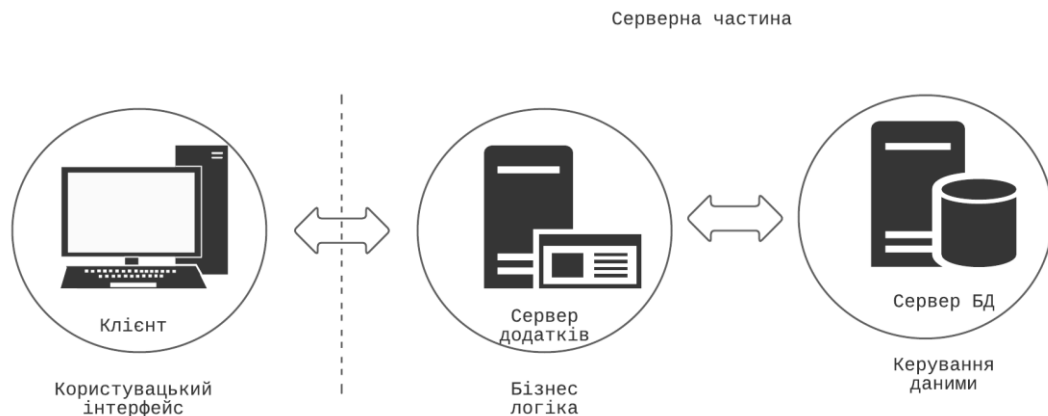


Рисунок 3.1 – Структура вебсистеми

Надалі розробимо структуру саме вебсайту. Правильна структура сайту дає наступні переваги [13]:

– спрощує та підвищує якість взаємодії користувача з сайтом (збільшуючи ймовірність користувача стати постійним клієнтом), покращує

поведінковий досвід (глибину, час перегляду тощо), позитивно впливає на позиції та трафік із пошукових систем;

- покращує та прискорює індексацію сайту (спрощуючи маршрут руху пошуковим роботам);

- допомагає коректно розподіляти вагу по всіх сторінках відповідно до їх важливості і для користувача, і для просування. Таким чином, це також впливає на ранжування та позиції сайту у видачі;

- дає можливість охоплювати більше запитів (створення додаткових розділів, категорій та посадкових сторінок під вузькі групи запитів дає можливість ранжуватися за великою кількістю ключових слів і залучати більше органічного трафіку);

- зменшує кількість технічних помилок на сайті (дозволить уникнути великої кількості дублів сторінок, що дозволить заощадити час та гроші на просування);

- впливає на відображення додаткових посилань на сайт у сніпеті («анонсі» сторінки в пошуку, що мотивує відвідування сайту відвідувачем).

Перелічимо стандартні вимоги для структури сайту [15] - [16]:

- наявність навігаційних ланцюжків (хлібних крихт);
- дотримання правила 3-х кліків (рівень вкладеності вебсторінок не перевищує 4);

- використання категорій для угруповання пов'язаного контенту;
- використання HTML та CSS для навігації;
- наявність посилань на головну з усіх сторінок сайту;
- незмінність структури при збільшенні чи зменшенні кількості категорій та підкатегорій;

- наявність картки сайту для відвідувачів.

Вимоги пошукової системи Google до структури сайту:

- створити гранично просту, логічну та зрозумілу для людини структуру сайту;

- використовувати людино–зрозумілі URL. Google орієнтується в їх визначеннях релевантності сторінки;
- використовувати пунктуацію у посиланнях. Google рекомендує ставити дефіс («–»), а не нижній прочерк («_»);
- не використовувати складні та довгі URL. Оптимальна довжина адреси сторінки до 75 символів.

Структура створеного сайту показано рисунку 3.2.

Сайт складається з наступних частин:

- головне меню навігації (шапки);
- кількох сторінок з інформацією та зображеннями.

За допомогою головного меню навігації можна переміститися на певну сторінку сайту (головна сторінка, тренування, новини, про нас, контакти, оформлення абонементу).



Рисунок 3.2 – Структура вебсайту

Таким чином спроектована структура вебсайту складається з 5 сторінок та двох форм.

3.2 Розробка бази даних

База даних вебсистеми тренажерного залу зберігає інформацію про абонементи (табл. 3.1), клієнтів, які оформили абонементи (табл. 3.2), тренерів (табл. 3.3), публікації блогу (табл. 3.4).

Таблиця 3.1 – Сутності таблиці «subscription»

Сутність	Тип	Припустиме значення	Ключ
subId	числовий	Not null	+
clientId	числовий	Not null	
trainerId	числовий	null	
subName	текстовий	–	
subText	текстовий	null	
subStartDate	дата	–	
subEndDate	дата	–	
subPrice	числовий	0	

Таблиця 3.2 – Сутності таблиці «client»

Сутність	Тип	Припустиме значення	Ключ
clientId	числовий	Not null	+
clientName	текстовий	Not null	
clientSurname	текстовий	Not null	
clientTelephone	текстовий	Not null	
clientEmail	текстовий	Not null	

Таблиця 3.3 – Сутності таблиці «trainer»

Сутність	Тип	Припустиме значення	Ключ
trainerId	числовий	Not null	+
trainerName	текстовий	Not null	
trainerSurname	текстовий	Not null	
trainerTelephone	текстовий	Not null	
trainerEmail	текстовий	Not null	
trainerPhoto	текстовий	null	
trainerDescription	текстовий	null	

Таблиця 3.4 – Сутності таблиці «blog»

Сутність	Тип	Припустиме значення	Ключ
blogId	числовий	Not null	+
trainerId	числовий	null	
blogImage	текстовий	Not null	
blogText	текстовий	Not null	
blogTitle	текстовий	Not null	
blogMeta	текстовий	Not null	
blogKeywords	текстовий	Not null	

Модель бази даних, розробленої в phpMyAdmin, представлено на рисунку 3.3.

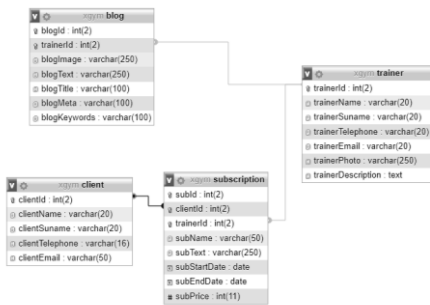


Рисунок 3.3 – Модель бази даних

3.3 Розробка дизайну вебсайту

Залежно від бюджету та основних завдань, дизайн спортивних сайтів може бути максимально простим, статичним чи яскравим, інтерактивним. Головне правило — створити єдиний візуальний стиль із соціальними мережами компанії, загальним оформленням тренажерних залів. Дизайн має бути агресивний та енергійний. Контрастні тони, підтягнуті тіла, великі написи. Повинна бути відповідність та послідовність.

Також при створенні дизайну сайту спорт-клубу та виборі зображень слід враховувати цільову аудиторію [11]:

- використовуйте авторські фотографії замість шаблонів з Інтернету. Так користувачі зможуть познайомитись із реальними інструкторами, тренажерами, доступними у вашому залі. Це підвищить довіру;

- уникайте «хардкорних» зображень стилю CrossFit та бруталного оформлення, якщо ваша аудиторія не є професійними спортсменами. Досить просто використовувати фотографії струнких, підтягнутих жінок та чоловіків;

- переконайтеся, що сторінка правильно відображається на смартфонах та планшетах, особливо якщо ваша аудиторія – молодь, або ви просуваєте сайт через соціальні мережі.

Головна сторінка має відображати максимально корисну та актуальну інформацію про роботу тренажерного клубу. Можливість прочитати про види тренувань, мотиваційний девіз, щоб стати клієнтом клубу, інформацію про абонементи, презентацію тренерів, розрахунок IBM, хедер та футер сайту (рис. 3.4). Хедер – це меню з посиланнями на сторінки сайту і логотипом. Футер сайту складається з рядка із соціальними мережами клубу, номером телефону та кнопкою «підписатися» на повідомлення від клубу. При натисканні на певний значок соціальних мереж користувача перенаправляє на офіційну сторінку клубу в обраній соціальній мережі.

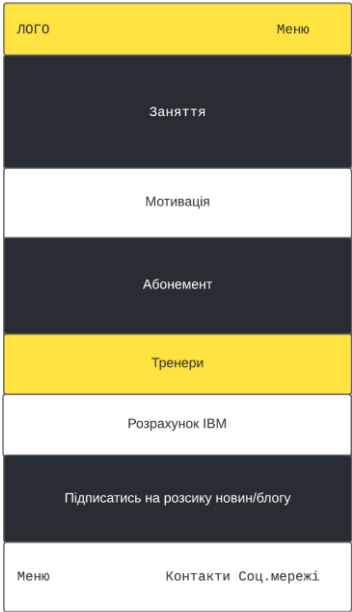


Рисунок 3.4 – Дизайн сторінки «Головна»

Сторінка «Про нас» відображає інформацію про персонал тренажерного залу та його напрям діяльності. Блоки, які до неї входять: текстовий контент, статистика, інформація про тренерів, форма зв'язку з тренером, хедер та футер сайту (рис. 3.5).

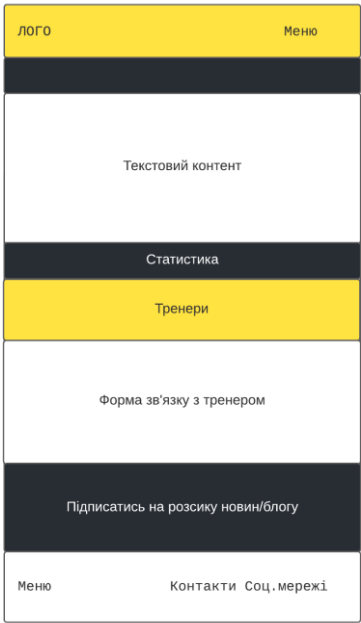


Рисунок 3.5 – Дизайн сторінки «Про нас»

Сторінка «Тренування». Один з основних розділів сайту з інформацією про послуги клубу (рис. 3.6).

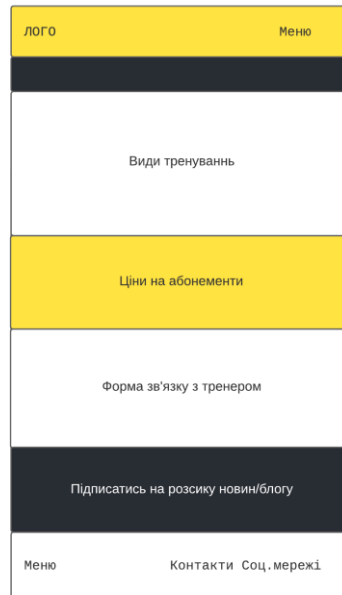


Рисунок 3.6 – Дизайн сторінки «Тренування»

Сторінка сайту «Новини» виглядає в стилі блогу тренажерного залу (рис. 3.7). Блоки цієї сторінки поділяються на нові публікації, категорії, архів публікацій, теги публікацій.

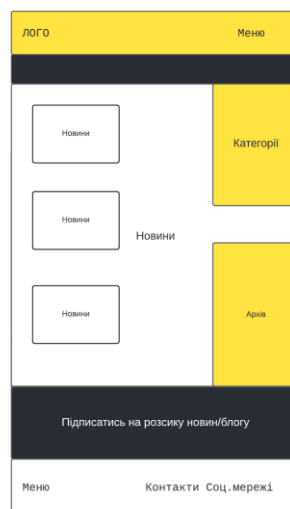


Рисунок 3.7 – Сторінка «Новини»

Контакти – це останній розділ головної сторінки сайту (рис. 3.8).



Рисунок 3.8 – Дизайн сторінки «Контакти»

На сторінці можливо, залишити своє питання безпосередньо через спеціальну форму.

Форма бронювання абонементу наведена на рисунку 3.9.

