

Національний дієотехнічний університет України

(повна назва університету, адреса, код ЄДРПОУ)

Навчально-науковий інститут комп'ютерних наук та
інформаційних технологій

(повна назва інституту, адреса факультету (навчального))

Кафедра комп'ютерних наук

(повна назва кафедри (предметної, спеціальної тощо))

Пояснювальна записка

до дипломної роботи
перший (бакалаврський)

(рівень вищої освіти)

на тему: Розроблення вебсистеми діяльності спортивного клубу комплексу
LEOLAND засобами PHP

Виконав: студент 2 курсу, групи КНС-21
спеціальності

122 – “Комп'ютерні науки”

(номер та назва напрямку підготовки, спеціальності)

Ковальчук І.І.

(прізвище та ініціали)

Керівник Карашецький В.П.

(прізвище та ініціали)

Рецензент Прокіш Ю.С.

(прізвище та ініціали)

Львів – 2025 року

Національний лісотехнічний університет України

(повна підпорядкована вищого навчального закладу)

НЦІ комп'ютерних наук та інформаційних технологій

Кафедра комп'ютерних наук

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 – "Комп'ютерні науки"

(номер і назва)

ЗАТВЕРДЖУЮ

Завідувач кафедри КН

Борещак І.Б.

"10" серпня 2025 року

**ЗАВДАННЯ
НА ДИПЛОМНУ РОБОТУ СТУДЕНТУ**

Ковальчук Ігор Іванович

(прізвище, ім'я, по батькові)

1. Тема роботи: Розроблення вебсистеми діяльності спортивного клубу
комплексу LEO LAND засобами PHP

керівник роботи Карашецький Володимир Петрович, к.т.н., доцент,

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

затверджені наказом вищого навчального закладу від "15" 11 2024р. № C-884

2. Термін подання студентом роботи 10.06.2025р.

3. Вихідні дані до роботи Розробити вебсистему діяльності спортивного клубу
комплексу LEO LAND засобами PHP.

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

Стан проблемної області

Програмне та математичне забезпечення

Програмне та технічне забезпечення

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

6. Дата видачі завдання 18 листопада 2024р.

АНОТАЦІЯ

Дипломна робота містить 61 сторінку пояснювальної записки, 31 малюнок, 1 таблицю, 20 джерел.

У роботі розроблено вебсистему діяльності спортивного клубу LEOLAND, що включає функціонал запису на тренування, управління розкладом, а також адміністрування клієнтів та тренерів. Для реалізації було використано технології PHP, HTML, CSS, JavaScript, XAMPP та СУБД MySQL.

Розроблена система надає зручний інтерфейс для клієнтів клубу та адміністрації, забезпечуючи ефективну комунікацію і автоматизацію процесів. Основна мета – підвищення якості обслуговування та оптимізація роботи спортивного клубу через IT-рішення.

Ключові слова: вебсистема, спортивний клуб, PHP, HTML, CSS, JavaScript, MySQL, XAMPP, LEOLAND.

ABSTRACT

The graduate work includes 61 pages of explanatory notes, 31 figures, 1 tables, and is based on 20 sources.

This thesis presents the development of a web system for the LEOLAND sports club, including training registration, schedule management, and administration of clients and trainers. The system is implemented using PHP, HTML, CSS, JavaScript, XAMPP, and MySQL database.

The developed platform offers a convenient interface for clients and staff, aiming to enhance service quality and streamline club operations through digital solutions.

Keywords: web system, sports club, PHP, HTML, CSS, JavaScript, MySQL, XAMPP, LEOLAND.

ТЕХНІЧНЕ ЗАВДАННЯ

Розробити комплексну, надійну та зручну у використанні вебсистему для автоматизації управління діяльністю спортивного клубу комплексу LEOLAND, що дозволить оптимізувати процеси обліку клієнтів, управління абонементами, розкладом, гуртками та заявками, а також покращити взаємодію з клієнтами.

Для розробки архітектури вебсистеми обрати клієнт-серверну архітектуру для забезпечення чіткого розподілу відповідальності між рівнями представлення, бізнес-логіки та даних, що підвищує масштабованість, безпеку та гнучкість у подальшій розробці та підтримці системи. Ефективний обмін даними між клієнтською частиною, сервером та базою даних (БД) реалізувати наступними засобами:

- Серверна частина (Backend) – PHP.
- Клієнтська частина (Frontend) – HTML, CSS, JavaScript.
- База даних – MySQL.
- Середовище розробки – XAMPP (локальний вебсервер), VS Code (інтегроване середовище розробки).

Система повинна надавати наступні функціональні можливості:

- Для клієнтів:
 - Реєстрація та авторизація – Створення нового облікового запису та входу в систему, а також функція відновлення пароля.
 - Перегляд інформації про гуртки – Доступ до повного переліку гуртків спортивного клубу з детальною інформацією (графік роботи, ціна, тренер, кількість доступних місць).
 - Вибір та запис до гуртка – Вибір конкретного гуртка, дня та години для запису, з подальшим оформленням заявки.
 - Перегляд власних записів – Доступ до особистого кабінету для перегляду історії бронювань та їхнього статусу.
- Для адміністраторів:
 - Управління гуртками – Повний контроль над інформацією про гуртки, включаючи додавання, редагування та видалення гуртка.

- Управління заявками на запис – Перегляд усіх заявок від клієнтів з можливістю їх схвалення або відхилення.

- Управління користувачами – Перегляд та редагування даних зареєстрованих клієнтів.

По завершенню розробки та тестування, система має бути повністю функціональною, забезпечувати автоматизацію ключових бізнес-процесів спортивного клубу, покращувати взаємодію з клієнтами, надавати керівництву ефективні інструменти для управління та аналізу.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ	9
ВСТУП.....	10
РОЗДІЛ 1. СТАН ПРОБЛЕМНОЇ ОБЛАСТІ.....	12
1.1. Огляд діяльності спортивного клубу комплексу LEOLAND	12
1.2. Аналіз існуючих вебсистем для спортивних клубів.....	15
РОЗДІЛ 2. ІНФОРМАЦІЙНЕ ТА МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ	18
2.1. Обґрунтування вибору технологій розробки.....	18
2.2. Вибір потрібного інформаційного забезпечення.....	18
2.3. Побудова дерева проблем та дерева цілей.....	23
2.4. Аналіз і вибір методів, алгоритмів та засобів розв'язання задачі	24
РОЗДІЛ 3. ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ.....	25
3.1. Структура даних та схема бази даних.	25
3.2. Розробка інтерфейсу користувача та опис роботи вебсистеми	27
3.3 Вимоги до апаратного та програмного забезпечення	40
3.4 Розгортання системи	40
ВИСНОВКИ	42
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	43
ДОДАТОК	45

ПЕРЕЛІК СКОРОЧЕНЬ ТА УМОВНИХ ПОЗНАЧЕНЬ

БД – база даних;

ІС – інформаційна система;

ООП – об'єктно-орієнтоване програмування;

ПЗ – Програмне забезпечення;

СКБД – система керування базами даних;

HTML – Hyper Text Markup Language;

MySQL – система керування реляційними базами даних;

JS – JavaScript;

CSS – Cascading Style Sheets;

Visual Studio Code – інтегроване середовище розробки;

Web – розподілена інформаційна мультимедійна система, заснована на гіпертексті.

ВСТУП

У сучасному світі ефективне управління та автоматизація бізнес-процесів є ключовими для успіху будь-якого підприємства, включаючи спортивні клуби. Метою цього проєкту є створення сучасної вебсистеми, яка підвищить ефективність діяльності спортивного клубу комплексу LEOLAND та якість обслуговування клієнтів.

Шляхом розробки інноваційної вебсистеми за допомогою PHP оптимізуються внутрішні процеси спортивного клубу комплексу LEOLAND, включаючи управління клієнтами, абонементами, розкладом тренувань та персоналом, а також покращується взаємодія з відвідувачами. Застосування PHP як основи для розробки покликане забезпечити надійне та гнучке рішення для сучасних потреб спортивного клубу.

Актуальність теми обумовлена зростаючою потребою спортивних клубів у комплексних вебсистемах для оптимізації бізнес-процесів, покращення взаємодії з клієнтами та підвищення ефективності. Розробка такої системи для спортивного клубу комплексу LEOLAND дозволить автоматизувати рутинні операції, забезпечити онлайн-доступ до інформації (бронювання, розклад), покращити аналітику та підвищити конкурентоспроможність. Використання PHP є обґрунтованим через його поширеність, гнучкість та наявність розвиненої спільноти.

Об'єктом дослідження – вебсистеми управління діяльністю спортивних закладів.

Предметом дослідження – вебсистема діяльності спортивного клубу комплексу LEOLAND засобами PHP.

Мета роботи – розроблення функціональної вебсистеми спортивного клубу комплексу LEOLAND, яка дозволить ефективно керувати клієнтами, абонементами, розкладом тренувань, а також автоматизувати інші ключові аспекти діяльності, забезпечуючи зручність як для адміністрації, так і для відвідувачів.

Для досягнення поставленої мети були сформульовані наступні завдання:

1. Проаналізувати предметну область, виявити бізнес-процеси та вимоги спортивного клубу комплексу LEOLAND.
2. Дослідити існуючі аналоги та обґрунтувати вибір технологій (PHP, СКБД, HTML, CSS, JS).
3. Розробити архітектуру системи, включаючи БД та функціональні модулі.
4. Створити ергономічний дизайн інтерфейсу користувача.
5. Реалізувати вебсистему, забезпечивши її функціональність та надійність.
6. Провести тестування системи.

Практичне значення – впровадження розробленої вебсистеми спортивного клубу комплексу LEOLAND, яка:

- Автоматизує облік клієнтів, абонементів, фінансових операцій.
- Оптимізує управління розкладом та завантаженістю залів.
- Покращує обслуговування клієнтів через зручний онлайн-доступ та бронювання.
- Зменшує операційні витрати та підвищує ефективність персоналу.
- Надає керівництву інструменти для аналізу та прийняття обґрунтованих рішень.

РОЗДІЛ 1. СТАН ПРОБЛЕМНОЇ ОБЛАСТІ

1.1 Огляд діяльності спортивного клубу комплексу LEOLAND

Сучасний ринок спортивних послуг в Україні, зокрема у місті Львові, є надзвичайно динамічним та конкурентним. Спортивні клуби, які пропонують широкий і диверсифікований спектр послуг (тренажерні зали, групові фітнес-програми, бойові мистецтва плавальні басейни, SPA-зони, дитячі секції та послуги персональних тренерів, функціонують в умовах постійно зростаючих очікувань клієнтів. Для ефективного управління таким розмаїттям послуг та забезпечення високого рівня обслуговування, спортивним клубам необхідна сучасна, надійна та інтегрована система управління.

Проте, багато спортивних закладів на поточному етапі своєї діяльності стикаються з низкою управлінських та операційних викликів, зумовлених недостатньою автоматизацією бізнес-процесів. Застосування традиційних методів ведення обліку, що базуються на паперових носіях, ручному введенні даних у розрізнені електронні таблиці (наприклад, Excel) або використання окремих, неінтегрованих програмних застосунків, призводить до значних операційних неефективностей. Це створює перешкоди для подальшого масштабування бізнесу, утримання клієнтів та підвищення загальної рентабельності закладу.

Детальний опис ключових проблем, що виникають у діяльності спортивного клубу комплексу LEOLAND та обґрунтовують необхідність впровадження автоматизованої вебсистеми, включає:

- Низьку ефективність управління клієнтською базою та абонементом:
 - Розрізненість даних – Інформація про нових та існуючих клієнтів (контактні дані, історія відвідувань, придбані абонементи, їх терміни дії, персональні переваги) часто зберігається у декількох джерелах, що призводить до дублювання, неузгодженості та складності швидкого доступу до повної картини по кожному відвідувачу.
 - Повільний процес обслуговування – Ручний пошук або перевірка даних клієнта на рецепції займає значний час, створюючи черги та знижуючи швидкість

обслуговування, особливо у пікові години. Це безпосередньо впливає на задоволеність клієнтів.

- Проблеми з лояльністю та утриманням – Відсутність централізованої системи аналізу даних про клієнтів ускладнює своєчасне інформування про необхідність продовження абонементу, надання персоналізованих пропозицій (наприклад, знижок на день народження або спеціальних умов для постійних клієнтів), а також проведення ефективних маркетингових кампаній, що веде до втрати потенційного доходу та зниження рівня утримання клієнтів.