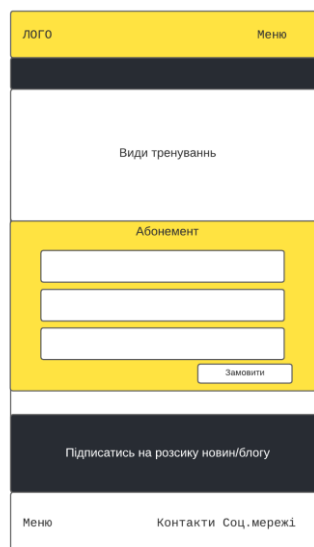


Рисунок 3.9 – Форма бронювання абонементу

На кожній сторінці знаходиться точна адреса тренувального центру, рядок із соціальними мережами клубу, номером телефону та кнопка «підписатися» на повідомлення від клубу. При натисканні на певний значок соціальних мереж користувача перенаправляє на офіційну сторінку клубу в обраній мережі.

3.4 Висновки за розділом

В даній роботі вебсистема складається з трьох основних компонентів: клієнтська частина вебпрограми – це графічний інтерфейс; серверна частина вебсистеми – це програма або скрипт на сервері, що обробляє запити користувача (точніше, запити браузера); база даних.

Сайт складається з наступних частин: головне меню навігації (шапки); кількох сторінок з інформацією та зображеннями. За допомогою головного меню навігації можна переміститися на певну сторінку сайту (головна сторінка, тренування, новини, про нас, контакти, оформлення абонементу).

База даних вебсистеми тренажерного залу зберігає інформацію про абонементи, клієнтів, які оформили абонементи, тренерів, публікації блогу.

Дизайн головної сторінки відображає максимально корисну та актуальну інформацію про роботу тренажерного клубу. Можливість прочитати про види тренувань, мотиваційний девіз, щоб стати клієнтом клубу, інформацію про абонементи, презентацію тренерів, розрахунок IBM, хедер та футер сайту. Хедер – це меню з посиланнями на сторінки сайту і логотипом. Футер сайту складається з рядка із соціальними мережами клубу, номером телефону та кнопкою «підписатися» на повідомлення від клубу. При натисканні на певний значок соціальних мереж користувача перенаправляє на офіційну сторінку клубу в обраній соціальній мережі. Сторінка «Про нас» відображає інформацію про персонал тренажерного залу та його напрям діяльності. Блоки, які до неї входять: текстовий контент, статистика, інформація про тренерів, форма зв'язку з тренером, хедер та

футер сайту. Сторінка «Тренування». Один з основних розділів сайту з інформацією про послуги клубу. Сторінка сайту «Новини» виглядає в стилі блогу тренажерного залу. Блоки цієї сторінки поділяються на нові публікації, категорії, архів публікацій, теги публікацій. Контакти – це останній розділ головної сторінки сайту.

4 РОЗРОБКА ВЕБСАЙТУ ТРЕНАЖЕРНОГО ЗАЛУ

4.1 Керівництво програміста

Метатеги в документах HTML надають корисну інформацію про документ, включаючи опис, ключові слова, автора, дати змін та близько 90 інших полів (рис. 4.1).

```
<!DOCTYPE html>
<html lang="zxx">
<head>
  <title>Gym - Fitness </title>
  <meta charset="UTF-8">
  <meta name="description" content=" Gym Fitness ">
  <meta name="keywords" content="fitness, gym">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
```

Рисунок 4.1 – Метатеги сторінок

Стилі можуть бути створені декількома способами, що дозволяє різного ступеня повторного використання та обсягу, коли вони вказані у вихідному документі (рис. 4.2). Зовнішні таблиці стилів можна повторно використовувати на сторінках. Вбудовані таблиці стилів застосовуються до всього документа, де вони вказані. Вбудовані стилі застосовуються лише до окремого HTML-елемента, на якому вони вказані.

З опернсорсних стилів були використані bootstrap, font-awesome, owl, flaticon, slicknav.

```
<!-- Google font -->
<link href="https://fonts.googleapis.com/css?family=PT+Sans:400,400i,700,700i&display=swap"

<!-- Stylesheets -->
<link rel="stylesheet" href="css/bootstrap.min.css"/>
<link rel="stylesheet" href="css/font-awesome.min.css"/>
<link rel="stylesheet" href="css/owl.carousel.min.css"/>
<link rel="stylesheet" href="css/flaticon.css"/>
<link rel="stylesheet" href="css/slicknav.min.css"/>

<!-- Main Stylesheets -->
<link rel="stylesheet" href="css/style.css"/>
```

Рисунок 4.2 – Підключення стилів

Bootstrap – це самовпевнена структура для HTML, CSS та Javascript. Він містить базовий стиль та функціональність для того, що стало прийнятим елементами [User Interface], наприклад елементами форми, кнопками, модальними вікнами та елементами навігації. Bootstrap – це адаптивна вебінфраструктура, тобто вона призначена для адаптації макета та дизайну для розмірів екрану великих та малих розмірів, таких як мобільні пристрої, планшети та настільні комп'ютери, все в одній базі коду.

Font Awesome це масштабовані векторні іконки, які можна легко персоналізувати — розмір, колір, тіні і все, що можна зробити силами CSS. До цього додається безкоштовна база іконок flaticon.

SlickNav – адаптивне меню. Плагін для jQuery.

Хедер сторінки вебсайту складається з логотипу, який є посиланням на головну сторінку, та меню навігації (рис. 4.3).

```
<!-- Header section -->
<header class="header-section">
  <a href="index" class="site-logo">
    
  </a>
  <ul class="main-menu">
    <li><a href="index">Головна</a></li>
    <li><a class="active" href="about">Про нас</a></li>
    <li><a href="classes">Тренування</a></li>
    <li><a href="blog">Новини</a></li>
    <li><a href="contact">Контакти</a></li>
    <li class="header-right">
      <div class="hr-box">
        
        <h6>Запоріжжя, Україна</h6>
      </div>
    </li>
  </ul>
</header>
<div class="clearfix"></div>
<!-- Header section end -->
```

Рисунок 4.3 – Хедер вебсайту тренажерного залу

Футер вебсторінок сайту тренажерного залу розроблен на рисунку 4.4

```

<footer class="footer-section set-bg" data-setbg="img/footer-bg.jpg">
  <div class="container">
    <div class="row">
      <div class="col-lg-3 col-sm-6">
        <div class="footer-widget">
          <h4>Місцезнаходження</h4>
          <div class="fw-info-box">
            
            <div class="fw-info-text">
              <p>Запоріжжя, Україна</p>
            </div>
          </div>
        </div>
      </div>
      <div class="col-lg-3 col-sm-6">
        <div class="footer-widget">
          <h4>Телефон</h4>
          <div class="fw-info-box">
            
            <div class="fw-info-text">
              <p>066 111-11-11</p>
            </div>
          </div>
        </div>
      </div>
      <div class="col-lg-3 col-sm-6">
        <div class="footer-widget">
          <h4>E-mail</h4>
          <div class="fw-info-box">
            
            <div class="fw-info-text">
              <p>Contact@xgym.com</p>
              <p>www.xgym.com</p>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```

Рисунок 4.4 – Футер сайту тренажерного залу

Контейнерний вебдизайн для розміщення окремих фрагментів вмісту вебсторінок (рис. 4.5). Окремі контейнери створюються для заголовка вебсайту, смуги навігації, основного вмісту та відомостей про авторські права. Основний вміст має складну структуру і складається з багатьох незалежних частин, для кожної з них створюється окремий контейнер. Для визначення різних параметрів блокових контейнерів передбачені спеціальні атрибути стилю CSS. До таких параметрів відносяться розміри (ширина та висота), розташування контейнерів та їх поведінка при переповненні. Також задані для контейнерів колір фону, відступи та рамки, щоб їх [10].

```

<div class="container">
  <div class="row justify-content-md-center">
    <div class="col-lg-5">
      <div class="section-title text-center">
        <h2><span>Ми пілкуємось</span> про ваше тіло</h2>
      </div>
    </div>
  </div>
  <div class="row">
    <div class="col-lg-4">
      <div class="about-text">
        <p>Головний плюс, який дарує регулярний спорт – здоров'я та гар
      </div>
    </div>
    <div class="col-lg-4">
      <div class="about-text">
        <p>Відвідування тренажерної зали – тренування сили волі. Якщо в
      </div>
    </div>
    <div class="col-lg-4">
      <div class="about-img">
        
      </div>
    </div>
  </div>
  <div class="text-center pt-4">
    <a href="classes" class="site-btn">Почати тренування</a>
  </div>
</div>
</section>

```

Рисунок 4.5 – Контейнер зі стратегією роботи тренажерного залу

Блок, який використовує інтерактивні елементи для комунікації з клієнтом (рис. 4.6).

```

<div class="achievement-section set-bg" data-setbg="img/achievement-bg.jpg">
  <div class="container">
    <div class="row">
      <div class="col-lg-3 col-sm-6">
        <div class="circle-progress" data-cptitle="Щасливі клієнти" data-cpid="id-1" data-cpvalue="50" data-cpcolo
      </div>
      <div class="col-lg-3 col-sm-6">
        <div class="circle-progress" data-cptitle="Зменшили вагу" data-cpid="id-2" data-cpvalue="75" data-cpcolor=
      </div>
      <div class="col-lg-3 col-sm-6">
        <div class="circle-progress" data-cptitle="Задоволені результатом" data-cpid="id-3" data-cpvalue="100" dat
      </div>
      <div class="col-lg-3 col-sm-6">
        <div class="circle-progress" data-cptitle="Збільшили енергійність" data-cpid="id-4" data-cpvalue="25" data
      </div>
    </div>
  </div>
</div>

```

Рисунок 4.6 – Контейнер з мотиваційної інформацією

Загальна інформація для знайомства з тренером. Використовуються класи для вставки зображень та блочного опису тексту (рис. 4.7).

```
<div class="container">
  <div class="section-title text-center">
    <h2>Познайомитись з <span>Тренером</span></h2>
  </div>
  <div class="row">
    <div class="col-lg-3 col-sm-6">
      <div class="trainer-item">
        <div class="trainer-pic">
          
        </div>
        <h4>Михайло Прозь</h4>
        <p>Тренер (персональний тренер) має ширші знання, ніж фітнес-інструктор, повинен вміти
      </div>
    </div>
    <div class="col-lg-3 col-sm-6">
      <div class="trainer-item">
        <div class="trainer-pic">
          
        </div>
        <h4>Юлія Квітка</h4>
        <p>Фітнес-інструктор повинен добре розумітися на таких питаннях, як будова людського т
      </div>
    </div>
  </div>
</div>
```

Рисунок 4.7 – Контейнер «Знайомство з тренерами»

Насамперед JavaScript широко використовується у фронтенд-розробці. Ця мова разом з HTML і CSS входить до базового набору інструментів фронтендера. На JavaScript створюються програми, які виконуються у браузері клієнта. Вони забезпечують інтерактивність вебсайтів. Підключення скриптів наведено на рисунку 4.8. Серед опенсорсних скриптів використані jquery, bootstrap, slicknav, carousel, circle-progress. jQuery — це підібраний набір інтерфейсів, ефектів, віджетів і тем, створених на основі бібліотеки JavaScript.

```
<script src="js/jquery-3.2.1.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<script src="js/jquery.slicknav.js"></script>
<script src="js/owl.carousel.min.js"></script>
<script src="js/circle-progress.min.js"></script>
<script src="js/main.js"></script>
```

Рисунок 4.8 – Підключення скриптів

В основному стилі сайту використані три кольори: чорний, жовтий, білий. Класи стилів головних тегів наведені на рисунку 4.9.

```
html,
body {
    height: 100%;
    font-family: "PT Sans", sans-serif;
    -webkit-font-smoothing: antialiased;
    font-smoothing: antialiased;
}

h1,
h2,
h3,
h4,
h5,
h6 {
    margin: 0;
    color: #353535;
    font-weight: 700;
    line-height: 1;
}
```

Рисунок 4.9 – Класи стилів головних тегів

Стилі кнопок наведені на рисунку 4.10.

```
.site-btn {
    display: inline-block;
    border: none;
    font-size: 18px;
    font-weight: 700;
    min-width: 228px;
    padding: 20px 40px;
    background: #fcff18;
    color: #353535;
    line-height: normal;
    cursor: pointer;
    text-align: center;
}
```

Рисунок 4.10 – Стиль кнопок

Сторінка будь-якого сайту або вебпрограми не завантажується миттєво. На завантаження та відображення сторінки потрібно деякий час. При цьому сторінка при її завантаженні може змінюватись. Зазвичай це відбувається під час завантаження стилів, шрифтів, картинок. Щоб цей непривабливий момент приховати від користувача, можна на час завантаження сторінки відобразити користувачеві анімований прелоадер. А

після того, як сторінка повністю завантажиться, його прибрати. Основна мета прелоадера – це покращити враження користувача про сайт (рис. 4.11).