- Складність управління розкладом та неефективність системи бронювання:

- Динамічний розклад – Розклад групових та індивідуальних занять є динамічним, з частими змінами, замінами тренерів, або відмінами через свята чи технічні причини. Ручне оновлення та донесення цієї інформації до клієнтів (через оголошення на стендах, соціальні мережі або телефонні дзвінки) є трудомістким та часто запізнілим процесом.

- Обмеження для клієнтів – Клієнти не мають можливості у зручний для них час (24/7) самостійно переглянути актуальний розклад занять, наявність вільних місць у гуртках, дізнатися інформацію про тренера, чи забронювати місце онлайн. Це призводить до необхідності телефонувати або особисто звертатися до клубу, що є незручним та обмежує спонтанні рішення відвідувати клуб.

- Адміністративні труднощі – Ручне бронювання занять (гуртків, персональних тренувань) адміністраторами призводить до помилок (наприклад, подвійний запис на одне місце), неузгодженостей та займає значний робочий час персоналу, який міг би бути спрямований на більш важливі завдання.

- Відсутність централізованого обліку та аналізу інформації про гуртки та персонал:

- Неповна інформація про гуртки – Деталі про кожен гурток (конкретний графік, актуальна ціна, профіль тренера, опис програми, максимальна кількість учасників).

- Облік зайнятості та продуктивності – Без автоматизованої системи складно відстежувати фактичну завантаженість тренерів, кількість відвідувань

кожного гуртка, його популярність. Це ускладнює оптимізацію розкладу, прийняття рішень щодо відкриття нових напрямків або закриття нерентабельних.

- Управління заявками – Процес розгляду та затвердження/відхилення заявок клієнтів на запис до гуртків є повільним та вимагає постійного втручання адміністратора, що може затримувати процес реєстрації та призводити до втрати потенційних клієнтів.

- Обмежені можливості для аналізу та стратегічного планування:

- Відсутність консолідованої звітності – Збір та аналіз ключових статистичних даних (наприклад, щомісячна відвідуваність, динаміка продажів абонементів, дохід від різних послуг, коефіцієнт утримання клієнтів) вимагає ручної консолідації даних з багатьох джерел. Це є трудомістким, схильним до помилок та забирає багато часу.

- Необґрунтовані рішення – Відсутність оперативного доступу до точних аналітичних звітів перешкоджає керівництву клубу приймати своєчасні та обґрунтовані управлінські рішення щодо його розвитку, оптимізації ресурсів, розробки ефективних маркетингових кампаній та розширення асортименту послуг.

- Втрата конкурентоспроможності на ринку:

- У сучасному цифровому світі клієнти все частіше очікують від спортивних клубів наявності зручних онлайн-сервісів: можливість онлайн-реєстрації, самостійного бронювання занять, перегляду актуального розкладу та повної інформації про послуги. Відсутність такої функціональності у клубі може призвести до відтоку клієнтів до конкурентів, які вже впровадили подібні рішення, тим самим знижуючи частку ринку та прибутковість.

Таким чином, незважаючи на наявність широкого спектру послуг та якісну інфраструктуру, поточний стан управління діяльністю спортивного клубу комплексу LEOLAND вимагає негайного впровадження комплексної вебсистеми. Така система дозволить не лише автоматизувати рутинні операції, але й централізувати управління всіма ключовими бізнес-процесами, суттєво покращити взаємодію з клієнтами шляхом надання їм зручних онлайн-інструментів, оптимізувати роботу

персоналу та забезпечити керівництво надійними даними для стратегічного розвитку.

Впровадження вебсистеми для спортивного клубу комплексу LEOLAND забезпечить наступні значні переваги:

- Підвищення ефективності та швидкості обслуговування клієнтів – Автоматизація процесів реєстрації, бронювання та обліку абонементів суттєво зменшить час очікування та підвищить якість взаємодії.

- Оптимізація внутрішніх бізнес-процесів – Систематизація даних та автоматизація рутинних завдань дозволить адміністраторам та менеджерам зосередитися на стратегічно важливіших завданнях.

- Покращення доступності інформації – Клієнти зможуть у будь-який час переглядати актуальний розклад, інформацію про гуртки, тренерів, ціни та доступність місць.

- Збільшення лояльності клієнтів – Зручні онлайн-сервіси, персоналізовані пропозиції та оперативна комунікація сприятимуть зростанню задоволеності та утримання клієнтів.

- Надання інструментів для прийняття управлінських рішень – Централізований збір даних та можливість генерації аналітичних звітів дозволять керівництву спортивного клубу ефективніше планувати розвиток, оптимізувати ресурси та підвищувати прибутковість.

- Підвищення конкурентоспроможності – Модернізація та цифровізація послуг виведе спортивний клуб на новий рівень, відповідаючи сучасним вимогам ринку та дозволяючи успішно конкурувати з іншими спортивними закладами.

1.2 Аналіз існуючих вебсистем для спортивних клубів

На сучасному ринку програмного забезпечення представлено низку рішень для автоматизації діяльності спортивних клубів та фітнес-центрів. Ці системи варіюються від простих програм для обліку клієнтів до комплексних хмарних ERP-систем. Їх функціонал часто включає управління абонементом, розкладом, фінансовий облік, CRM-модулі та системи онлайн-бронювання.

Загальний огляд існуючих рішень дозволяє виділити такі категорії:

- Коробкові рішення (пропрієтарні системи) – Це готові програмні продукти, які надаються комерційними розробниками (наприклад, FitnesPlan, GigaSport, Fitness CRM):

- Переваги – Зазвичай мають широкий функціонал "з коробки", технічну підтримку, регулярні оновлення.

- Недоліки – Висока вартість ліцензій та обслуговування, обмежені можливості кастомізації під унікальні потреби конкретного клубу, залежність від постачальника, можливі щорічні абонентські плати. Не завжди адаптовані під локальні вимоги та специфіку.

- Хмарні SaaS-рішення (Software as a Service) – Доступ до функціоналу надається через веббраузер за підпискою (наприклад, Mindbody, GymMaster, Bitrix24):

- Переваги – Не потребують інвестицій у власну інфраструктуру, доступність з будь-якого пристрою, автоматичні оновлення, масштабованість.

- Недоліки – Залежність від інтернет-з'єднання, висока щомісячна абонентська плата, потенційні ризики безпеки даних у хмарі, обмежена кастомізація, відсутність повного контролю над даними та функціоналом.

- Індивідуальні розробки – Системи, створені під конкретного замовника:

- Переваги – Повна відповідність усім унікальним потребам та бізнес-процесам клубу, можливість інтеграції з існуючими внутрішніми системами, контроль над кодом та даними.

- Недоліки – Значно вища вартість початкової розробки, довший термін реалізації, необхідність залучення кваліфікованих розробників для підтримки та подальшого розвитку.

Аналіз існуючих рішень показує, що, хоча на ринку присутні різноманітні системи, вони не завжди повною мірою відповідають усім потребам спортивного клубу комплексу LEOLAND.

Враховуючи прагнення спортивного клубу комплексу LEOLAND до оптимізації витрат, забезпечення високої гнучкості системи та можливості її

подальшого розвитку під власні унікальні потреби, оптимальним рішенням є розробка індивідуальної вебсистеми. Такий підхід дозволить створити інструмент, який буде точно відповідати бізнес-процесам комплексу, забезпечуючи необхідний функціонал без зайвих елементів, а також гарантуючи повний контроль над даними та безпекою.

РОЗДІЛ 2. ІНФОРМАЦІЙНЕ ТА МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ

2.1 Обґрунтування вибору технологій розробки

Вибір технологічного стеку є критично важливим етапом у розробці вебсистеми, оскільки він визначає її продуктивність, масштабованість, безпеку, гнучкість у подальшому розвитку та вартість підтримки. Для розробки вебсистеми діяльності спортивного клубу комплексу LEOLAND було обрано наступний стек технологій: PHP для серверної частини (Backend), MySQL як система керування базами даних, JavaScript для розробки клієнтської частини (Frontend) у поєднанні з CSS для стилізації, а також інструменти розробки Visual Studio Code та локальний вебсервер XAMPP. Цей вибір ґрунтується на аналізі вимог до системи, переваг кожної технології та досвіду розробки подібних вебзастосунків.

2.2 Вибір потрібного інформаційного забезпечення

Одним з нелегких завдань є правильне обрання та обґрунтування програмних засобів для розв'язання задачі. Це є чудовим механізмом підтвердження того, що система, яку необхідно розробити, дійсно буде розвиватись та вдосконалюватись. Це допомагає гарантувати, що проєкт розв'язує конкретну проблему і пояснює, чому це рішення краще в даній області.

Серед великої кількості інструментів, які в процесі розробки пришвидшують та спрощують розробку продукту, слід обрати найбільш оптимальний варіант. При виборі інструментів потрібно звернути увагу на ряд факторів, а найбільш уважно потрібно зосередитися на їх функціоналі. Не всі інструменти обов'язково підходять до даного проєкту, в той же час як інші можуть мати вирішальне значення. Необхідна функціональність і характер застосування програмного забезпечення системи повинні впливати на обрання методів та інструментів, які використовуватимуться в процесі розробки програмного забезпечення.

Для розробки даної вебсистеми було обрано Visual Studio Code як середовище розробки, та мови програмування PHP і JavaScript як інструменти розробки ПЗ. Дана робота спрямована на реалізацію вебсистеми діяльності спортивного клубу. Для

цього було використано середовище розробки VS Code у поєднанні з локальним вебсервером XAMPP, та інструмент MySQL Workbench для візуального проєктування бази даних СКБД MySQL. Інструменти розробки обрані на підставі означених цілей системи та особливостей її використання.

Оскільки система з вебінтерфейсом повинна містити в собі багато динамічно генерованих даних та водночас забезпечувати доступ до них за допомогою мережі Інтернет, нами була обрана мова програмування PHP для серверної частини, та JavaScript для клієнтської частини. Для графічного оформлення інтерфейсу та відображення структури системи ми використали мову гіпертекстової розмітки HTML та каскадні таблиці стилів CSS. Для зберігання інформації було обрано СКБД MySQL. Для розробки системи локально на комп'ютері було використано вебсервер XAMPP.

PHP (Hypertext Preprocessor) – скриптова мова програмування, яка призначена для веб-розробки. Даний мова програмування дуже популярна у наш час і найбільш поширена у використанні. Вона містить всі необхідні засоби, які використовують під час створення різних проєктів, коли їх об'єднати в одне ціле, то отримаємо готовий проєкт. Ця мова програмування підтримується усіма провайдерами – це одна з її позитивних сторін [10].

Код на мові PHP можна писати безпосередньо у HTML-коді сторінок, які у свою чергу, будуть оброблені PHP-інтерпретатором. Код починає виконуватися коли бачить відкриваючий тег (`<?php`) а зупиняється коли бачить закриваючий тег (`?>`). Завдяки своїм функціоналам і великою кількістю функцій, вона дозволяє забути про написання великих рядків коду (різних методів), як це можна спостерігати у подібних мовах програмування. Дана мова містить усі необхідні стандартні бібліотеки для роботи з PostgreSQL, Informix, MySQL, Oracle, Hyperware, InterBase.

JavaScript (JS) – це мова програмування, розроблена компанією Netscape Communications, основним призначенням якої є створення і розробка, як клієнтських так і серверних інтернет-застосунків. JS – це об'єктно-орієнтована і динамічно типізована мова програмування [11], реалізована за стандартом ECMAScript. Дана мова програмування зазвичай використовується як вбудований елемент, за

допомогою якого відбувається взаємодія вебсторінки з браузером. Вона дозволяє асинхронно керувати і витягувати дані з вебсервера, керувати зовнішнім виглядом і структурою сторінки. На JavaScript вплинули багато мов програмування. При її розробці метою було отримати мову схожу на Java, але при цьому легку для використання програмістами початківцями [1].

До основні архітектурних особливостей можна віднести такі властивості: динамічна типізація, прототипне програмування, наявність автоматичного управління пам'яттю, не сильна типізація. Саме ці, основні властивості дозволили JS зайняти провідне місце серед інших мов програмування.

jQuery – одна із багатьох бібліотек JavaScript, яка фокусується на взаємодії HTML, CSS та JavaScript. Дана бібліотека допомагає легко отримувати доступ до будь-якого елемента, розташованого на сторінці, звертатися до атрибутів і вмісту сторінки, керувати нею за допомогою селекторів. Також бібліотека jQuery надає зручний API для роботи з асинхронними AJAX запитам до сервера [12].

HTML (Hyper Text Markup Language) – це стандартна мова призначена для створення гіпертекстових документів, саме завдяки їй створюється більшість вебсторінок. HTML файл обробляється браузером та відтворюється на екрані у звичному для людини вигляді [2].

CSS (Cascading Style Sheets) – мова стилів, що характеризує відображення HTML документів у браузері. Стиль – це набір певних параметрів, які задають зовнішній вигляд об'єкта. За допомогою CSS, можна суттєво скоротити розмір кода і зробити його читабельним. Дозволяє задавати такі параметри, які не можна задати лише самим HTML [5]. Наприклад, CSS працює з шириною, кольорами, висотою, зображеннями, рядками, шрифтами, позиціонуванням елементів, полями і багатьма іншими властивостями.

Для роботи з базами даних використовувалася структурована мова SQL (Structured Query Language). Дана мова використовується для формування запитів, керування реляційною базою даних (видалення і оновлення даних). Дозволяє створювати і модифікувати різні за складністю схеми даних, контролювати доступ до інформації збереженої у БД. База даних зберігає у собі різні дані – від простого

списку до каталогу товарів інтернет-магазину. Усі дані в БД є структурованими. Для роботи з БД використовується СКБД, яка має назву MySQL. Вона дозволяє керувати БД, а саме видаляти, додавати, обробляти дані, здійснювати різні маніпуляції з даними.

У реляційній БД, дані зберігаються не всі разом, а в окремих таблицях, що дозволяє досягнути високої швидкодії і гнучкості. В такій БД є можливість пов'язати таблиці між собою, використовуючи відношення. Це дозволяє виконувати надскладні запити до даних з декількох таблиць.

Інструментом для візуального проєктування таких БД є MySQL Workbench (рис. 2.1). Цей інструмент об'єднує моделювання, створення, проєктування і експлуатації бази даних в одне ціле [6] і користується такими основними перевагами:

- наявністю утиліти, яка перевіряє моделі даних і повідомляє про помилки;
- має візуальний редактор;
- можна створити декілька моделей в одному середовищі;
- підтримує таблиці, види, процедури, які в сукупності утворюють БД;
- наявний графічний інтерфейс, який дозволяє писати і редагувати запити до вебсервера.

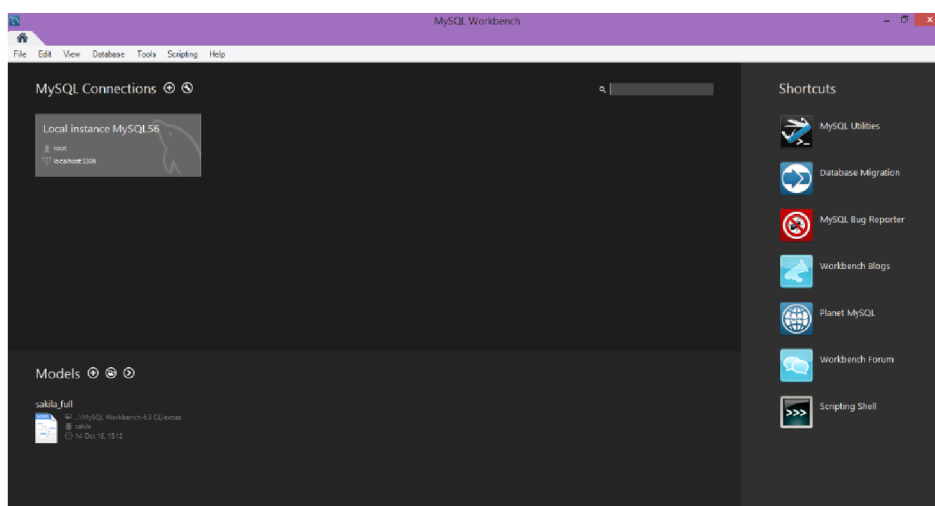


Рисунок 2.1 – Інтерфейс інструменту MySQL Workbench

Щоб наша вебсистема працювала її необхідно розмістити на вебсервері [7]. Мій проєкт буде розміщуватися на локальному сервері, а найкраще для цього підходить XAMPP (рис. 2.2). Це збірка вебсервера, що містить вебсервер Apache,

MySQL, мову програмування PHP, а також усі необхідні бібліотеки, які дозволяють запуснути повноцінний вебсервер на комп'ютері. Вона найкраще підходить для тестування роботи, без використання Інтернету.

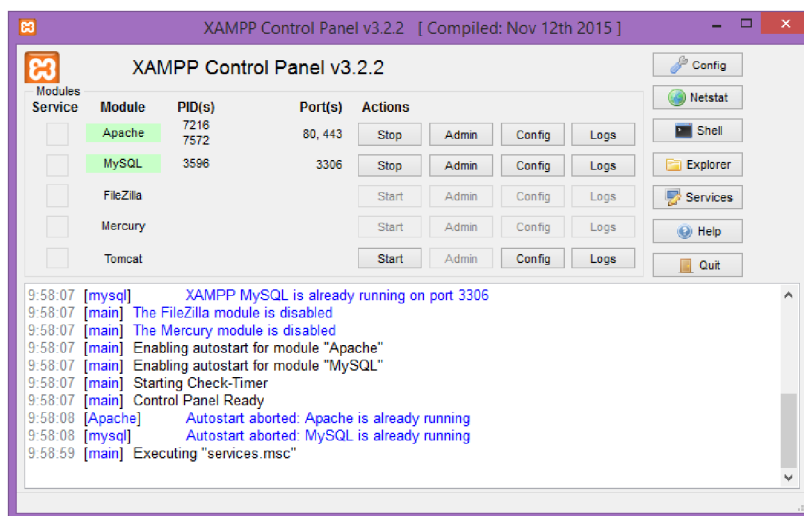


Рисунок 2.2 – Інтерфейс збірки вебсервера XAMPP

Для редагування і написання коду було використано інтегроване середовище розробки Visual Studio Code (рис. 2.3), розроблене Microsoft для Windows, Linux і macOS. Воно містить в собі інструменти для роботи з Git, наявний пакетний менеджер, зручний інтерфейс, навігаційну панель, перевірку синтаксису, автозберігання і автодоповнення коду. Дозволяє користувачеві самостійно вибрати тему редактора, колір і розмір тексту, налаштування гарячих клавіш, розробляється як ПЗ з відкритим вихідним кодом.

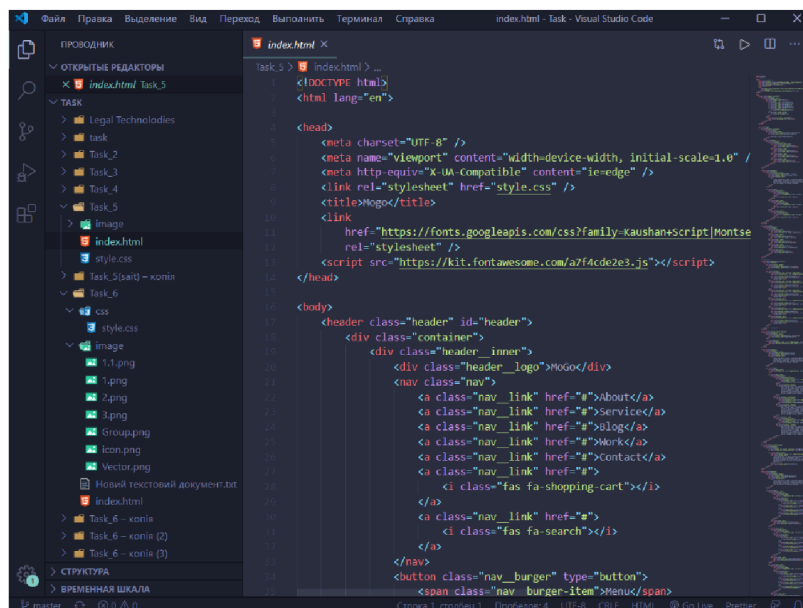


Рисунок 2.3 – Інтерфейс середовища розробки Visual Studio Code

2.3 Побудова дерева проблем та дерева цілей

Для створення своєї вебсистеми необхідно виявити, впорядкувати та розв'язати ряд проблем. Для цього потрібно побудувати дерево цілей для системи.

Дерево цілей – це наочне графічне відображення взаємозв'язку та підпорядкованості цілей, що показує розподіл загальної (основної) мети або місії на завдання, під цілі та окремі дії [3].

Побудова дерева цілей репрезентує тільки методику розроблення стратегії здобутку установленної головної мети, а саме розроблення вебсистеми із використанням інтернет ресурсів. Правила побудови дерева: генеральна мета зобов'язана бути вершиною дерева, гілками стають локальні цілі, які допомагають досягти цілей верхнього рівня.

Головне правило – це повнота описаних цілей [4]. Кожна мета повинна бути представлена у вигляді під цілей наступного ступеня. Отже, об'єднання усіх цілей має повністю характеризувати основну, генеральну мету. Дерево цілей для системи відображено на рисунку 2.4.



Рисунок 2.4 – Дерево цілей вебсистеми

2.4 Аналіз і вибір методів, алгоритмів та засобів розв'язання задачі

Для розробки архітектури інтерактивної вебсистеми була взята клієнт-серверна архітектура. Обрана архітектура найчастіше використовується в роботі з БД та мережі і забезпечує обмін даними між вказаними компонентами. Архітектура клієнт-сервер передбачає такі три основні компоненти:

- сервери, що обробляють отримані запити та видають відповідний результат;
- клієнти, що звертаються до серверів з запитом про дані;
- мережа, що забезпечує обмін даними між клієнтами та серверами.

Обробка та збереження даних відбувається на боці вебсервера, відображення даних і надсилання запитів на вебсервер виконується на боці клієнта.

На рисунку 2.5 зображена трирівнева схема архітектури вебсистеми.

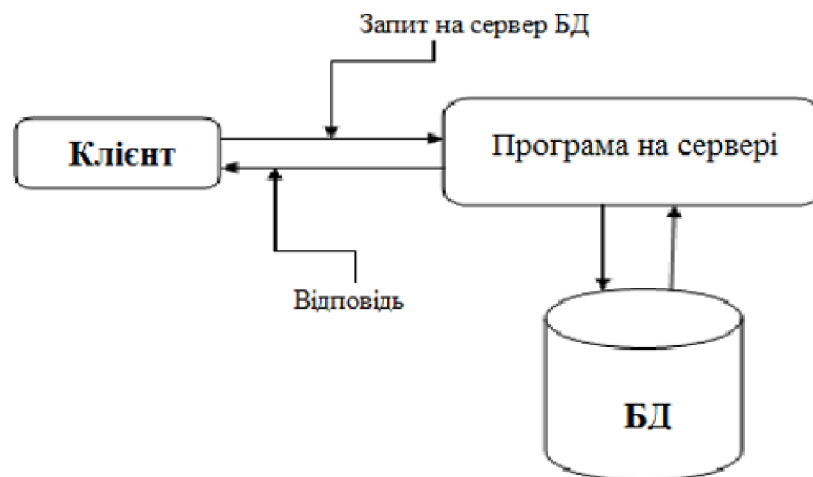


Рисунок 2.5 – Схема клієнт-серверної архітектури

Перший рівень – це клієнт, який відсилає запити на вебсервер та приймає результат обробки запитів. Клієнтом є браузер користувача.

Другий рівень – це бізнес-логіка вебсистеми. Це логіка, за якою вебсервер обробляє отримані від клієнта запити.

Третій рівень – це сама СУБД, яка отримує запити від вебсервера і повертає потрібні дані на вебсервер або зберігає їх.

РОЗДІЛ 3. ПРОГРАМНЕ ТА ТЕХНІЧНЕ ЗАБЕЗПЕЧЕННЯ

3.1 Структура даних та схема бази даних

Адміністратор БД спочатку створює узагальнений неформальний опис створюваної БД. Інфологічна модель даних – це модель, яка містить опис, виконаний із використанням природної мови, математичних формул, таблиць, графіків і інших засобів, зрозумілих усім, хто працює над проектуванням баз даних.

Ціль інфологічного моделювання – забезпечення найбільш дійсних для людини способів збору та представлення тієї інформації, яку зберігають в створюваній БД. Основними конструктивними елементами інфологічних моделей є дійсності, зв'язки між ними та їх властивості (атрибути).

При визначенні інфологічної моделі необхідно брати до уваги таке, що БД повинна:

- задовольняти актуальним інформаційним потребам організації;
- забезпечувати одержання необхідних даних за прийнятний час, тобто відповідати заданим вимогам продуктивності;
- задовольняти виявленим і знову виникаючим вимогам усіх користувачів;
- розширюватися при реорганізації і розширенні предметної області;
- змінюватися при зміні програмного й апаратного середовища.