```
#preloader {  
    position: fixed;  
    width: 100%;  
    height: 100%;  
    top: 0;  
    left: 0;  
    z-index: 999999;  
    background: #000;  
}  
  
.loader {  
    width: 40px;  
    height: 40px;  
    position: absolute;  
    top: 50%;  
    left: 50%;  
    margin-top: -13px;  
    margin-left: -13px;  
    border-radius: 60px;  
    animation: loader 0.8s linear infinite;  
    -webkit-animation: loader 0.8s linear infinite;  
}
```

Рисунок 4.11 – Прелоадер сорінки

Стилі головного меню та логотипу наведені на рисунку 4.12. Сайт також оптимізований під мобільну версію.

```

] .site-logo {
    display: inline-block;
    padding: 15px 35px;
-}

] .main-menu {
    list-style: none;
    float: right;
-}

] .main-menu li {
    display: inline-block;
    padding-top: 33px;
-}

] .main-menu li a {
    font-size: 20px;
    font-weight: 700;
    color: #353535;
    margin-right: 25px;
    padding: 2px 13px;
    -webkit-transition: all 0.3s;
    transition: all 0.3s;
-}

] .main-menu li a.active,
] .main-menu li a:hover {
    background: #ffff18;
-}

```

Рисунок 4.12 – Стилі меню

Блок стилів для підписки на розсилку, яка знаходиться в футері сайту (рис. 4.13).

```

.newsletter-section {
    padding: 84px 0;
}

.newsletter-section .section-title {
    padding-top: 8px;
}

.newsletter-form {
    margin-left: -55px;
}

.newsletter-form input {
    width: calc(100% - 228px);
    height: 63px;
    padding: 10px 35px;
    float: left;
    border: none;
}

```

Рисунок 4.13 – Підписка на розсилку

Розроблені стилі футеру для різних версій сайту (рис. 4.14).

```

.footer-section {
    padding-top: 106px;
    padding-bottom: 23px;
}

.footer-widget {
    margin-bottom: 50px;
}

.footer-widget h4 {
    color: #fff;
    margin-bottom: 30px;
}

.footer-widget .fw-info-box img {
    float: left;
    margin-right: 12px;
    padding-top: 7px;
}

.footer-widget .fw-info-box p {
    color: #b5babc;
    line-height: 1.8;
    margin-bottom: 0;
}

```

Рисунок 4.14 – Стили футеру

Функція завантаження прелоадера наведена на рисунку 4.15.

```

$(window).on('load', function() {
    /*-----
    |       Preloder
    |-----*/
    $(".loader").fadeOut();
    $("#preloder").delay(400).fadeOut("slow");
});

```

Рисунок 4.15 – Функція завантаження прелоадера

Функція завантаження іконок та меню описана на рисунку 4.16.

```

/*-----
Navigation
-----*/

$(".main-menu").slicknav({
  appendTo: '.header-section',
  allowParentLinks: true
});

/*-----

```

Рисунок 4.16 – Функція завантаження іконок та меню

Скрипт заднього фону сторінки (рис. 4.17).

```

/*-----
Background Set
-----*/

$('.set-bg').each(function() {
  var bg = $(this).data('setbg');
  $(this).css('background-image', 'url(' + bg + ')');
});

```

Рисунок 4.17 – Скрипт заднього фону сторінки

Скрипт розрахунку маси тіла людини (рис. 4.18).

```

/*-----
Index calculator
-----*/
function calcIndex () {

  var cm = Number(document.getElementById("IndexHeight").value);
  var kg = Number(document.getElementById("IndexKg").value);
  var result = kg / (cm * cm);

  var Index = Math.round(result * 100) / 100;
  if (Index < 18.5) {
    document.getElementById("IndexResult").value = Index + " (Underkg)";
  } else if (Index >= 17 && Index <= 24) {
    document.getElementById("IndexResult").value = Index + " (Normal)";
  } else if (Index >= 24 && Index <= 31) {
    document.getElementById("IndexResult").value = Index + " (Big)";
  } else if (Index > 31) {
    document.getElementById("IndexResult").value = Index + " (OverKg)";
  } else {
    document.getElementById("IndexResult").value = Index;
  }
}

$('#Index-suIndext').on('click', function(){
  computeIndex();
});

```

Рисунок 4.18 – Скрипт розрахунку маси тіла людини

Повний текст файлу скриптів наведений в додатку А.

4.2 Керівництво користувача

За допомогою головного меню навігації можна переміститися на певну сторінку сайту (головна сторінка, тренування, новини, про нас, контакти, оформлення абонементу). При наведенні мишкою на ім'я розділу і клацання по ньому, користувача переносить на потрібний йому розділ з інформацією, що його цікавить (рисунок 4.19).

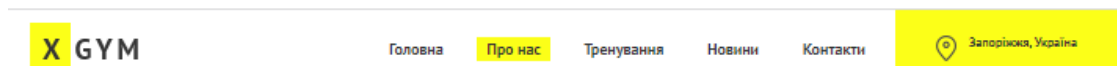


Рисунок 4.19 – Меню сайту тренажерного залу

На кожній сторінці сайту розміщується блок футеру. Він складається з декількох частин: підписка на розсилку новин від тренажерного залу, місцезнаходження, контактна інформація, посилання на соціальні мережі, другорядне меню (рис. 4.20).

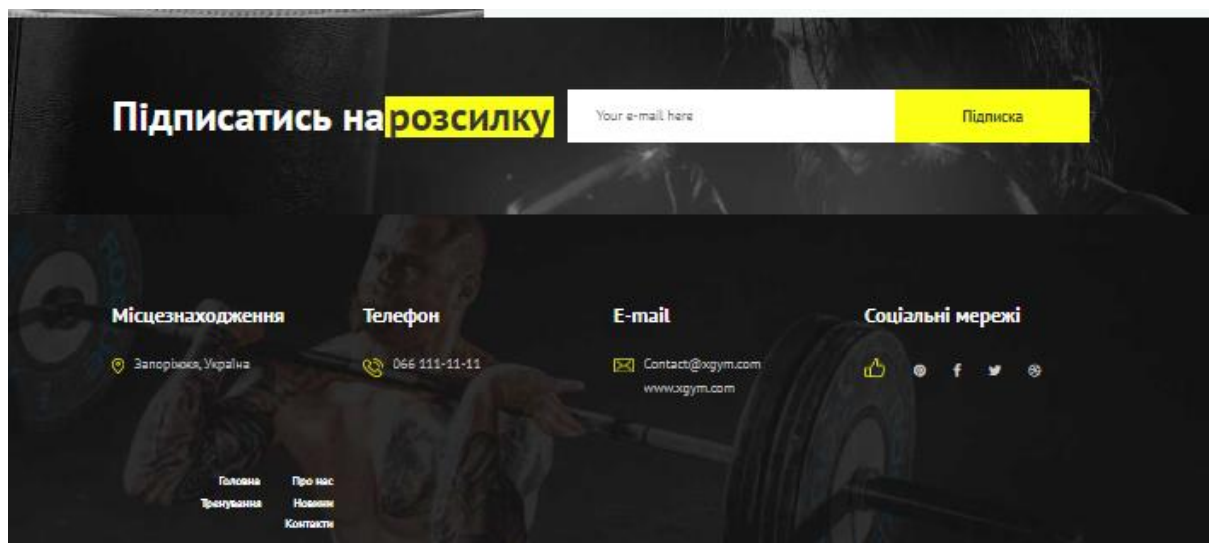


Рисунок 4.20 – Футер сайту тренажерного залу

Головна сторінка. Цей розділ містить всю основну інформацію про клуб. При переході на сайт це головна сторінка, яка привертає увагу користувачів і збирає основні дані про потенційних клієнтів клубу. Розділ має кілька сторінок, між якими користувач може переміщатися шляхом прокручування коліщатком миші (рис 4.21).

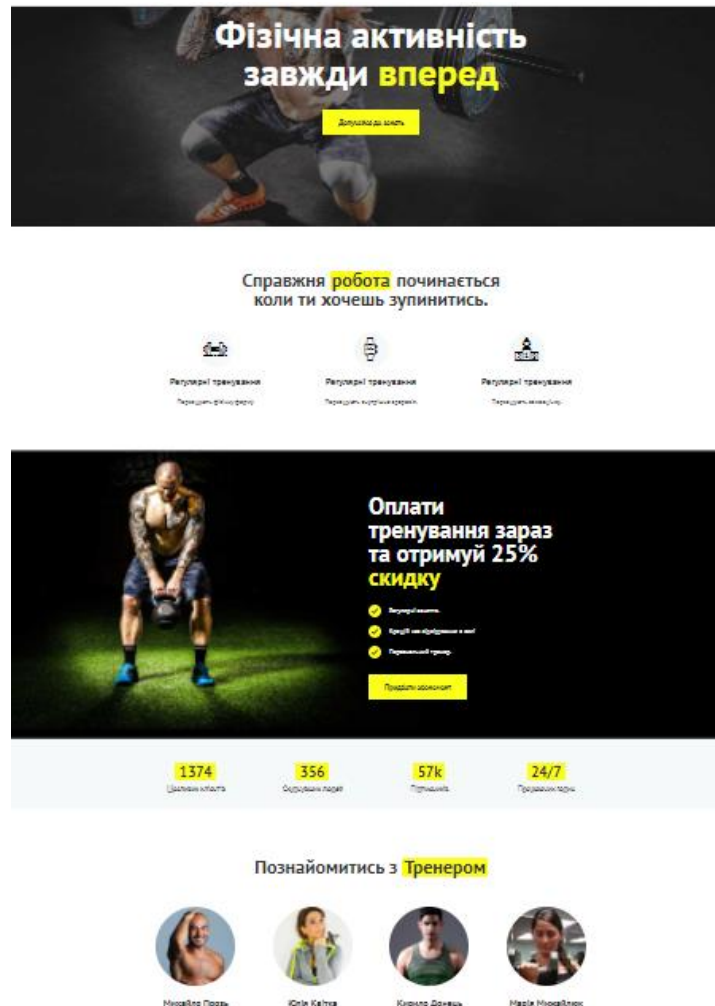


Рисунок 4.21– Головна сторінка сайту тренажерного залу

Головна сторінка складається із розділів:

- девіз клубу та кнопка, яка переходить до видів тренувань (рис. 4.22);
- мотиваційний контент для нових користувачів сайту (рис 4.22);
- акційна інформація (рис. 4.23);
- знайомство з тренерами (рис. 4.24);
- розрахунок показника маси тіла (рис. 4.25);

– футер сайту (рис. 4.20).

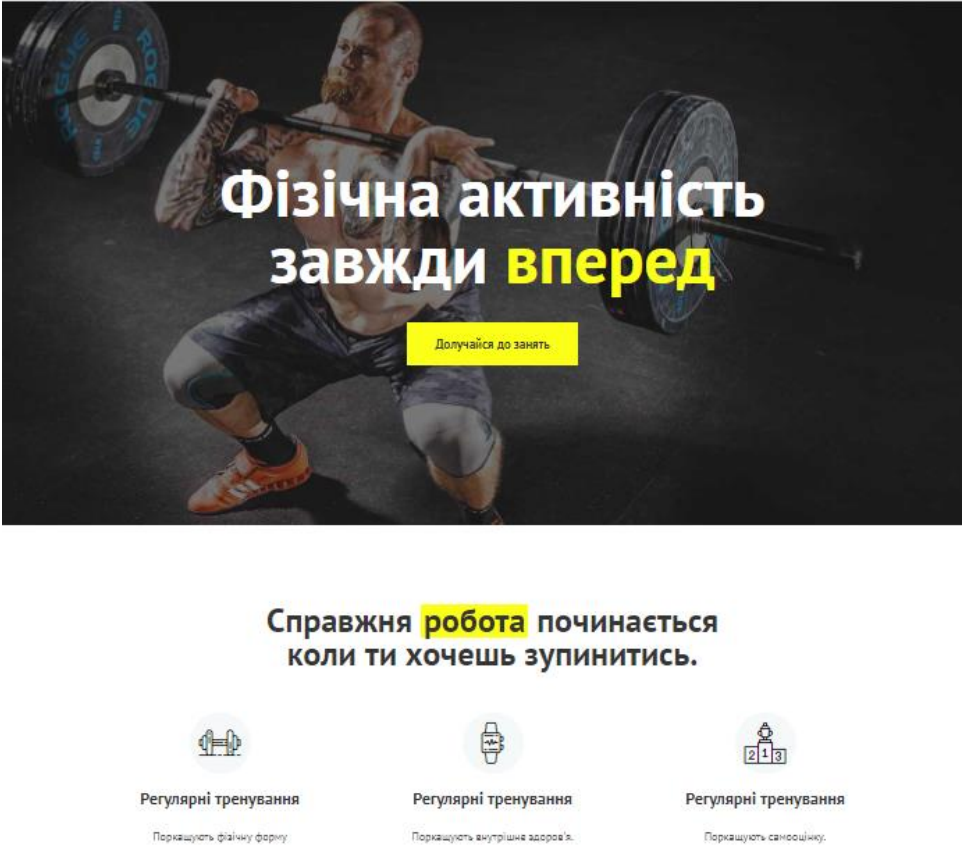


Рисунок 4.22 – Девіз та мотиваційний контент

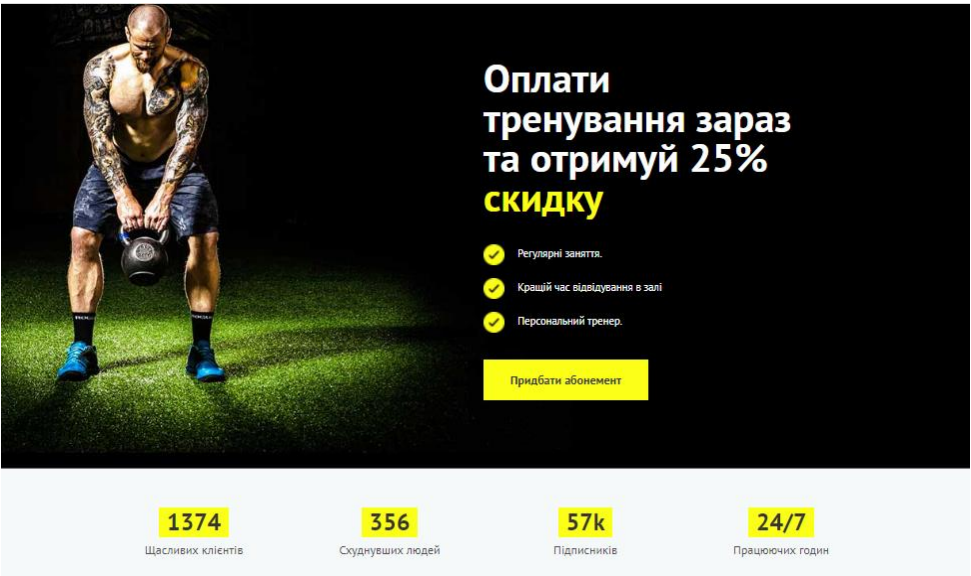


Рисунок 4.23 – Акційна інформація

Познайомитись з Тренером



Михайло Прозь

Тренер (персональний тренер) має ширші знання, ніж фітнес-інструктор, повинен вміти побудувати індивідуальну програму відповідно до фізіологічних особливостей клієнта та бажаного результату (у тому числі для особливих груп клієнтів із захворюваннями серцево-судинної системи, опорно-рухового апарату, вагітних жінок і т.д.). У нього можна отримати компетентну консультацію щодо правильного харчування.



Юлія Квітка

Фітнес-інструктор повинен добре розумітися на таких питаннях, як будова людського тіла, знати, як працюють системи та органи при фізичних навантаженнях, знати основні види тренувань та особливості впливу тих чи інших вправ на організм.



Кирило Донець

Тренер (персональний тренер) має ширші знання, ніж фітнес-інструктор, повинен вміти побудувати індивідуальну програму відповідно до фізіологічних особливостей клієнта та бажаного результату (у тому числі для особливих груп клієнтів із захворюваннями серцево-судинної системи, опорно-рухового апарату, вагітних жінок і т.д.). У нього можна отримати компетентну консультацію щодо правильного харчування.



Марія Михайлюк

Майстер-тренер - це тренер, який повинен знати про спорт все, а також знатися на основах психології, педагогіки, вміти продати свій продукт, тобто. «персональне тренування», навчити персонал нижчої категорії. Саме вони готують клієнтів до змагань та проводять спеціальні тренування. Обов'язково є наявність вищої спортивної освіти та підтвердження категорії з отриманням відповідного сертифікату.

Рисунок 4.24 – Знайомство з тренерами



Розрахунок вашого BMI

Індекс маси тіла (BMI) – величина, що дозволяє оцінити ступінь відповідності маси людини та її зростання і тим самим побічно судити про те, чи є маса недостатньою, нормальною чи надмірною.

Маса (кг)

Зріст (м)

Розрахунок

Індекс маси тіла слід застосовувати з обережністю виключно для орієнтовної оцінки.

Рисунок 4.25 – Розрахунок показника маси тіла

Сторінка «Про нас» відображає інформацію про персонал тренажерного залу та його напрям діяльності (рис. 4.26, рис. 4.27).

Відмінною характеристикою є наявність блоку з показниками статистики клієнтів тренажерного залу (рис. 4.28).

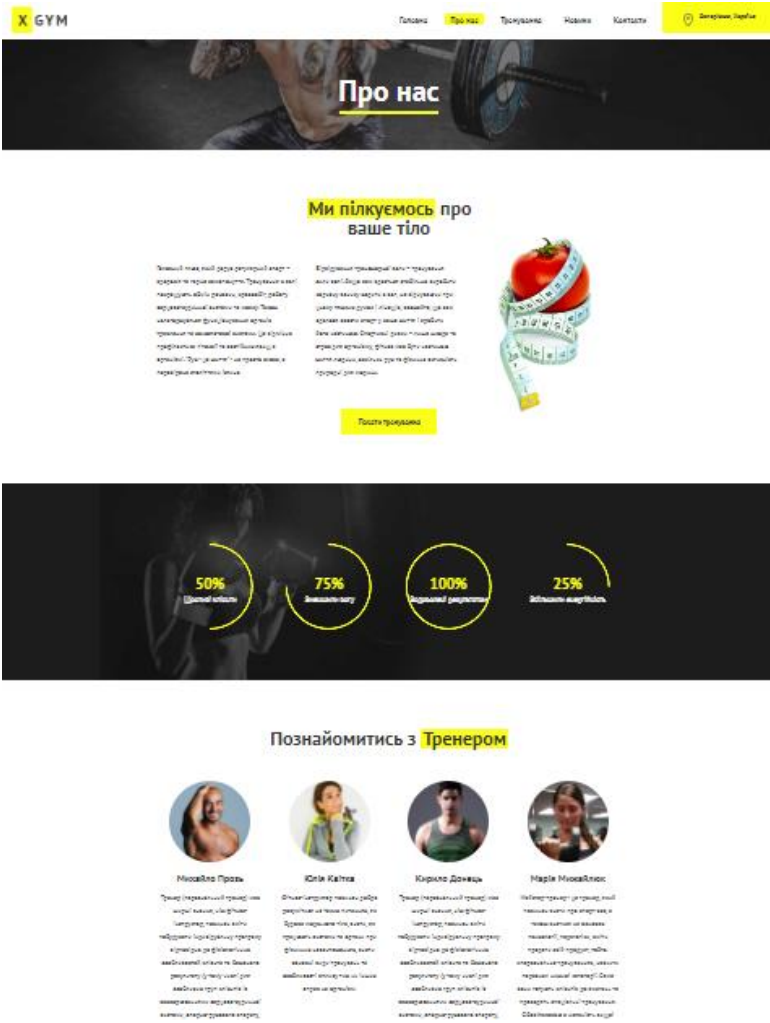


Рисунок 4.26 – Сторінка «Про нас»

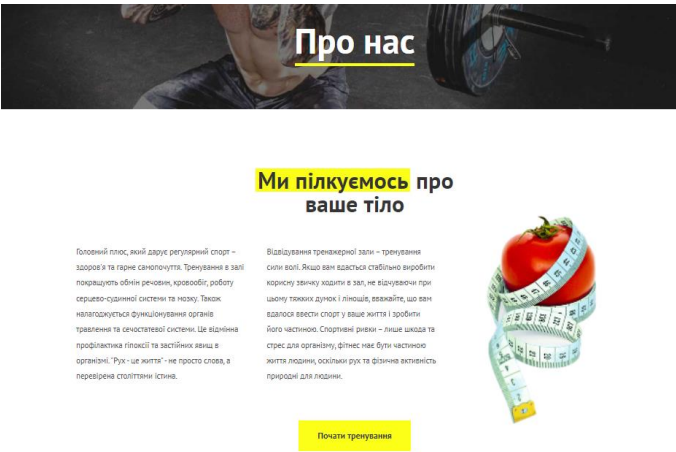


Рисунок 4.27 – Головний напрям роботи тренажерного залу



Рисунок 4.28 – Показники статистики клієнтів тренажерного залу

Унікальна послуга тренажерного залу – це замовити сертифікат консультацій з професійним тренером (рис. 4.29). Мається на увазі онлайн підтримка людини, якщо навіть вона займається в іншому клубі. Професійний тренер надалі розробить чи зкорегує курс тренувань та харчування для отримання результату.

Отримати сертифікат до Професійного тренера

Достиження поставленого результату залежить від того, подобається Вам ті чи інші вправи чи ні. Допустимо у Вас «не лежить душа» до певної вправи, то, як наслідок, Ви його і не робитимете або ж не викладатиметеся під час його. У таких ситуаціях найкраще замінити вправу на іншу з альтернативною віддачею. При плануванні програми тренувань Ви повинні ґрунтуватися на власному здоров'ї, фізичній підготовці, вашому особистому темпі, віці, статурі та на багато іншого. При неправильному рішенні відкинути ті чи інші вправи Ви ризикуєте не досягти оптимального результату, оскільки всі вправи взаємопов'язані між собою і впливають один на одного.

Ваше ім'я

Ваш e-mail

Ваша мета

Надіслати

Рисунок 4.29 – Замовити сертифікат консультацій з професійним тренером

Сторінка «Тренування». Один з основних розділів сайту з інформацією про послуги клубу (рис. 4.30).

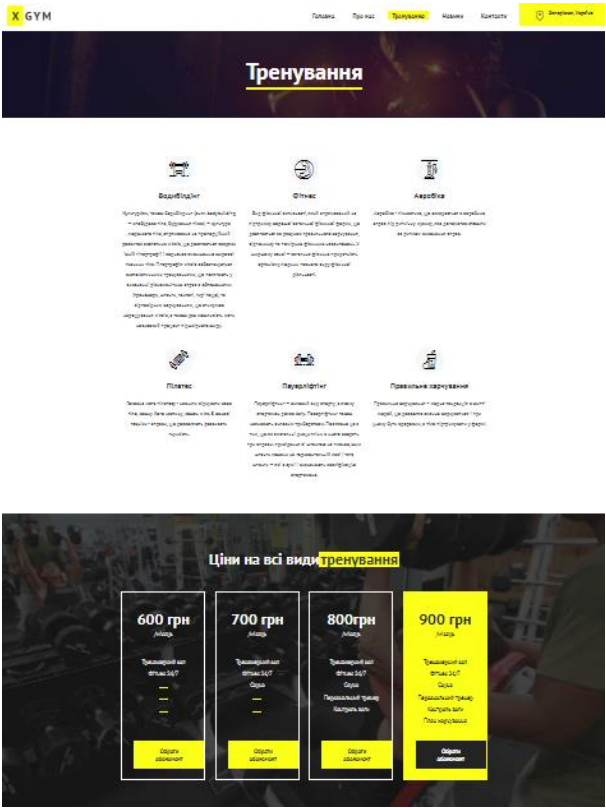


Рисунок 4.30 – Сторінка «Тренування»

Дана сторінка розроблена з декількома блоками. Перший блок описує види тренувань, які можливі в даному тренажерному залі (рис. 4.31).

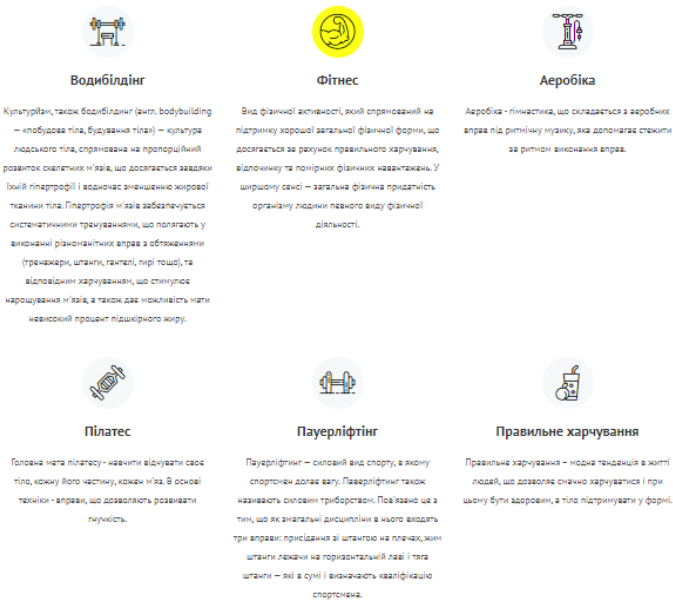


Рисунок 4.31 – Види тренувань

Наступний блок з цінами абонементів та категоріями послуг, які входять у вартість (рис. 4. 32).

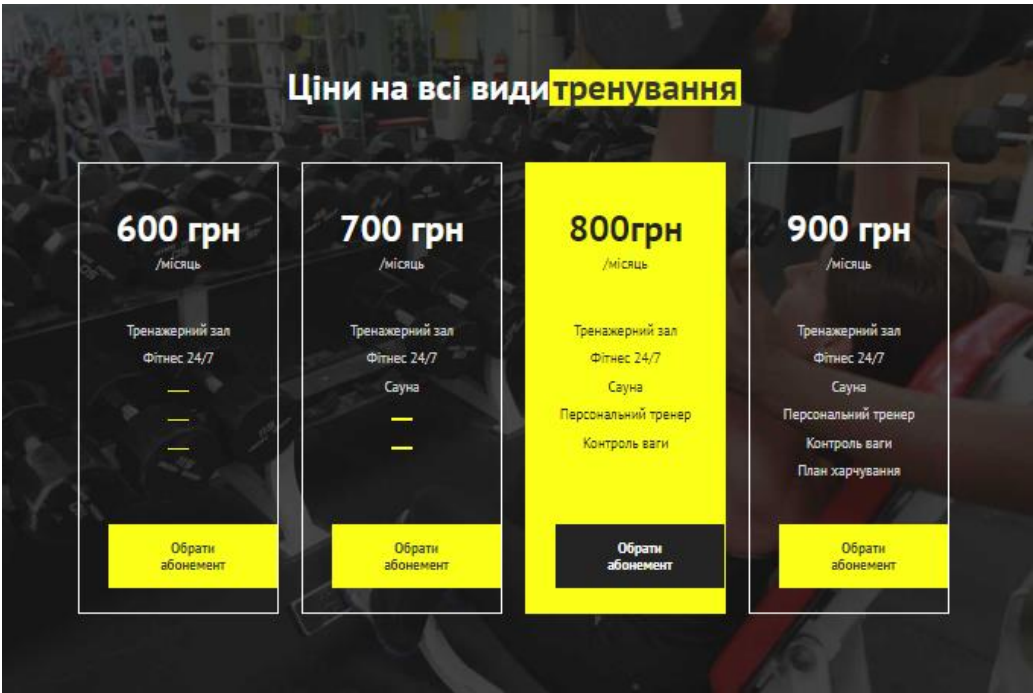


Рисунок 4.32 – Блок з цінами абонементів та категоріями послуг

Третій блок сторінки «Тренувань» це зв’язок з персональним тренером. За допомогою кнопки можна залишити свої контактні дані для того,щоб тренер за явним запитом зв’язався з вами (рис. 4.33).

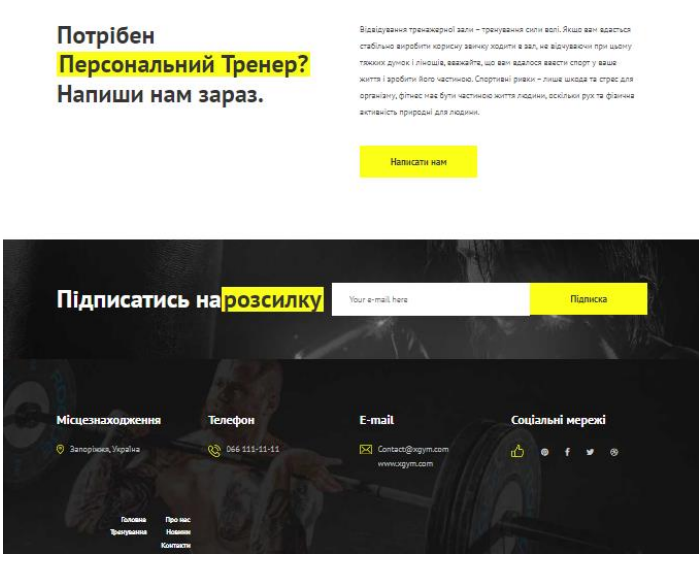


Рисунок 4.33 – Зв’язок з персональним тренером

Сторінка сайту «Новини» виглядає в стилі блогу тренажерного залу (рис. 4.34). Блоки цієї сторінки поділяються на нові публікації, категорії, архів публікацій, теги (рис. 4.35, рис. 4.36).

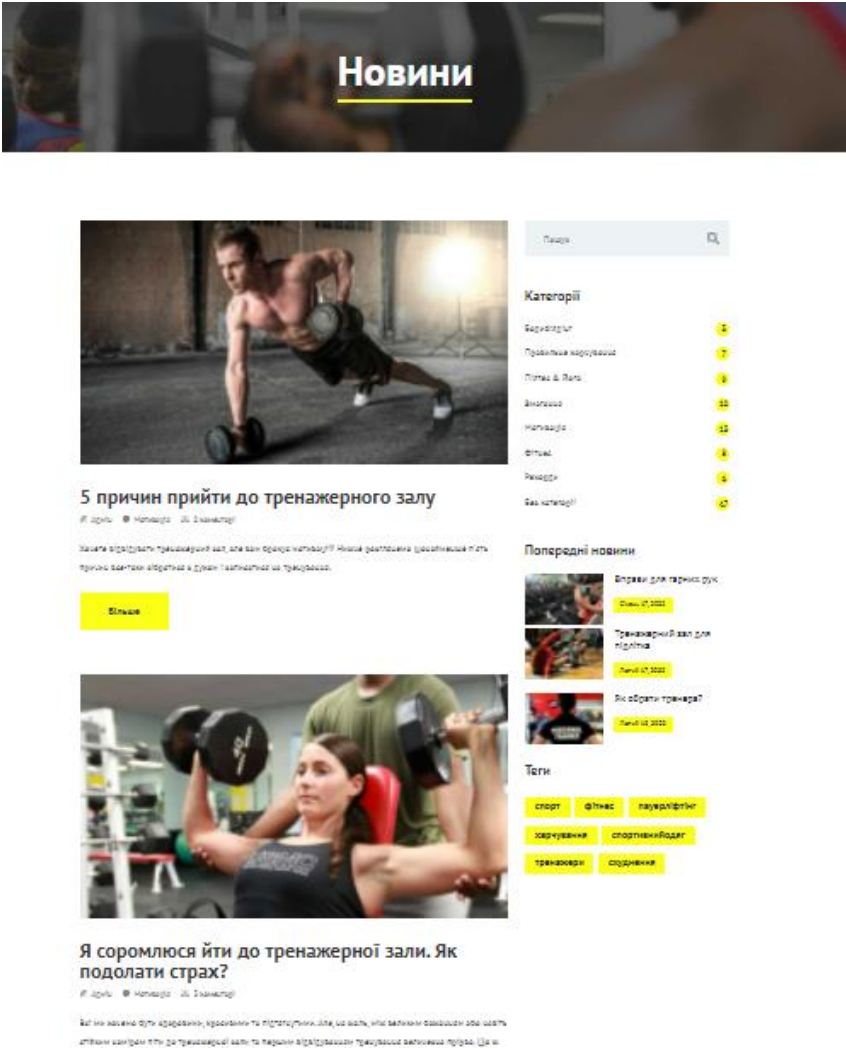