Інфологічна модель відображає реальний світ у деякій зрозумілій людині концепції, цілком незалежній від параметрів середовища збереження даних. Існує множина підходів до побудови таких моделей: семантичні мережі, модель “сутність-зв'язок”, графові моделі. Найбільше популярної з них виявилася модель “сутність-зв'язок” або інша її назва ER-модель (від англ. Entity-Relationship).

Дана інфологічна модель показує, які об'єкти повинні зберігатися в БД, яким чином вони описані (структуровані) та як вони пов'язані між собою. Тобто модель відбиває дані через їх склад, структуру та взаємозв'язки між ними, що дозволяють впливати на об'єкт змінюючи інший.

Моя вебсистема – це багатокористувацька модель, так як її може використовувати необмежена кількість користувачів, а також її адміністрування

може здійснюватись одним або декількома адміністраторами.

У кожного користувача буде унікальний логін та пароль, які він створює при реєстрації в систему, звичайно, якщо цей логін вже не використовує один з уже зареєстрованих користувачів. Після реєстрації, користувач зможе вільно заходити в систему у будь-який час та з будь-якого пристрою, що має підключення до мережі Інтернет.

Важливим компонентом будь-якої вебсистеми є база даних. Моделлю БД називають графічне або текстове представлення БД, де виділяються сутності, атрибути та зв'язки між ними. Згідно з побудованою моделлю варіантів використання моя вебсистема повинна зберігати всю інформацію в БД.

Для проєктування таблиць БД мною було використано інструмент MySQL Workbench.

За допомогою цього застосунку я спроектував БД та таблиці, які потрібні для її коректної роботи на сайті.

Для даної системи створено інфологічну модель зі структурою БД представленою на рисунку 3.1.

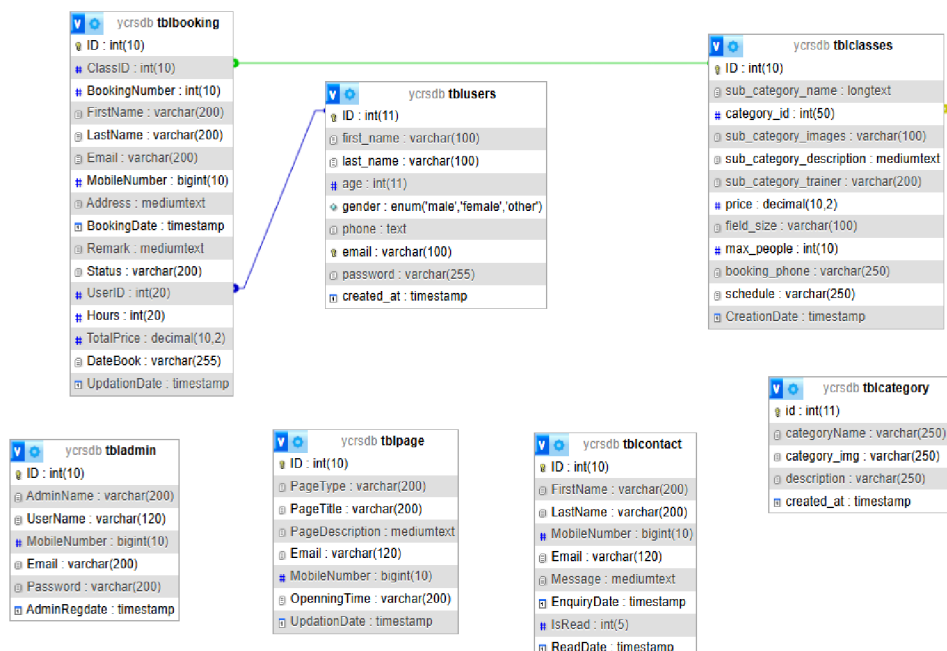


Рисунок 3.1 – Структура бази даних

Зв'язок – це асоціювання двох чи більше об'єктів. Якщо б призначенням бази даних було тільки зберігання відокремлених, не пов'язаних між собою даних, то її

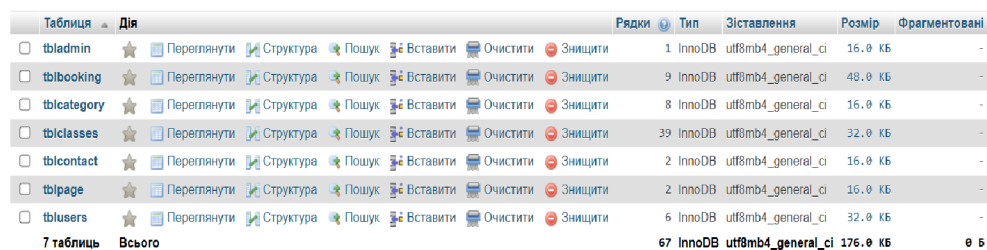
структура могла бути досить простою. Однак однією з основних вимог до організації БД – це забезпечення можливості пошуку одних об’єктів за значеннями інших, для чого необхідно встановити між ними певні зв’язки. А так як в реальних БД нерідко міститься сотні чи навіть тисячі об’єктів, то теоретично між ними може бути встановлено велика кількість зв’язків. Наявність такої множини зв’язків і визначає тяжкість інфологічних моделей.

Інфологічна модель виступає початковим етапом в проєктуванні БД, вона допомагає наочно представити роботу вебсистеми та в подальшому побудувати таблиці та зв’язки між ними.

При створенні інфологічної моделі для побудови ER-діаграм можна використовувати редактор Microsoft Visio.

Загалом було створено базу даних, яка містить 7 таблиць. Після проєктування БД в середовищі програми було отримано SQL-код, який я спочатку експортував у текстовий файл, а потім його імпортував у вебзастосунок PhpMyAdmin, що викликається зі збірки вебсервера XAMPP (рис 3.2).

Для коректної роботи з вебсайтом дані таблиць БД були закодовані у форматі UTF-8.



Таблиця	Дія	Рядки	Тип	Зіставлення	Розмір	Фрагментовані
<input type="checkbox"/> tbladmin	☆ Переглянути Структура Пошук Вставити Очистити Знищити	1	InnoDB	utf8mb4_general_ci	16.0 КБ	-
<input type="checkbox"/> tblbooking	☆ Переглянути Структура Пошук Вставити Очистити Знищити	9	InnoDB	utf8mb4_general_ci	48.0 КБ	-
<input type="checkbox"/> tblcategory	☆ Переглянути Структура Пошук Вставити Очистити Знищити	8	InnoDB	utf8mb4_general_ci	16.0 КБ	-
<input type="checkbox"/> tblclasses	☆ Переглянути Структура Пошук Вставити Очистити Знищити	39	InnoDB	utf8mb4_general_ci	32.0 КБ	-
<input type="checkbox"/> tblcontact	☆ Переглянути Структура Пошук Вставити Очистити Знищити	2	InnoDB	utf8mb4_general_ci	16.0 КБ	-
<input type="checkbox"/> tblpage	☆ Переглянути Структура Пошук Вставити Очистити Знищити	2	InnoDB	utf8mb4_general_ci	16.0 КБ	-
<input type="checkbox"/> tblusers	☆ Переглянути Структура Пошук Вставити Очистити Знищити	6	InnoDB	utf8mb4_general_ci	32.0 КБ	-
7 таблиць	Всього	67	InnoDB	utf8mb4_general_ci	176.0 КБ	0 Б

Рисунок 3.2 – Структура БД в phpMyAdmin

3.2 Розробка інтерфейсу користувача та опис роботи вебсистеми

Графічний інтерфейс користувача призначений для зручної взаємодії користувача з вебсистемою.

Розроблений інтерфейс вебсистеми представлений HTML-сторінками з використанням CSS, які пов’язані з БД з допомогою скриптової мови PHP.

Огляд вебсистеми починається з її головної сторінки, яка є візитною карткою спортивного клубу комплексу LEOLAND. На ній представлена загальна інформація, ключові секції та елементи навігації (рис. 3.3).

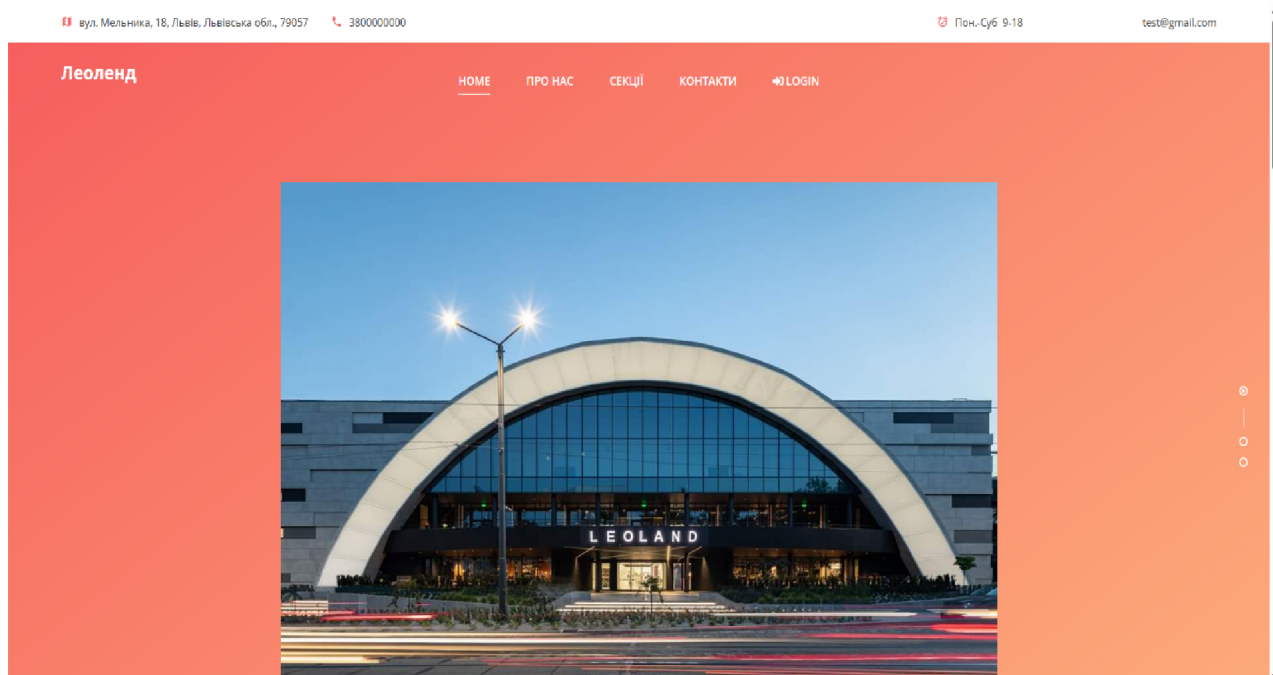


Рисунок 3.3 – Головна сторінка вебсистеми

На панелі навігації, розташованій у верхній частині сторінки, відображається логотип “Леоленд” та основні пункти меню: “HOME”, “ПРО НАС”, “СЕКЦІЇ”, “КОНТАКТИ”. Праворуч розташовані кнопки “LOGIN” або “УВІЙТИ”, а також “ADMIN / УВІЙТИ” для доступу до адміністративної панелі. У верхньому лівому куті відображається контактна інформація: адреса, номер телефону та електронна пошта, а також час роботи спортивного клубу. Нижня навігаційна панель (footer) містить інформаційний текст авторських прав сайту “© Система реєстрації клубів 2025. Усі права захищені” (рис. 3.4).

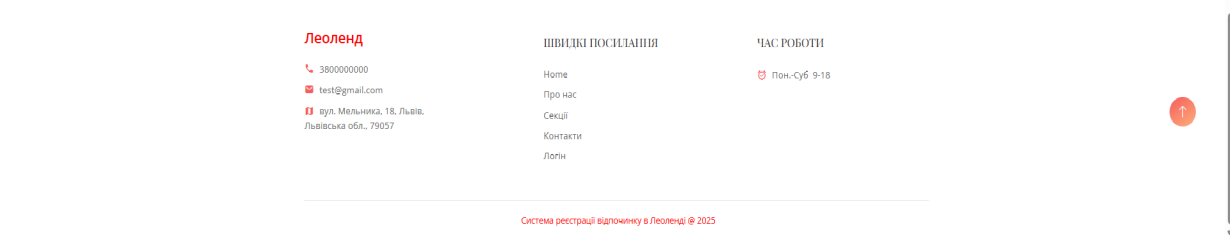


Рисунок 3.4 – Нижній колонтитул (footer) вебсистеми

Сторінки входу та реєстрації розроблені з акцентом на інтуїтивну зрозумілість та мінімалізм, що забезпечує комфортний та швидкий процес для користувача. Сторінка входу (рис. 3.5) містить поля для логіна та пароля, а також посилання “Забув свій пароль” та “Новий акаунт? Зареєструватися”.

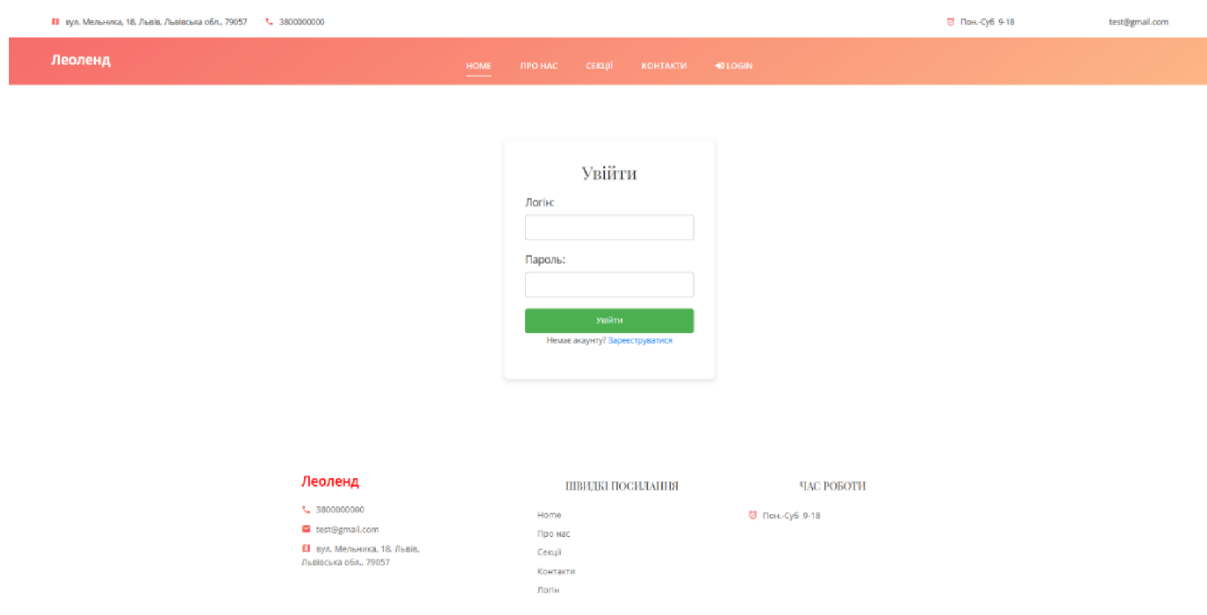


Рисунок 3.5 – Сторінка входу

Форма реєстрації (рис. 3.6) включає поля для введення імені, прізвища, віку, статі, номера телефону, пошти та пароля користувача, з кнопкою “Зареєструватися” та посиланням “Вже маєте акаунт? Увійти”.

Реєстрація

Ім'я:
Ігор

Прізвище:
Ковальчук

Вік:
30

Стать:
Чоловіча

Номер Телефона:
0981111111

Пошта:
igor111@gmail.com

Пароль:
....

Зареєструватися

[Вже маєте акаунт? Увійти](#)

Рисунок 3.6 – Сторінка реєстрації

Після успішної реєстрації та входу користувач отримує доступ до персоналізованого інтерфейсу. На верхній навігаційній панелі замість “LOGIN” з’являється ім’я авторизованого користувача, наприклад, ІГОР КОВАЛЬЧУК, що свідчить про успішну авторизацію та надає доступ до особистих даних та історії записів (рис. 3.7).

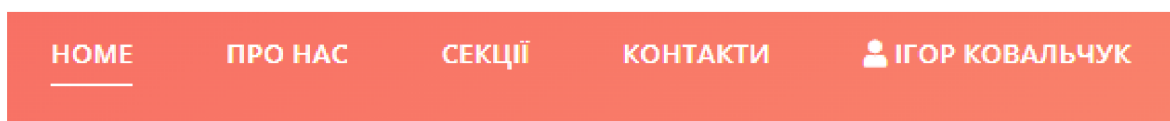


Рисунок 3.7 – Верхня панель навігації для зареєстрованого користувача

Розділ “НАШІ СЕКЦІЇ” (рис. 3.8) представляє огляд усіх доступних категорій гуртків спортивного клубу, таких як “Футбольні поля”, “Плавання”, “Йога”, “Тренажерний зал”, “Баскетбольні майданчики”, “Танці”, “Карате”, “Бокс”. Кожна категорія представлена візуальною карткою з відповідним зображенням та короткою назвою, що дозволяє клієнту швидко орієнтуватися у пропозиціях клубу.

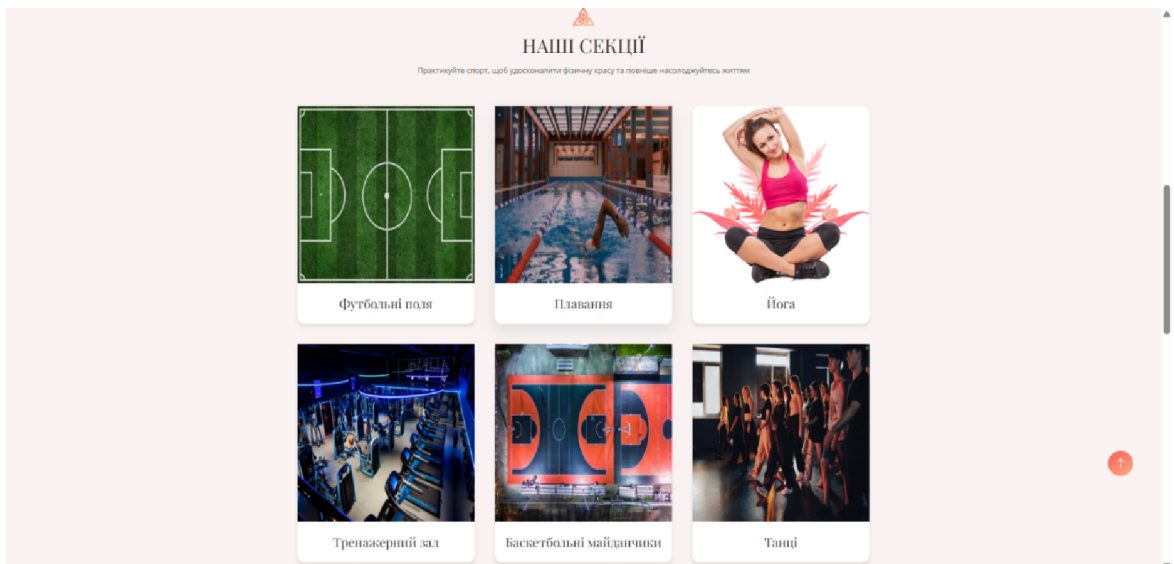


Рисунок 3.8 – Огляд категорій секцій

При виборі конкретної категорії, наприклад, “Футбольні поля”, відкривається сторінка з переліком доступних гуртків у цій категорії (рис. 3.9). Кожен гурток представлений окремою картою з його назвою, коротким описом та кнопкою “ЗАБРОНЮВАТИ”.

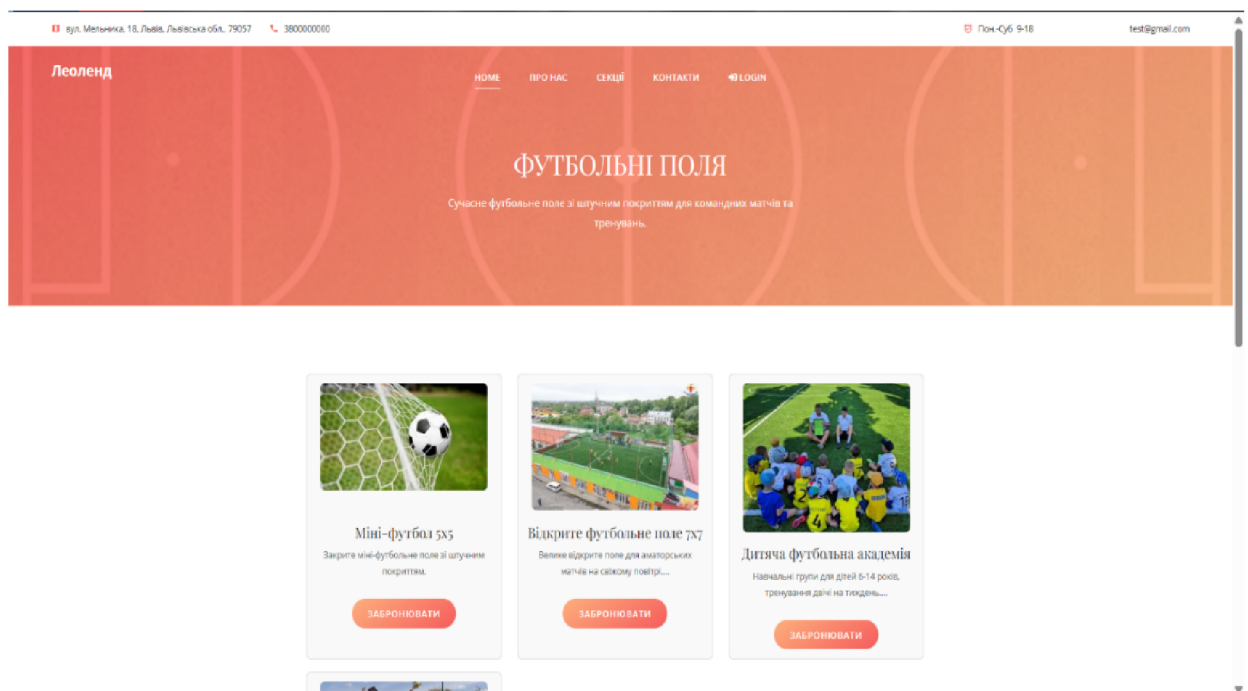


Рисунок 3.9 – Сторінка категорії “Футбольні поля”


При виборі конкретного гуртка, наприклад, “Міні-футбол 5x5”, відкривається детальна сторінка гуртка (рис. 3.10). Тут представлена повна інформація:

зображення, детальний опис, ціна, розмір поля, максимальна кількість людей, телефон для бронювання та розклад. Кнопка "ЗАБРОНЮВАТИ" дозволяє ініціювати процес запису на заняття.



Рисунок 3.10 – Детальна сторінка гуртка “Міні-футбол 5x5”

Процес бронювання передбачає заповнення форми (рис. 3.11), де відображаються дані про гурток (зображення, тренер, опис, ціна за годину, кількість годин, загальна вартість). Користувач вводить свої контактні дані (ім'я, прізвище, пошта, номер телефону) та дату бронювання. Після заповнення всіх полів, натискання кнопки “ОФОРМИТИ ЗАМОВЛЕННЯ” відправляє заявку.

Зображення	Тренер	Опис	Ціна за годину	Кількість годин	Загальна вартість
	Доступний за запитом (500 грн/год)	Закрите міні-футбольне поле зі штучним покриттям.	500.00 грн	2 год.	1000 грн

Ігор

Ковальчук

ihor111@gmail.com

12-06-2025

0981111111

ОФОРМИТИ ЗАМОВЛЕННЯ

Рисунок 3.11 – Форма оформлення замовлення (бронювання)

Після успішного оформлення замовлення, система відображає сторінку підтвердження (рис. 3.12) з повідомленням “Дякуємо за бронювання!”. Тут вказано, що бронювання успішно оформлено і команда зв’яжеться для підтвердження деталей. Користувачу пропонується можливість переглянути свій профіль, натиснувши кнопку “Переглянути ваш профіль”, де він може відстежити статус своїх замовлень.

Дякуємо за бронювання!

Ваше бронювання успішно оформлено. Наша команда зв’яжеться з вами найближчим часом для підтвердження всіх деталей.