Рисунок 4.34 – Сторінка «Новини»

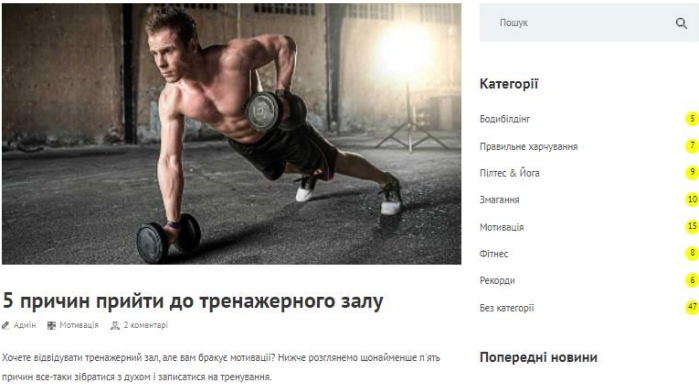


Рисунок 4.35 – Блок нових публікацій та категорій новин

Попередні новини



Теги



Рисунок 4.36 – Архів публікацій та теги новин

Контакти – це останній розділ головної сторінки сайту (рис. 4.37). На сторінці можливо, залишити своє питання безпосередньо через спеціальну форму. Також тут знаходиться точна адреса тренувального центру з геолокацією.

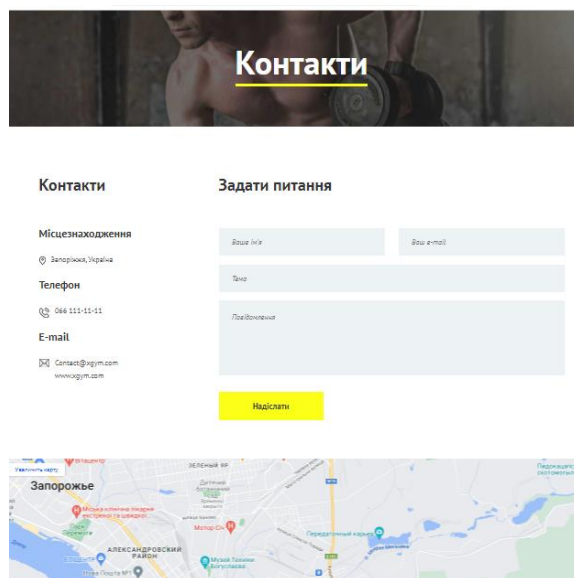


Рисунок 4.37 – Сторінка «Контакти»

4.3 Висновки за розділом

Результати розробки є продукт, що дозволяє підвищити якість взаємодії з клієнтами і підвищити імідж організації, а також надає можливість зменшити тимчасові витрати на роботу з потенційним клієнтом. Використання розробленого вебсайту для тренувального центру спрощує процеси просування компанії, ведення і видачі інформації, що призводить до значного зниження трудових витрат, забезпечує скорочення часу обробки інформації, підвищує достовірність інформації, що видається.

В процесі роботи над сайтом були розроблені наступні сторінки: «Головна», «Про нас», «Тренування», «Новини», «Контакти». Розроблені форми для зворотнього зв'язку з тренажерним клубом, зв'язку з персональним тренером, оформлення абонементи з різними видами послуг. В цілому сайт містить багато інтеративних функцій та змін в дизайні при маніпуляції мушкою за допомогою JavaScript. Має досить контрастні корпоративні кольори: чорний, жовтий, білий.

ВИСНОВКИ

В ході виконання дипломної кваліфікаційної роботи бакалавра було розроблено вебсистему тренажерного залу. Метою роботи було створити якісний спортивний сайт або фітнес–лендінг важливо для залучення нових клієнтів, зручності нинішніх користувачів.

В даній роботі було проведено аналіз сучасних технологій, методів та середовищ розробки вебсистем. Для розробки проєкту була обрана Aptana Studio 3. У ній можна настроїти графічний інтерфейс під себе. Є й такі функції: вбудований термінал, інтеграція з Git, відладчик, а також майстер розгортання. Автодоповнення в ній доступне для HTML, CSS та JavaScript.

Для розробки системи було обрано найбільш підходящі програмні засоби. Як СУБД обрано MySQL. Мови програмування: для верстки HTML, JavaScript, CSS, JavaScript, бібліотека jQuery, для серверної розробки PHP. Як серверну мову програмування було обрано PHP.

Вебсистема тренажерного залу складається з трьох основних компонентів: клієнтська частина вебпрограми, серверна частина вебсистеми, база даних. Сайт складається з наступних частин: головне меню навігації (шапки); кількох сторінок з інформацією та зображеннями. За допомогою головного меню навігації можна переміститися на певну сторінку сайту (головна сторінка, тренування, новини, про нас, контакти, оформлення абонементу). База даних вебсистеми тренажерного залу зберігає інформацію про абонементи, клієнтів, які оформили абонементи, тренерів, публікації блогу.

Дизайн головної сторінки відображає максимально корисну та актуальну інформацію про роботу тренажерного клубу. Можливість прочитати про види тренувань, мотиваційний девіз, щоб стати клієнтом клубу, інформацію про абонементи, презентацію тренерів, розрахунок IBM, хедер та футер сайту. Хедер – це меню з посиланнями на сторінки сайту і логотипом. Футер сайту складається з рядка із соціальними мережами клубу,

номером телефону та кнопкою «підписатися» на повідомлення від клубу. При натисканні на певний значок соціальних мереж користувача перенаправляє на офіційну сторінку клубу в обраній соціальній мережі. Сторінка «Про нас» відображає інформацію про персонал тренажерного залу та його напрям діяльності. Блоки, які до неї входять: текстовий контент, статистика, інформація про тренерів, форма зв'язку з тренером, хедер та футер сайту. Сторінка «Тренування». Один з основних розділів сайту з інформацією про послуги клубу. Сторінка сайту «Новини» виглядає в стилі блогу тренажерного залу. Блоки цієї сторінки поділяються на нові публікації, категорії, архів публікацій, теги публікацій. Контакти – це останній розділ головної сторінки сайту.

Сайт оптимізований до стандартних екранів гаджетів та буде виглядати комфортно на будь-якому пристрої користувача. Розроблені скрипти додали інтерактивні відображення інформації.

Таким чином, можна констатувати, що поставлена мета бакалаврської роботи досягнута.

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Дакетт Д. HTML и CSS. Разработка и дизайн веб-сайтов / Д. Дакетт. – М.: Эксмо, 2019. – 480 с.
2. Barrell D. Agile Accessibility Explained: A practical guide to sustainable accessible software development / D. Barrell. – Amazon Digital Services, 2019. – 340 p.
3. Byrne J. 60 hot to touch Accessible Web Design tips – the tips no web developer can live without! / Jim Byrne. – Boston, 2006. – 346 p.
4. Rachel A. The New CSS Layout / A. Rachel – A Book Apart, 2017. – 250 p.
5. Нильсен Я. Веб-дизайн / Я. Нильсен. – М.: Символ, 2015. – 512 с.
6. George Carole A. Usability testing and design of a library website: an iterative approach / A/ George Carole A // OCLC Systems & Services: International digital library perspectives. – 2005. - № 21(3). – pp. 167–180.
7. Алексеев А. Введение в Web-дизайн. Учебное пособие / А. Алексеев. – М.: ДМК Пресс, 2019. – 184 с.
8. Сырых Ю.А. Современный веб-дизайн. Настольный и мобильный / Ю.А. Сырых. – М.: Диалектика, 2019. – 384 с.
9. Кузнецов М., Симдянов И. PHP. Практика создания Web-сайтов / М. Кузнецов, И. Симдянов. – СПб.: БХВ-Петербург, 2012. – 347 с.
10. Клименко Р. Веб-мастеринг : изучаем HTML5, CSS3, JavaScript, PHP, CMS, AJAX, SEO / Р. Клименко. – СПб.: Питер, 2013. – 508 с.
11. Цеслів О.В. WEB-програмування : навч. посібник / О.В. Цеслів ; М-во освіти і науки, молоді та спорту України, Нац. техн. ун-т України “Київ. політехн. ін-т”. – К.: НТУУ “КПІ”, 2011. – 296 с.

12. Никсон Р. Создаем динамические веб-сайты с помощью PHP, MySQL и JavaScript = Learning PHP, MySQL and JavaScript / Р. Никсон ; [пер. с англ. Н. Вильчинского]. – М.; СПб.: Питер, 2013. – 496 с.
13. Пасічник О.Г. Основи веб-дизайну / О.Г. Пасічник, О.В. Пасічник, І.В. Стеценко. – К.; Вид.групаВНЦ, 2009. – 336 с.
14. Основи будування сайтів / В. Монако та інші. – К.: Шкільний світ, 2006. – 120 с.
15. Балик Н.Р., Мандзюк В.І. MySQL: лабораторний практикум / Н.Р. Балик, В.І. Мандзюк. – Тернопіль: Навчальна книга – Богдан, 2008. – 88 с.
16. Проценко О. Б. Web-програмування та web-дизайн. Технологія XML : навч. посіб. / О.Б. Проценко. – Суми : СумДУ, 2009. – 127 с.

ДОДАТОК А
Текст файлу main.js

```

'use strict';

$(window).on('load', function() {
    /*-----
        Preloder
    -----*/
    $(".loader").fadeOut();
    $("#preloder").delay(400).fadeOut("slow");

});

(function($) {
    /*-----
        Navigation
    -----*/
    $(".main-menu").slicknav({
        appendTo: '.header-section',
        allowParentLinks: true
    });

    /*-----
        Background Set
    -----*/
    $('.set-bg').each(function() {
        var bg = $(this).data('setbg');
        $(this).css('background-image', 'url(' + bg +
    '));

    });

    /*-----
        Index calculator
    -----*/
    function calcIndex () {

        var cm =
        Number(document.getElementById("IndexHeight").value);

        var kg =
        Number(document.getElementById("IndexKg").value);

        var result = kg / (cm * cm);

        var Index = Math.round(result * 100) / 100;
        if (Index < 18.5) {
            document.getElementById("IndexResult").value
= Index + " (Underkg)";
        } else if (Index >= 17 && Index <= 24) {
            document.getElementById("IndexResult").value
= Index + " (Normal)";
        } else if (Index >= 24 && Index <= 31) {
            document.getElementById("IndexResult").value
= Index + " (Big)";
        } else if (Index > 31) {

```

```

        document.getElementById("IndexResult").value
= Index + " (OverKg)";
    } else {
        document.getElementById("IndexResult").value
= Index;
    }
}

$('#Index-suIndext').on('click', function(){
    computeIndex();
});

/*-----
    Accordions
-----*/
$('.panel-link').on('click', function (e) {
    $('.panel-link').removeClass('active');
    var $this = $(this);
    if (!$this.hasClass('active')) {
        $this.addClass('active');
    }
    e.preventDefault();
});

/*-----
    Circle progress
-----*/
$('.circle-progress').each(function() {
    var cpvalue = $(this).data("cpvalue");
    var cpcolor = $(this).data("cpcolor");
    var cptitle = $(this).data("cptitle");
    var cpid = $(this).data("cpid");

    $(this).append('<div class="' + cpid + '"></div><div
class="progress-info"><h2>' + cpvalue + '%</h2><p>' + cptitle
+'</p></div>');

    if (cpvalue < 100) {

        $('.' + cpid).circleProgress({
            value: '0.' + cpvalue,
            size: 211,
            thickness: 5,
            fill: cpcolor,
            emptyFill: "rgba(0, 0, 0, 0)"
        });
    } else {
        $('.' + cpid).circleProgress({
            value: 1,
            size: 211,
            thickness: 5,

```

```
        fill: cpcolor,  
        emptyFill: "rgba(0, 0, 0, 0)"  
    });  
    }  
});  
})(jQuery);
```

ДОДАТОК Б
Текст файлу main.php

```

<?php include('header'); ?>
<?php          if(($_SESSION['login']=='1')          and
(($_SESSION['user_role']==2) or ($_SESSION['user_role']==3))) {
?>
<!DOCTYPE html>
<html lang="zxx">
<head>
    <title>Gym - Fitness </title>
    <meta charset="UTF-8">
    <meta name="description" content=" Gym Fitness ">
    <meta name="keywords" content="fitness, html">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">

    <!-- Favicon -->
    <link href="img/favicon.ico" rel="shortcut icon"/>

    <!-- Google font -->
    <link
href="https://fonts.googleapis.com/css?family=PT+Sans:400,400i,7
00,700i&display=swap" rel="stylesheet">

    <!-- Stylesheets -->
    <link rel="stylesheet" href="css/bootstrap.min.css"/>
    <link rel="stylesheet" href="css/font-
awesome.min.css"/>
    <link rel="stylesheet"
href="css/owl.carousel.min.css"/>
    <link rel="stylesheet" href="css/flaticon.css"/>
    <link rel="stylesheet" href="css/slicknav.min.css"/>

    <!-- Main Stylesheets -->
    <link rel="stylesheet" href="css/style.css"/>

</head>
<body>
    <!-- Page Preloder -->
    <div id="preloder">
        <div class="loader"></div>
    </div>

    <!-- Header section -->
    <header class="header-section">
        <a href="home.html" class="site-logo">
            
        </a>
        <ul class="main-menu">
            <li><a href="index.html">Головна</a></li>
            <li><a href="about.html">Про нас</a></li>

```

```

        <li><a
href="classes.html">Тренування</a></li>
        <li><a href="blog.html">Новини</a></li>
        <li><a href="contact.html">Контакти</a></li>
        <li class="header-right">
            <div class="hr-box">
                
                <h6>Запоріжжя, Україна</h6>
            </div>
        </li>
    </ul>
</header>
<div class="clearfix"></div>
<!-- Header section end -->

<!-- Hero section -->
<section      class="hero-section      set-bg"      data-
setbg="img/hero-bg.jpg">
    <div class="container">
        <div class="hero-text">
            <h2>Фізична активність</h2>
            <h2>завжди <span>вперед</span></h2>
            <a href="" class="site-btn">Долучайся до
занять</a>
        </div>
    </div>
</section>
<!-- Hero section end -->

<!-- Feature section -->
<section class="feature-section">
    <div class="container">
        <div class="row justify-content-md-center">
            <div class="col-lg-8">
                <div      class="section-title      text-
center">
                    <h2>Справжня
<span>робота</span> починається коли ти хочеш зупинитись.</h2>
                </div>
            </div>
        </div>
        <div class="row">
            <div class="col-md-4">
                <div class="icon-box-item">
                    <div class="ib-icon">
                        <i      class="flaticon-004-
dumbbell"></i>
                    </div>
                    <h4>Регулярні тренування</h4>
                    <p>Покращують фізичну форму
</p>
                </div>
            </div>
        </div>
    </div>

```

```

        </div>
        <div class="col-md-4">
            <div class="icon-box-item">
                <div class="ib-icon">
                    <i class="flaticon-031-
app-1"></i>

                    </div>
                    <h4>Регулярні тренування</h4>
                    <p>Покращують внутрішнє
здоров'я. </p>

                </div>
            </div>
            <div class="col-md-4">
                <div class="icon-box-item">
                    <div class="ib-icon">
                        <i class="flaticon-010-
podium"></i>

                        </div>
                        <h4>Регулярні тренування</h4>
                        <p>Покращують самооцінку.
</p>

                    </div>
                </div>
            </div>
        </div>
    </section>
    <!-- Feature section end -->

    <!-- Add section -->
    <section class="add-section set-bg" data-
setbg="img/add-bg.jpg">
        <div class="container">
            <div class="row">
                <div class="col-lg-7 ml-auto">
                    <div class="add-text">
                        <h2>Оплати тренування зараз та
отримуй 25% <span>скидку</span></h2>
                        <ul>
                            <li>Регулярні заняття.</li>
                            <li>Кращий час відвідування в залі</li>
                            <li>Персональний тренер.</li>
                        </ul>
                        <a href="#" class="site-
btn">Придбати абонемент</a>