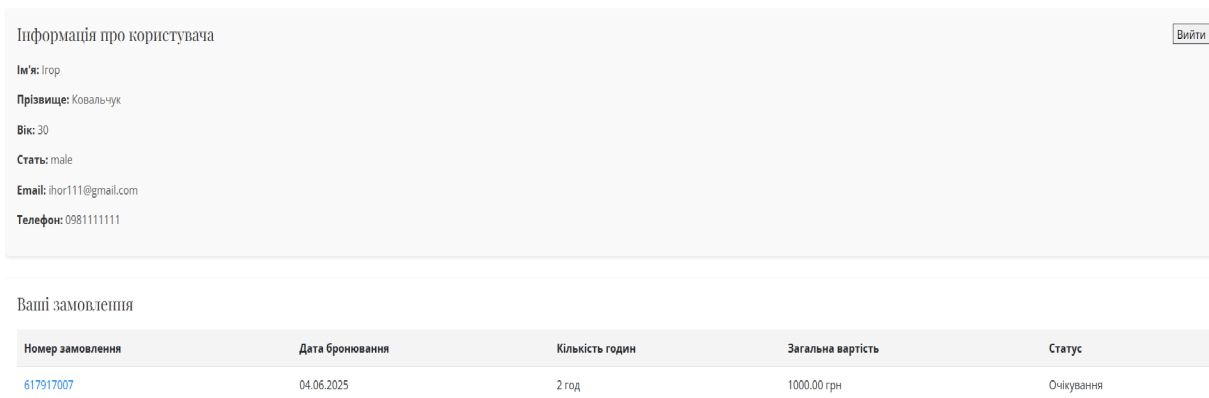
У разі виникнення запитань, ви можете звернутися до нас через профіль. Ви також можете переглянути ваші замовлення в будь-який час.

Переглянути ваш профіль

Бажаємо вам чудового дня!

Рисунок 3.12 – Сторінка підтвердження бронювання

Особистий кабінет клієнта (рис. 3.13) надає доступ до “Інформації про користувача” (ім’я, прізвище, вік, стать, email, телефон) та розділу “Ваші замовлення”, де відображається таблиця з номером замовлення, датою бронювання, кількістю годин, загальною вартістю та статусом замовлення.



Інформація про користувача Вийти

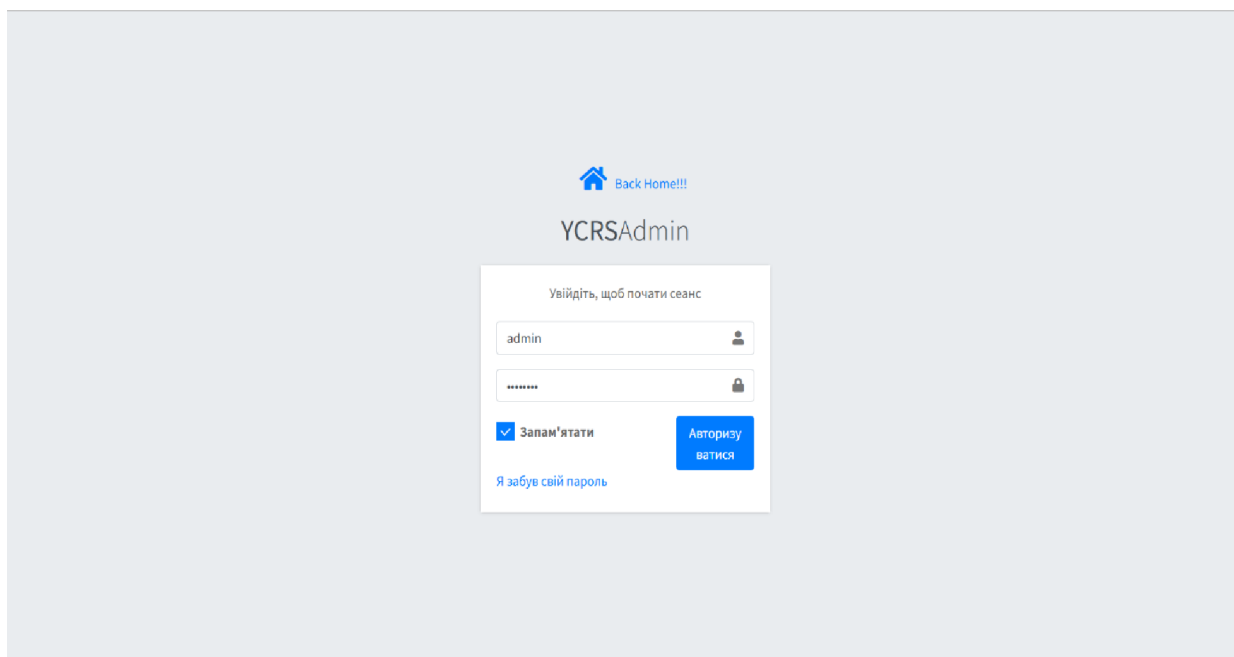
Ім'я: Igor
Прізвище: Ковальчук
Вік: 30
Стать: male
Email: ihor111@gmail.com
Телефон: 0981111111

Ваші замовлення

Номер замовлення	Дата бронювання	Кількість годин	Загальна вартість	Статус
617917007	04.06.2025	2 год	1000.00 грн	Очікування

Рисунок 3.13 – Особистий кабінет клієнта

Для адміністраторів системи передбачена окрема панель управління, доступ до якої здійснюється через сторінку входу адміністратора (рис. 3.14).



Back Home!!!

YCRSAdmin

Увійдіть, щоб почати сеанс

admin

.....

Запам'ятати Авторизуватися

[Я забув свій пароль](#)

Рисунок 3.14 – Сторінка входу адміністратора

Після успішної авторизації, адміністратор потрапляє на “Приладову панель” (Dashboard) (рис. 3.15), яка надає швидкий огляд ключових показників: кількість нових, затверджених, скасованих та загальних бронювань.

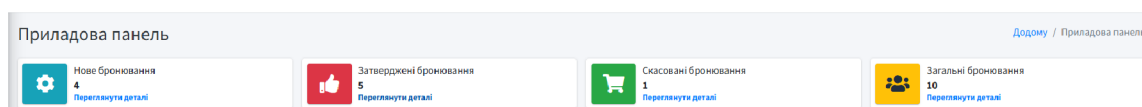


Рисунок 3.15 – Приладова панель адміністратора

Навігація адміністративної панелі (рис. 3.16) включає розділи: “Інформаційна панель”, “Категорія” (Додати категорію, Керувати категоріями), “Класи” (Додати гурток, Керувати гуртками), “Бронювання” (Нове бронювання, Схвалене бронювання, Скасоване бронювання, Усі бронювання), а також “Адміністратор” (Профіль, Змінити пароль, Вийти).

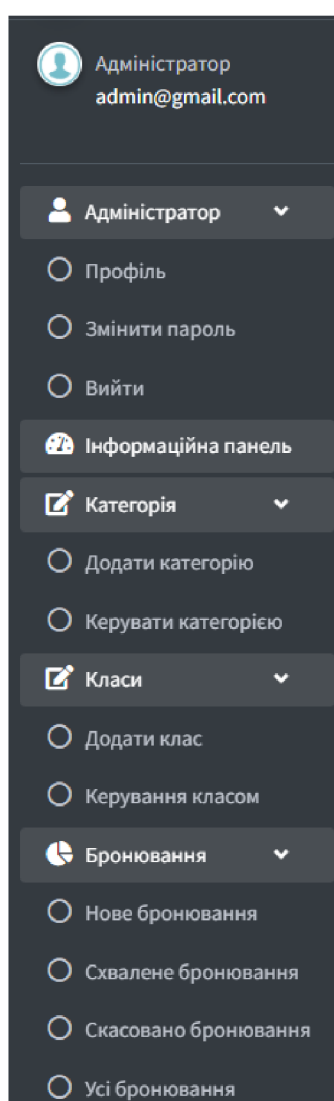
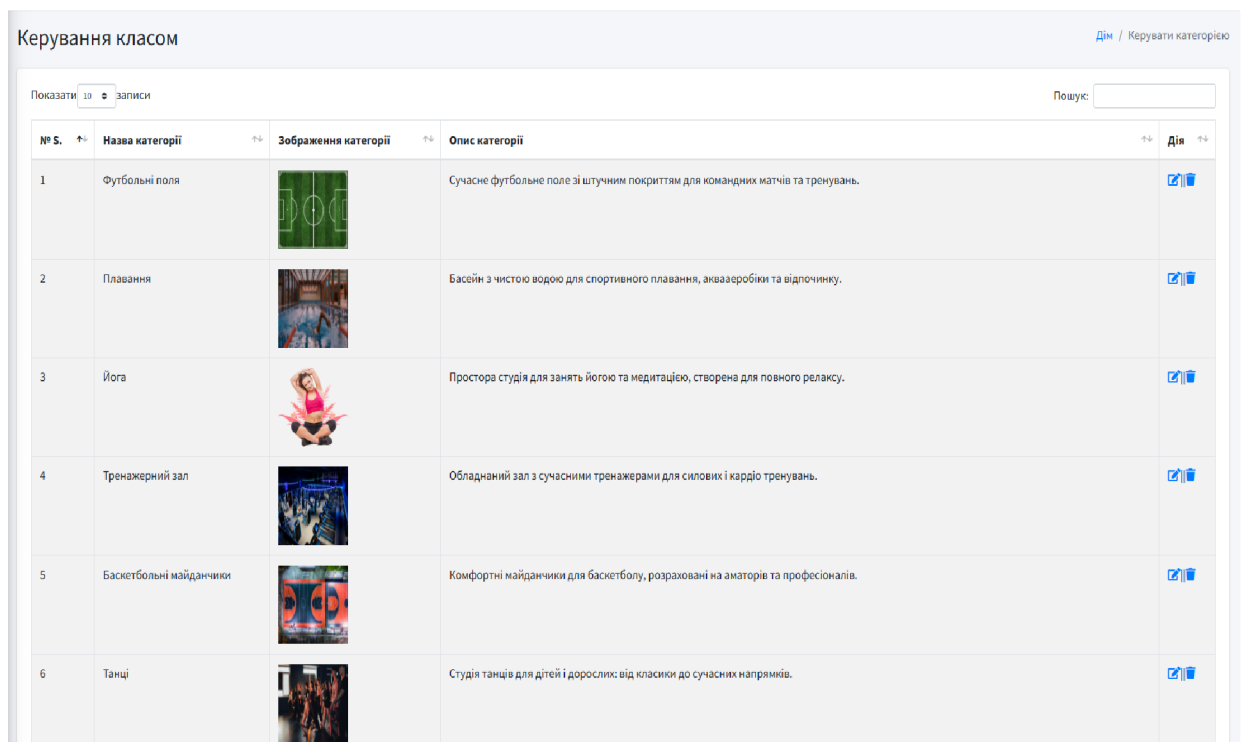


Рисунок 3.16 – Бічна панель навігації адміністратора

Модуль управління категоріями дозволяє адміністратору керувати основними видами спортивних напрямків (наприклад, “Футбольні поля”, “Плавання”, “Йога”).

Сторінка “Керування категорією” (рис. 3.17) відображає таблицю з існуючими категоріями, їх зображеннями, описами та діями (редагувати, видалити).



Керування класом

Показати 10 записи

Пошук:



















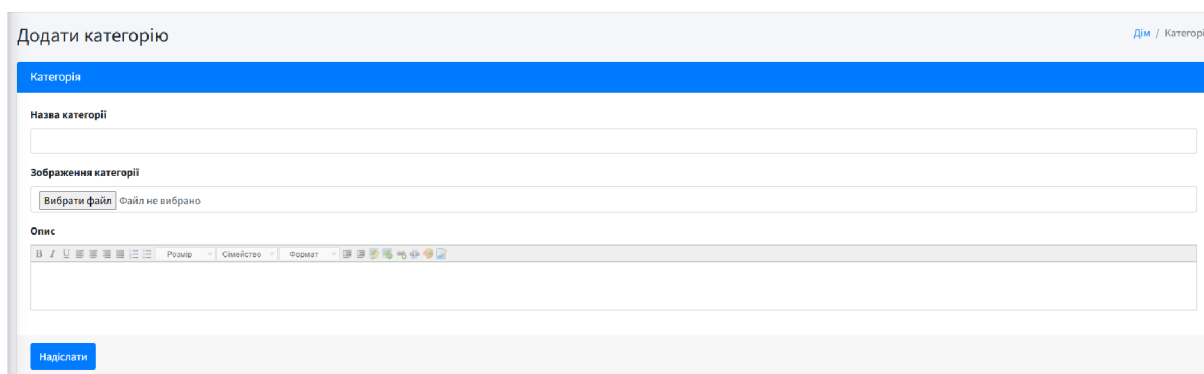
№ С.	Назва категорії	Зображення категорії	Опис категорії	Дія
1	Футбольні поля		Сучасне футбольне поле зі штучним покриттям для командних матчів та тренувань.	 
2	Плавання		Басейн з чистою водою для спортивного плавання, аквааerобіки та відпочинку.	 
3	Йога		Простора студія для занять йогою та медитацією, створена для повного релаксу.	 
4	Тренажерний зал		Обладнаний зал з сучасними тренажерами для силових і кардіо тренувань.	 
5	Баскетбольні майданчики		Комфортні майданчики для баскетболу, розраховані на аматорів та професіоналів.	 
6	Танці		Студія танців для дітей і дорослих від класики до сучасних напрямків.	 