                    </div>
                </div>
            </div>
        </div>
    </section>
    <!-- Add section end -->

```

```

<!-- Milestones section -->
<section class="milestones-section">
  <div class="container">
    <div class="row">
      <div class="col-lg-3 col-sm-6">
        <div class="milestone">
          <h2>1374</h2>
          <p>Щасливих клієнтів</p>
        </div>
      </div>
      <div class="col-lg-3 col-sm-6">
        <div class="milestone">
          <h2>356</h2>
          <p>Схуднувших людей</p>
        </div>
      </div>
      <div class="col-lg-3 col-sm-6">
        <div class="milestone">
          <h2>57k</h2>
          <p>Підписників</p>
        </div>
      </div>
      <div class="col-lg-3 col-sm-6">
        <div class="milestone">
          <h2>24/7</h2>
          <p>Працюючих годин</p>
        </div>
      </div>
    </div>
  </div>
</section>
<!-- Milestones section end -->

  <!-- Trainers section -->
  <section class="trainers-section">
    <div class="container">
      <div class="section-title text-center">
        <h2>Познайомитись
        <span>Тренером</span></h2>
      </div>
      <div class="row">
        <div class="col-lg-3 col-sm-6">
          <div class="trainer-item">
            <div class="trainer-pic">
              
            </div>
            <h4>Михайло Прозь</h4>
            <p>Тренер (персональний
тренер) має ширші знання, ніж фітнес-інструктор, повинен вміти
побудувати індивідуальну програму відповідно до фізіологічних
особливостей клієнта та бажаного результату (у тому числі для

```

особливих груп клієнтів із захворюваннями серцево-судинної системи, опорно-рухового апарату, вагітних жінок і т.д.). У нього можна отримати компетентну консультацію щодо правильного харчування.</p>

```
</div>
</div>
<div class="col-lg-3 col-sm-6">
  <div class="trainer-item">
    <div class="trainer-pic">
      
    </div>
    <h4>Юлія Квітка</h4>
    <p>Фітнес-інструктор повинен
добре розумітися на таких питаннях, як будова людського тіла,
знати, як працюють системи та органи при фізичних навантаженнях,
знати основні види тренувань та особливості впливу тих чи інших
вправ на організм.</p>
```

```
</div>
</div>
<div class="col-lg-3 col-sm-6">
  <div class="trainer-item">
    <div class="trainer-pic">
      
    </div>
    <h4>Кирило Донець</h4>
    <p>Тренер (персональний
тренер) має ширші знання, ніж фітнес-інструктор, повинен вміти
побудувати індивідуальну програму відповідно до фізіологічних
особливостей клієнта та бажаного результату (у тому числі для
особливих груп клієнтів із захворюваннями серцево-судинної
системи, опорно-рухового апарату, вагітних жінок і т.д.). У
нього можна отримати компетентну консультацію щодо правильного
харчування.</p>
```

```
</div>
</div>
<div class="col-lg-3 col-sm-6">
  <div class="trainer-item">
    <div class="trainer-pic">
      
    </div>
    <h4>Марія Мижайлюк</h4>
    <p>Майстер-тренер - це тренер,
який повинен знати про спорт все, а також знатися на основах
психології, педагогіки, вміти продати свій продукт, тобто.
«персональне тренування», навчити персонал нижчої категорії.
Саме вони готують клієнтів до змагань та проводять спеціальні
тренування. Обов'язковою є наявність вищої спортивної освіти та
підтвердження категорії з отриманням відповідного
сертифікату.</p>
```

```
</div>
```

```

        </div>
    </div>
</div>
</section>
<!-- Trainers section end -->

<!-- BMI section -->
<section class="bmi-section spad">
    <div class="bmi-bg set-bg" data-setbg="img/bmi-
bg.jpg"></div>
    <div class="container">
        <div class="row">
            <div class="col-lg-7 ml-auto">
                <div class="section-title mb-0">
                    <h2>Розрахунок          вашого
<span>BMI</span></h2>
                    <p>Індекс маси тіла (BMI) –
величина, що дозволяє оцінити ступінь відповідності маси людини
та її зростання і тим самим побічно судити про те, чи є маса
недостатньою, нормальною чи надмірною.</p>
                    </div>
                    <div class="bmi-calculator-warp">
                        <div class="bmi-calculator">
                            <div class="row">
                                <div class="col-sm-
6">
                                    <input
type="text" placeholder="Маса (кг)" id="bmi-weight">
                                </div>
                                <div class="col-sm-
6">
                                    <input
type="text" placeholder="Зріст (м)" id="bmi-hight">
                                </div>
                                <div class="col-sm-
6">
                                    <button
class="site-btn" id="bmi-submit">Розрахунок</button>
                                </div>
                                <div class="col-sm-
6">
                                    <input
type="text" id="bmi-result" readonly>
                                </div>
                            </div>
                        </div>
                    <p>Індекс маси тіла слід
застосовувати з обережністю виключно для орієнтовної оцінки.</p>
                    </div>
                </div>
            </div>
        </div>
    </div>
</section>

```

```

<!-- BMI section end -->

<!-- Newsletter section -->
<section class="newsletter-section set-bg" data-
setbg="img/newsletter-bg.jpg">
  <div class="container">
    <div class="row">
      <div class="col-lg-6">
        <div class="section-title text-
white mb-0">
          <h2>Підписатись
на<span>розсилку</span></h2>
        </div>
      </div>
      <div class="col-lg-6">
        <form class="newsletter-form">
          <input
placeholder="Your e-mail here">
            type="text"
          <button class="site-btn"
type="submit">Підписка</button>
        </form>
      </div>
    </div>
  </div>
</section>
<!-- Newsletter section end -->

<!-- Footer section -->
<footer class="footer-section set-bg" data-
setbg="img/footer-bg.jpg">
  <div class="container">
    <div class="row">
      <div class="col-lg-3 col-sm-6">
        <div class="footer-widget">
          <h4>Місцезнаходження</h4>
          <div class="fw-info-box">
            
            <div class="fw-info-
text">
              <p>Запоріжжя,
Україна</p>
            </div>
          </div>
        </div>
      </div>
      <div class="col-lg-3 col-sm-6">
        <div class="footer-widget">
          <h4>Телефон</h4>
          <div class="fw-info-box">
            

```

```

text">

```

```

нає</a></li>
href="">Тренування</a></li>
href="">Контакти</a></li>
</ul>
</div>
</div>
</div>
</footer>
<!-- Footer section end -->

<!--===== Javascripts & JQuery =====-->
<script src="js/jquery-3.2.1.min.js"></script>
<script src="js/bootstrap.min.js"></script>
<script src="js/jquery.slicknav.js"></script>
<script src="js/owl.carousel.min.js"></script>
<script src="js/circle-progress.min.js"></script>
<script src="js/main.js"></script>

</body>
</html>

```

ДОДАТОК В
Текст файлу form.php

```
<?php include('header'); ?>
<?php          if(($_SESSION['login']=='1')          and
(($_SESSION['user_role']==2) or ($_SESSION['user_role']==3))) {
?>

<div class='body'>
  <center><table border='0' width='1200'>
    <tr>
      <td height='100%' valign='top'>
        <div class="body">
          <div class="body_resize1">
            <div class="side_left1">
              <div
class="left1_1">
              <div
class="left_top1">
              <div
class="left_bottom1">

      <center><h2>Бронювання абонементу</h2></center>

<?php

$aa=1;

if($_GET['work']=='edit') {

$text="update page_text set clientName='".$_POST[clientName]','
clientSurname='".$_POST[clientSurname]','
  subStartDate='".$_POST[subStartDate]','
subStartEnd='".$_POST[subStartEnd]','
clientTelephone='".$_POST[clientTelephone]','
clientEmail='".$_POST[clientEmail]'"

  where page_text_id=$aa";

$query=mysql_query($text);

if($_FILES['text_page_image']['name']!='') {

  $good_photo="text_page_image";

  $folder="../temp";

  $image=upload_image($good_photo,$folder);

  $size = getimagesize($image);

  if($size[0]>300) {
```

```

        img_resize("$image", "$image", 300);

    }

    rename("$image", "../images_page/page_$aa.jpg");

    $good_image="images_page/page_$aa.jpg";

    $text="update                                page_text                                set
page_text_image='$good_image' where page_text_id=$aa";

    $query=mysql_query($text);

}

echo("<p align='center'><b>Готово</b></p>");

}

$text="select * from page_text where page_text_id=$aa";

$query=mysql_query($text);

while($temp=mysql_fetch_array($query)) {

    echo("<form                                action='$_SERVER[PHP_SELF]?work=edit'
method='post' enctype='multipart/form-data'>");

        echo("<br><center><table                                cellpadding='0'
cellspacing='0'>");

            echo("<tr>");

                echo("<td><p align='left'>І'мя.</p></td>");

                echo("<td><input type='text' class='max'
name='clientName' value='$temp[clientName]' style='width:
250px;'></td>");

            echo("</tr>");

            echo("<tr>");

                echo("<td                                valign='top'><p
align='left'>Прізвище</p></td>");

                echo("<td><textarea                                class='max'
name='clientSurname' style='width: 250px; height:
50px;'>$temp[clientSurname]</textarea></td>");

            echo("</tr>");

```

```

        echo("<tr><td          colspan='2'          style='height:
10px;'></td></tr>");

        echo("<tr><td          colspan='2'          style='height:
10px;'></td></tr>");

        echo("<tr>");

        echo("<td          valign='top'><p
align='left'>Адреса.</p></td>");

        echo("<td><textarea          class='max'
name='subStartDate'          style='width:          250px;          height:
50px;'>$temp[subStartDate]</textarea></td>");

        echo("</tr>");

        echo("<tr><td          colspan='2'          style='height:
10px;'></td></tr>");

        echo("<tr>");

        echo("<td><p
align='left'>Абонемент</p></td>");

        echo("<td><input          type='text'          class='max'
name='subStartEnd'          value='$temp[subStartEnd]'          style='width:
250px;'></td>");

        echo("</tr>");

        echo("<tr>");

        echo("<td colspan='2'>");

        echo("<textarea
id='clientTelephone' name='clientTelephone' style='width: 600px;
height: 600px;'>$temp[clientTelephone]</textarea>");
        ?>

        <script type="text/javascript">

                var          ckeditor1          =
CKEDITOR.replace( 'clientTelephone' );

                AjaxFileManager.init({

```

```

returnTo: 'ckeditor',
editor: ckeditor1

});

</script>
<?php
    echo("</td>");

    echo("</tr>");

    echo("<tr><td        colspan='2'        style='height:
10px;'></td></tr>");

    echo("<tr>");

    echo("<td colspan='2'>");

                                echo("<textarea        id='clientEmail'
name='clientEmail'        style='width:        600px;        height:
600px;'>$temp[clientEmail]</textarea>");
    ?>

    <script type="text/javascript">

        var        ckeditor1        =
CKEDITOR.replace( 'clientEmail' );

        AjaxFileManager.init({

            returnTo: 'ckeditor',

            editor: ckeditor1

        });

    </script>
<?php

    echo("</td>");

    echo("</tr>");

    echo("</table></center>");

    echo("<br><center><input                                type='image'
src='images/good_edit.jpg'></center><br>");

```

```

        echo("</form>");
    }

                                                                    ?>
                                                                    <div
class="clr"></div>
                                                                    </div>
                                                                    </div>
                                                                    </div>
                                                                    </div>
                                                                    <div class="clr"></div>
                                                                    </div>
                                                                    <div class="clr"></div>
                                                                    </td>
                                                                    <td      height='100%'      width='400'
valign='top'>
                                                                    <?php include("clone_menu"); ?>
                                                                    </td>
                                                                    </tr>
                                                                    </table></center>
</div>
<?php } else {?>

        <div class='body' style='width: 100%;'><center>
            <br><br><br><img      src='images/Stop.png'
style='padding-left: 400px;'><br><br>
            </center></div>
        <?php }?>
        <?php include('footer'); ?>

```

ДОДАТОК Г
Слайди презентації

Національний університет «Запорізька політехніка»
Кафедра програмних засобів

РОЗРОБКА ВЕБСИСТЕМИ ТРЕНАЖЕРНОГО ЗАЛУ

Виконав: студент гр. КНТ-138, Самойленко Є.В.

Керівник: доцент Скрупський Степан Юрійович



Рисунок Г.1 – Слайд 1

Мета роботи

Актуальність розробки вебсистеми тренажерного залу полягає в інформаційному забезпеченні фірми в реальному часі. Вебсистема дозволяє надати інформацію про компанію та її товари або послуги стисло і водночас повноцінно. Також сайт повідомляє про новини фірми, про зміни в прайсі або режим роботи, містить мотиваційну інформацію для майбутніх клієнтів.



2/14

Рисунок Г.2 – Слайд 2

Технології розробки

Фронтенд	Бекенд
HTML - її елементи дозволяють відображати стандартні блоки сторінок, а також являють собою форматований текст, зображення, таблиці, форми введення даних і т.д.	PHP - переважно застосовується, щоб зробити інтерактивними статичні HTML сторінки
CSS - за допомогою CSS задаються шрифти, кольори, стилі, розташування окремих елементів та відображення сторінок на різних пристроях.	JavaEE - кросплатформність, мультифункціональність, надійність та гнучкість. Фреймворки: -Spring; -Hibernate.
JavaScript - використовується для розробки інтерактивних об'єктів вебсторінок, які покращують взаємодію користувача. Популярні фреймворки: -React; -Angular; -Vue.	Node.js - кросплатформове середовище, яке виконує код JavaScript поза браузером. MySQL - реляційна БД зберігає структуровані дані, які зазвичай становлять об'єкти реального світу.

3/14

Рисунок Г.3– Слайд 3

Аналіз IDE

IDE	Переваги
Visual Studio	Підтримує мови ASP.NET, Python, Node.js, C++. Можна створювати середовища розробки прямо в хмарі.
IntelliJ IDEA	Основна увага тут приділяється функціям, що підвищують продуктивність (індексація коду, автодоповнення, рефакторинг).
Aptana Studio 3	В ній можна настроїти графічний інтерфейс під себе: вбудований термінал, інтеграція з Git, відладчик, а також майстер розгортання. Автодоповнення доступне для HTML, CSS та JS.
PyCharm	Середовище для розробки на Python. PyCharm перевіряє код на наявність помилок прямо під час написання програми та має функцію автодоповнення.
PhpStorm	Розроблена під PHP, але повністю підтримує інші фронтенд-мови: HTML 5, CSS, JavaScript та інші.
WebStorm	Вона включає функцію автодоповнення, виявляє помилки прямо під час написання коду і має вбудований налагодчик. Також підтримує модульне тестування та інтеграцію з такими сервісами: Git, SVN
NetBeans	Допомагає оптимізувати процес розробки за допомогою синтаксичного та семантичного підсвічування коду. Вона має безліч потужних інструментів для рефакторингу і при цьому є open source проектом
Komodo IDE	Вона має функції: автозаповнення, візуальний відладчик, підтримку модульного тестування, доповнення, спільну розробку з командою, інструментарій для управління проектами та багато іншого. Komodo IDE підтримує роботу з Git, SVN та може працювати з усіма мовами програмування.

4/14

Рисунок Г.4 – Слайд 4

Структура вебсистеми



Рисунок Г.5 – Слайд 5

Структура вебсайту

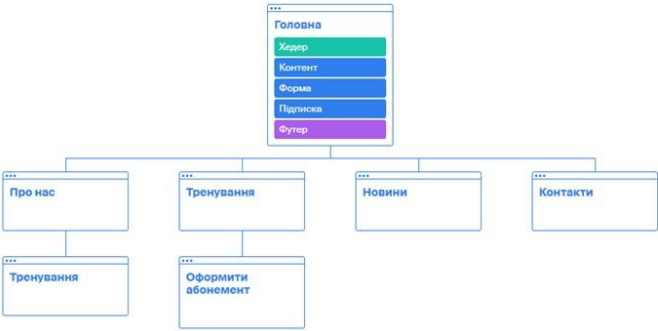


Рисунок Г.6 – Слайд 6

База даних

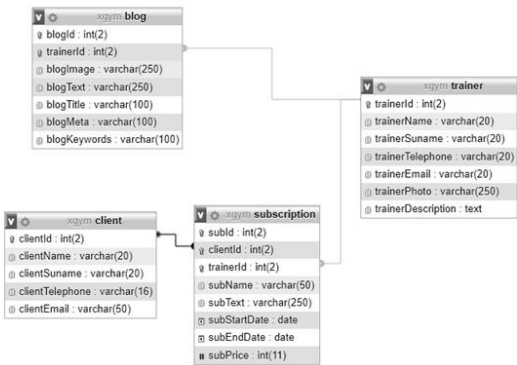


Рисунок Г.7 – Слайд 7

Макети сторінок

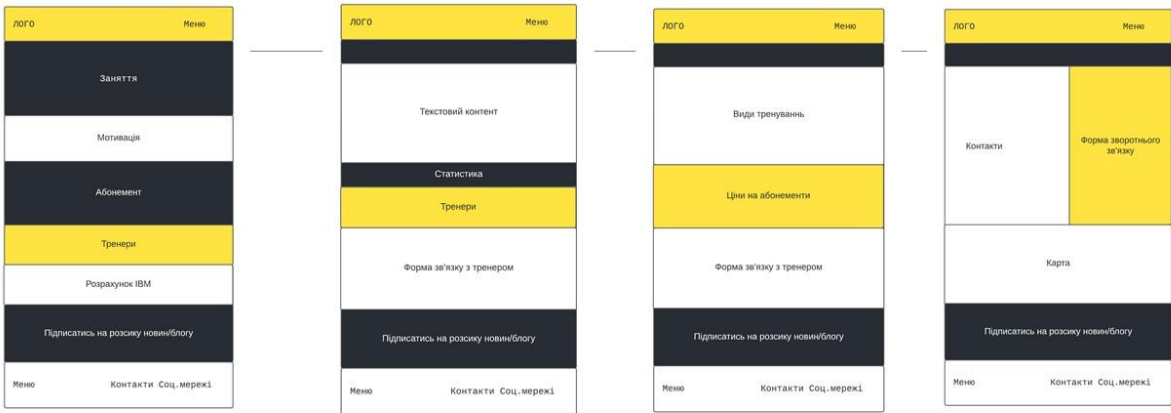


Рисунок 1 – Дизайн сторінки «Головна»

Рисунок 2 – Дизайн сторінки «Про нас»

Рисунок 3 – Дизайн сторінки «Тренування»

Рисунок 4 – Дизайн сторінки «К»

Рисунок Г.8 – Слайд 8

Макети сторінок

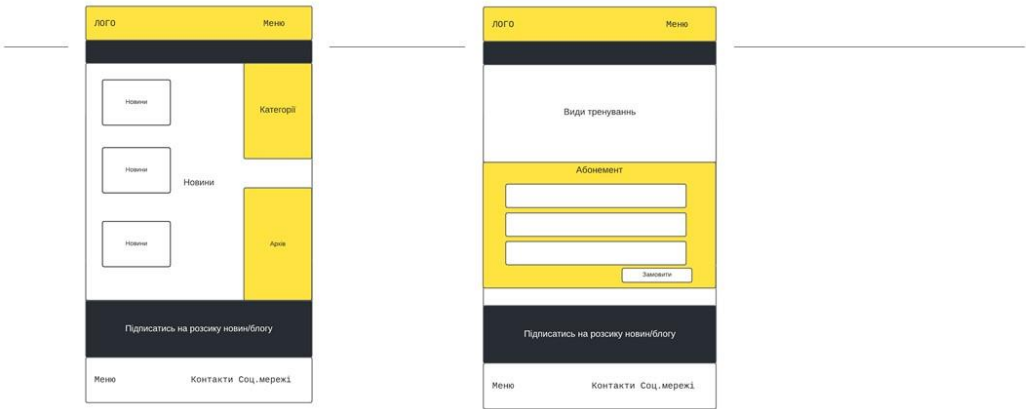


Рисунок 5 – Сторінка «Новини»

Рисунок 6 – Форма бронювання абонементу

Рисунок Г.9 – Слайд 9

Результати роботи: зовнішній вигляд хедера та футера



Рисунок 7 – Меню сайту тренажерного залу

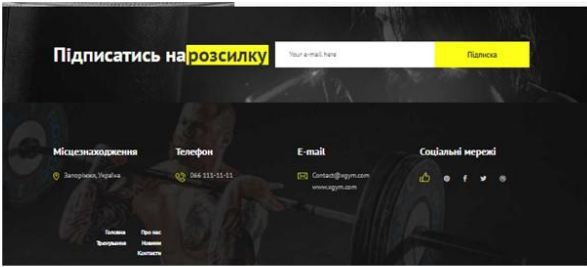


Рисунок 8 – Футер сайту тренажерного залу

Рисунок Г.10 – Слайд 10

Результати роботи: зовнішній вигляд сторінок

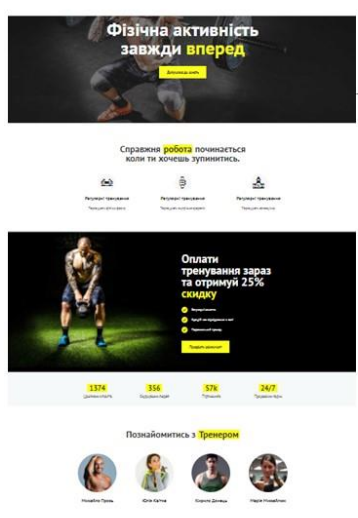


Рисунок 9 – Головна сторінка сайту тренажерного залу

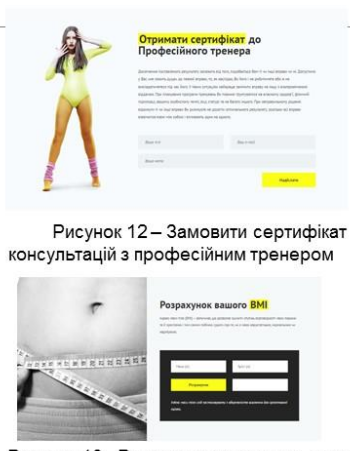


Рисунок 12 – Замовити сертифікат консультацій з професійним тренером



Рисунок 11 – Сторінка «Про нас»

Рисунок 10 - Розрахунок показника маси тіла

Рисунок Г.11 – Слайд 11

Результати роботи: інтерактивні об'єкти

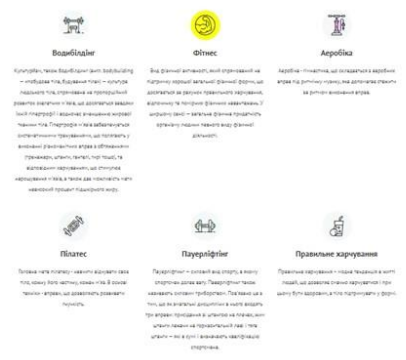


Рисунок 13 – Види тренувань



Рисунок 14 - Блок з цінами абонементів та категоріями послуг

Рисунок Г.12 – Слайд 12

Результати роботи: зовнішній вигляд сторінок

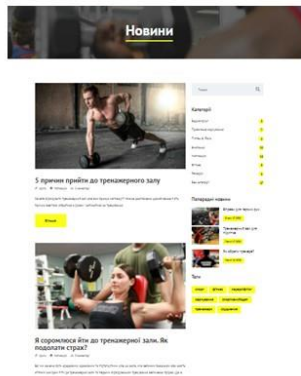


Рисунок 15 – Сторінка «Новини»

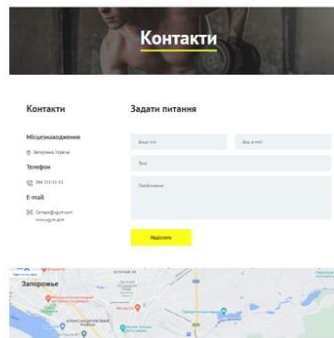


Рисунок 16 – Сторінка «Контакти»

13/14

Рисунок Г.13 – Слайд 13

Висновок

В результаті було розроблено вебсистему тренажерного залу та вирішено наступні задачі:

- ☐ проаналізовані методи та середовища розробки сучасні вебсистеми;
- ☐ спроектована база даних;
- ☐ розроблені структури вебсторінок;
- ☐ розроблено вебсайт.

14/14

Рисунок Г.14 – Слайд 14