Рисунок 3.17 – Сторінка “Керування категорією” панелі адміністратора

Для додавання нової категорії адміністратор використовує форму “Додати категорію” (рис. 3.18), де вводить назву категорії, завантажує зображення та надає опис.



Додати категорію

Категорія

Назва категорії

Зображення категорії

Вибрати файл

Файл не вибрано

Опис

Надіслати

Рисунок 3.18 – Сторінка "Додати категорію" панелі адміністратора

Модуль управління гуртками (“Класи”) дозволяє адміністратору керувати конкретними заняттями. Сторінка “Керування класом” (рис. 3.19) відображає таблицю з усіма гуртками, включаючи їх назву, категорію, зображення, опис,

тренера, ціну, розмір поля, максимальну кількість людей, контактний телефон для бронювання, розклад та дату створення, а також кнопки для редагування та видалення.

Керування класом Дім / Керування класом

Показати 10 записи Пошук:













№ S. ↑↓	Назва підкатегорії ↑↓	Ідентифікатор категорії ↑↓	Підкатегорія Зображення ↑↓	Опис підкатегорії ↑↓	Тренер ↑↓	Ціна ↑↓	Розмір поля ↑↓	Макс Піпл ↑↓	Телефон для бронювання ↑↓	Розклад ↑↓	Дата створення ↑↓	Дія ↑↓
1	Міні-футбол 5x5	1		Закрийте міні-футбольне поле зі штучним покриттям.	Доступний за запитом (500 грн/год)	500,00	42 м × 22 м	10	+380 67 123 45 67	Понеділок - п'ятниця: 16:00 - 18:00	2025-03-22 21:06:49	 
2	Відкрийте футбольне поле 7x7	1		Велике відкрите поле для аматорських матчів на свіжому повітрі.	Олександр Коваль	700,00	60 м × 30 м	14	+380 67 234 56 78	10:00 - 20:00	2025-03-22 21:50:38	 
3	Дитяча футбольна академія	1		Навчальні групи для дітей 6-14 років, тренування двічі на тиждень.	Олександр Коваль	2000,00		0	+380 67 345 67 89	Понеділок, Четвер 17:00 - 18:30	2025-03-22 21:52:23	 
4	Корпоративні турніри	1		Організація турнірів для компаній із суддівством та призами.		5000,00		0	+380 67 456 78 90	За попереднім записом	2025-03-22 21:57:22	 

Рисунок 3.19 – Сторінка “Керування класом” панелі адміністратора

Для додавання нового гуртка адміністратор використовує форму “Додати клас” (рис. 3.20), де вибирає категорію, вводить назву, опис, тренера, ціну, розмір поля, максимальну кількість людей та розклад.

Додати

Вибірть категорію

Назва категорії

Зображення категорії Файл не вибрано

Опис

Тренер категорії

Ціна розділу

Розмір поля

Максимально людей

Дні та час

Телефон для бронювання

Рисунок 3.20 – Сторінка “Додати клас” панелі адміністратора

Модуль управління бронюваннями надає адміністратору повний контроль над заявками клієнтів. Сторінка “Нове бронювання” (рис. 3.21) відображає нові, ще не оброблені заявки зі статусом “Not Updated Yet”.

Нове бронювання

Showing 1 to 1 of 1 entries

№	Номер бронювання	Ім'я	Електронна пошта	Контактний номер	Статус	Інформація
1	674978043	admin admin	admin@gmail.com	380000000	Not Updated Yet	🔗
2	596650723	admin admin	admin@gmail.com	777777777	Not Updated Yet	🔗
3	262262224	Igor Kovalchuk	kovalchuk@gmail.com	685555555	Not Updated Yet	🔗
4	617917007	Igor Kovalchuk	igor111@gmail.com	981111111	Not Updated Yet	🔗

Рисунок 3.21 – Сторінка “Нове бронювання” панелі адміністратора

Сторінка “Затвердженні бронювання” (рис. 3.22) відображає всі заявки, які були затверджені адміністратором.

Затвердженні бронювання

Showing 1 to 1 of 1 entries

№	Номер бронювання	Ім'я	Електронна пошта	Контактний номер	Статус	Інформація
1	399845592	Andrii Test	testAndrii@test3.com	962056184	Approved	🔗
2	487839487	Андрій Леськів	test451212@test3.com	962056184	Approved	🔗
3	656672881	Igor Kovalchuk	igor@gmail.com	685569083	Approved	🔗
4	197326758	admin admin	admin@gmail.com	1010101010	Approved	🔗
5	596486626	admin10 admin10	admin@gmail.com	1444215454	Approved	🔗

Рисунок 3.22 – Сторінка “Затвердженні бронювання” адміністративної панелі

Сторінка “Скасовані бронювання” (рис. 3.23) відображає заявки, які були скасовані.

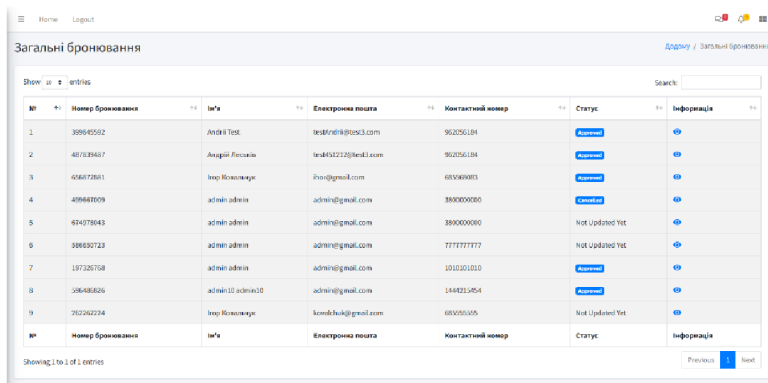
Скасовані бронювання

Showing 1 to 1 of 1 entries

№	Номер бронювання	Ім'я	Електронна пошта	Контактний номер	Статус	Інформація
1	499667009	admin admin	admin@gmail.com	380000000	Cancelled	🔗

Рисунок 3.23 – Сторінка "Скасовані бронювання" панелі адміністратора

Сторінка “Загальні бронювання” (рис. 3.24) надає повний перелік усіх заявок на бронювання, незалежно від їхнього статусу.



The screenshot shows a web interface titled "Загальні бронювання" (General Reservations). It features a table with the following columns: "№" (ID), "Номер бронювання" (Reservation Number), "Ім'я" (Name), "Електронна пошта" (Email), "Контактний номер" (Contact Number), "Статус" (Status), and "Інформація" (Information). The table contains 9 rows of data. The status column includes buttons like "Активувати" (Activate) and "Немає оновлення" (No updates yet). A search bar is located at the top right, and pagination controls are at the bottom.

№	Номер бронювання	Ім'я	Електронна пошта	Контактний номер	Статус	Інформація
1	39964552	Admin Test	test@redhat.com	962056184	Активувати	🔍
2	407834617	Андрій Песчані	test41221@red3.com	962056184	Активувати	🔍
3	656672861	Ігор Коваленко	Pro@gmail.com	693069883	Активувати	🔍
4	489648009	admin admin	admin@gmail.com	880000080	Активувати	🔍
5	614978443	admin admin	admin@gmail.com	880000080	Немає оновлення	🔍
6	88693723	admin admin	admin@gmail.com	777777777	Немає оновлення	🔍
7	197326708	admin admin	admin@gmail.com	1010101010	Активувати	🔍
8	28488626	admin10 admin10	admin@gmail.com	1444221454	Активувати	🔍
9	26262224	Ігор Коваленко	kovalehuk@gmail.com	693050005	Немає оновлення	🔍

Рисунок 3.24 – Сторінка “Загальні бронювання” панелі адміністратора

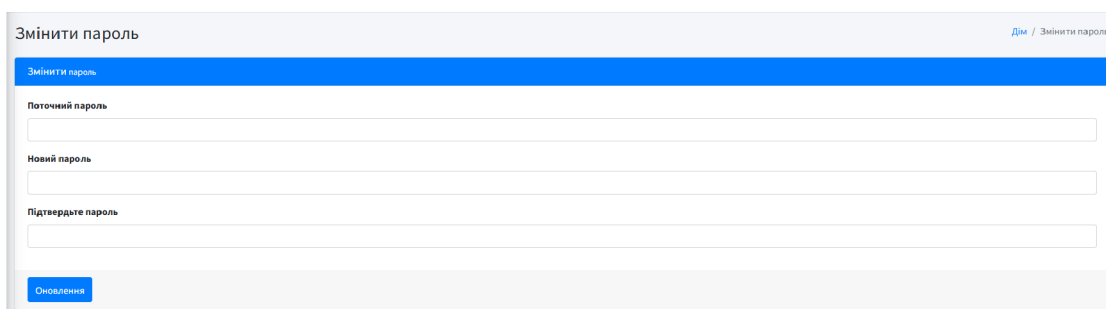
Адміністратор також може керувати своїм профілем та змінювати пароль. Сторінка “Профіль” (рис. 3.25) відображає основну інформацію про адміністратора (ім’я, логін, email, контактний номер, дату реєстрації) з можливістю її оновлення.



The screenshot shows a web interface titled "Профіль" (Profile). It displays a form for editing user information. The fields are: "Ім'я адміністратора" (Admin Name) with value "Admin", "Ім'я користувача" (Username) with value "admin", "Електронна пошта" (Email) with value "admin@gmail.com", "Контактний номер" (Contact Number) with value "989888888", and "Дата реєстрації адміністратора" (Admin Registration Date) with value "2025-03-22 09:24:25". A blue "Оновити" (Update) button is at the bottom left.

Рисунок 3.25 – Сторінка “Профіль” адміністратора

Сторінка “Змінити пароль” (рис. 3.26) дозволяє адміністратору змінити свій пароль, ввівши поточний та новий паролі.



The screenshot shows a web interface titled "Змінити пароль" (Change Password). It contains three input fields: "Поточний пароль" (Current Password), "Новий пароль" (New Password), and "Підтвердьте пароль" (Confirm Password). A blue "Оновлення" (Update) button is located at the bottom left.

Рисунок 3.26 – Сторінка “Змінити пароль” адміністратора

3.3 Вимоги до апаратного та програмного забезпечення

Для коректного функціонування вебсистеми діяльності спортивного клубу комплексу LEOLAND, як на етапі розробки, так і під час подальшої експлуатації, необхідне відповідне апаратне та програмне забезпечення наведеними у таблиці 3.1.

Таблиця 3.1. Вимоги до комп'ютера

Процесор (CPU)	2+ ядра, 2.5+ ГГц
Оперативна пам'ять (RAM)	4 ГБ або більше
Постійна пам'ять (SSD/NVMe/HDD)	50 ГБ SSD
Операційна система	Windows Server 2017+ / Ubuntu Server 20.04 LTS / Debian 9+ Windows 10/11, macOS або дистрибутив Linux
Програмне забезпечення	Apache HTTP Server (2.4+) PHP (8.0+) MySQL (5.7+) або MariaDB (10.2+) Веб-браузер: Будь-який сучасний (Chrome, Firefox, Edge, Safari) (рекомендовано останніх версій з підтримкою HTML5, CSS3, JavaScript)

Для користування системою в якості клієнта необхідно мати доступ до Інтернету на будь-якому пристрої з встановленим веббраузером.

3.4 Розгортання вебсистеми

Розгортання вебсистеми є ключовим етапом, що передуює її повноцінній експлуатації. Цей процес передбачає перенесення розробленого програмного забезпечення з локального середовища розробки на продуктивний вебсервер, доступний для кінцевих користувачів через мережу Інтернет. Основні кроки процесу розгортання включають:

1. Підготовка серверного середовища – Встановлення та налаштування операційної системи (наприклад, Linux), вебсервера (Apache або Nginx),

інтерпретатора PHP та СКБД MySQL на обраному хостингу або виділеному сервері.

2. Перенесення файлів проєкту – Копіювання всіх файлів вебсистеми (PHP-скрипти, HTML-файли, CSS-стилі, JavaScript-скрипти, зображення) на вебсервер у відповідні директорії.

3. Імпорт бази даних – Створення БД на продуктивному сервері та імпорт до неї структури та початкових даних з локального середовища розробки.

4. Налаштування підключення до БД – Оновлення PHP файлів вебсистеми для коректного підключення до БД на продуктивному сервері (зміна хоста, імені користувача, пароля БД).

5. Конфігурація вебсервера – Налаштування віртуальних хостів та інших параметрів вебсервера для коректної обробки запитів до вебсистеми.

6. Тестування після розгортання – Проведення фінального тестування функціоналу системи вже на продуктивному середовищі для переконання у її коректній роботі.

ВИСНОВКИ

У даній дипломній роботі було успішно розроблено та реалізовано вебсистему для автоматизації діяльності спортивного клубу комплексу LEOLAND. На основі детального аналізу предметної області та виявлених проблем, пов'язаних з неефективним ручним управлінням клієнтами, гуртками, розкладом та заявками, було обґрунтовано актуальність створення такої системи.

В рамках роботи було спроектовано тривірневу архітектуру вебсистеми, розроблено реляційну базу даних на базі MySQL та деталізовано функціональні модулі для взаємодії як клієнтів (реєстрація, авторизація, перегляд гуртків та розкладу, запис на заняття, особистий кабінет), так і адміністраторів (управління гуртками, схвалення/відхилення заявок).

Проведене тестування підтвердило функціональність, надійність та безпеку розробленої вебсистеми. Впровадження цієї системи дозволить спортивному клубу комплексу LEOLAND значно оптимізувати внутрішні бізнес-процеси, підвищити ефективність взаємодії з клієнтами, покращити доступність інформації та забезпечити керівництво інструментами для прийняття обґрунтованих управлінських рішень.

Таким чином, поставлена мета дипломної роботи була досягнута, а розроблена вебсистема є готовим рішенням для підвищення конкурентоспроможності спортивному клубу комплексу LEOLAND на ринку спортивних послуг.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Мова програмування JavaScript [Електронний ресурс] – Режим доступу до ресурсу: <http://www.znannya.org/?view=javascript/>(дата звернення 28.05.2025). – Назва з екрана.
2. HTML конструювання [Електронний ресурс] – Режим доступу до ресурсу: <http://htmlbook.in.ua/>(дата звернення 28.05.2025). – Назва з екрана.
3. Що таке -take-html CSS [Електронний ресурс] – Режим доступу до ресурсу: https://css.in.ua/article/shcho_10/(дата звернення 28.05.2025). – Назва з екрана.
4. Гольцман В. MySQL 5.0 [Електронний ресурс] – Режим доступу до ресурсу: <https://citforum.at.ua/MySQL.pdf>(дата звернення 28.05.2025). – Назва з екрана.
5. Веб-сервери [Електронний ресурс] – Режим доступу до ресурсу: <https://helpiks.org/8-59867.html/>(дата звернення 28.05.2025). – Назва з екрана.
6. Вступ в PHP [Електронний ресурс] – Режим доступу до ресурсу: <http://www.victka.net/node/14/>(дата звернення 28.05.2025). – Назва з екрана.
7. Мова програмування JavaScript [Електронний ресурс] – Режим доступу до ресурсу: <http://www.znannya.org/?view=javascript/>(дата звернення 28.05.2025). – Назва з екрана.
8. jQuery – Вступна [Електронний ресурс] – Режим доступу до ресурсу: <http://softwares.net.ua/jquery-vstupna/>(дата звернення 28.05.2025). – Назва з екрана.
9. Хаген Граф. Створення веб-сайтів [Електронний ресурс] – Режим доступу до ресурсу: <http://www.webtec.com.ua/uk/articles/index/view/2011-05-05/web-site/>(дата звернення 28.05.2025). – Назва з екрана.
10. Керівництво розробника CSS. [Електронний ресурс] – Режим доступу до ресурсу: https://developer.mozilla.org/ru/docs/Web/Guide/CSS/Getting_started/What_is_CSS/(дата звернення 28.05.2025). – Назва з екрана.
11. Довідник HTML [Електронний ресурс] – Режим доступу до ресурсу: <http://htmlbook.ua/html/>(дата звернення 28.05.2025). – Назва з екрана.
12. Mysql [Електронний ресурс] – Режим доступу до ресурсу: http://www.znannya.org/?view=PHP_SUBD_mysql(дата звернення 28.05.2025). – Назва з екрана.

13. Основи JavaScript [Електронний ресурс] – Режим доступу до ресурсу: https://developer.mozilla.org/ru/docs/Web/Guide/CSS/Getting_started/What_is_CSSh (дата звернення 28.05.2025). – Назва з екрана.

14. МЕТОДОЛОГИЯ IDEF. [Електронний ресурс] – Режим доступу до ресурсу: https://sites.google.com/site/anisimovkhv/learning/pris/lecture/tema6/tema6_2(дата звернення 28.05.2025). – Назва з екрана.

15. Що таке PHP? [Електронний ресурс] – Режим доступу до ресурсу: <https://www.php.net/manual/ua/intro-what-is.php>_(дата звернення 28.05.2025). – Назва з екрана.

16. Створення Web-сайта на базі WordPress CMS [Електронний ресурс] – Режим доступу до ресурсу: <https://www.ibm.com/developerworks/ru/library/oswordpress/index.html>/(дата звернення 28.05.2025). – Назва з екрана.

17. What is Database? What is SQL? [Електронний ресурс] – Режим доступу до ресурсу: <https://www.guru99.com/introduction-to-database-sql.html>/(дата звернення 28.05.2025). – Назва з екрана.

18. Що таке PHP? [Електронний ресурс] – Режим доступу до ресурсу: <http://www.znannya.org/?view=javascript>/(дата звернення 28.05.2025). – Назва з екрана.

ДОДАТОК

```
<?php session_start();

//error_reporting(0);

include('includes/dbconnection.php');

?>

<!DOCTYPE html>

<html lang="zxx">

<head>

    <title>Yoga Classes Registration System | Home Page</title>

    <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/nice-select.css"/>

    <link rel="stylesheet" href="css/magnific-popup.css"/>

    <link rel="stylesheet" href="css/slicknav.min.css"/>

    <link rel="stylesheet" href="css/animate.css"/>

    <!-- Main Stylesheets -->

    <link rel="stylesheet" href="css/style.css"/>

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.4/css/all.min.css">

</head>

<body>

    <!-- Header Section -->

    <?php include_once('includes/header.php'); ?>
```

```

<!-- Header Section end -->

<!-- Hero Section -->

<section class="hero-section">

  <div class="arrow-buttom">

  </div>

  <div class="hero-slider owl-carousel">

    <div class="hs-item">

      <div class="hs-style-1 text-center">

      </div>

    </div>

    <div class="hs-item">

      <div class="hs-style-2">

        <div class="container-fluid h-100">

          <div class="row h-100">

            <div class="col-lg-6 h-100 d-none d-lg-flex align-items-xl-end align-items-lg-center">

              <div class="hs-img">

              </div>

            </div>

            <div class="col-lg-6 d-flex align-items-center">

              <div class="hs-text-warp">

                <div class="hs-text">

                  <h2>Get slim and toned with yoga</h2>

                  <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor

```

incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis
nos-trud exercitation ullamco laboris nisi ut aliquip ex ea commodo
consequat. </p>

</div>

</div>

</div>

</div>

</div>

</div>

</div>

<div class="hs-item">

<div class="hs-style-3 text-center">

<div class="container">

<div class="hs-text">

<h2>Reduce your stress</h2>

<p>Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque
laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi.</p>

</div>

</div>

<div class="hs-img">

</div>

</div>

</div>

</div>

</section>

```
<!-- Hero Section end -->
```

```
<!-- About Section -->
```

```
<section class="about-section spad">
```

```
<div class="container">
```

```
<div class="section-title text-center">
```

```

```

```
<h2>Welcome to Yoga Classes</h2>
```

```
<p>Yoga is a journey to finding that balance, both on and off the mat. A yoga practice stretches and strengthens our bodies; it teaches us to mindfully move and connect with our breath, which leads us to be mindful and present in our daily tasks. It teaches us the freedom to be still and connects us.</p>
```

```
</div>
```

```
<div class="row">
```

```
<div class="col-lg-6">
```

```
<div class="about-img">
```

```

```

```
</div>
```

```
</div>
```

```
<div class="col-lg-6">
```

```
<div class="about-item">
```

```
<div class="ai-icon">
```

```

```

```
</div>
```

```
<div class="ai-text">
```

```
<h4>Full Rejuvenation</h4>
```

```
<p>Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</p>
```

```
</div>
```

```
</div>
```

```
<div class="about-item">
```

```
<div class="ai-icon">
```

```

```

```
</div>
```

```
<div class="ai-text">
```

```
<h4>Extension of Spring</h4>
```

```
<p>Sed ut perspiciatis unde omnis iste natus error sit voluptatem a cusantium doloremque.</p>
```

```
</div>
```

```
</div>
```

```
<div class="about-item">
```

```
<div class="ai-icon">
```

```

```

```
</div>
```

```
<div class="ai-text">
```

```
<h4>Against Aging</h4>
```

```
<p>Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit.</p>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</div>
```

```
</section>
```

```
<!-- About Section end -->
```

```

<!-- Classes Section -->

<section class="classes-section spad">

  <div class="container">

    <div class="section-title text-center">

      <h2>Наші секції</h2>

      <p>Практикуйте спорт, щоб удосконалити фізичну красу та повніше насолоджуйтесь життям</p>

    </div>

    <div class="category_box">

      <?php

      $sql = "SELECT * from tblcategory";

      $query = $dbh->prepare($sql);

      $query->execute();

      $results = $query->fetchAll(PDO::FETCH_OBJ);

      $cnt = 1;

      if ($query->rowCount() > 0) {

        foreach ($results as $row) {?>

          <div class="classes-item">

            <div class="ci-img">

            </div>

            <div class="ci-text">

              <h4>

                <a href="category-details.php?viewid=<?php echo $row->id; ?>"><?php echo $row->categoryName; ?></a>

```

```

        </h4>

    </div>

</div>

<?php $cnt = $cnt + 1;

    }

} ?>

</div>

<div class="classes-slider owl-carousel">

    <?php

        $sql = "SELECT * from tblcategory";

        $query = $dbh->prepare($sql);

        $query->execute();

        $results = $query->fetchAll(PDO::FETCH_OBJ);

        $cnt = 1;

        if ($query->rowCount() > 0) {

            foreach ($results as $row) {?>

                <div class="classes-item">

                    <div class="ci-img">

                    </div>

                    <div class="ci-text">

                        <h4>

                            <a href="category-details.php?viewid=<?php echo $row->id; ?>"><?php echo $row-
>categoryName; ?></a>

                        </h4>

                    </div>

                </div>

```

```

<!--          <div class="ci-bottom">-->

<!--          <a href="book-yoga-sesion.php?cid=--><?php //echo $row->ID; ?><!--" class="site-btn sb-
gradient">book-->

<!--          now</a>-->

<!--          </div>-->

          </div>

          <?php $cnt = $cnt + 1;

          }

          }?>

        </div>

    </div>

</section>

<!-- Classes Section end -->

<!-- Footer Section -->

<?php include_once('includes/footer.php'); ?>

<!-- Footer Section end -->

<div class="back-to-top"></div>

<!--===== Javascrpts & JQuery =====>

<script src="js/vendor/jquery-3.2.1.min.js"></script>

<script src="js/bootstrap.min.js"></script>

<script src="js/jquery.slicknav.min.js"></script>

```

```
<script src="js/owl.carousel.min.js"></script>

<script src="js/jquery.nice-select.min.js"></script>

<script src="js/jquery-ui.min.js"></script>

<script src="js/jquery.magnific-popup.min.js"></script>

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

</body>

</html>
```

```
<?php

session_start();

error_reporting(E_ALL);

ini_set('display_errors', 0);

include('includes/dbconnection.php');

if (strlen($_SESSION['ycrsaid'] == 0)) {

    header('location:logout.php');

} else {

    if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST['submit'])) {

        try {

            $category_id = intval($_POST['category_id']);

            $sub_category_name = trim($_POST['sub_category_name']);

            $description = trim($_POST['desc']);

            $ytrainer = trim($_POST['ytrainer']);

            $price = floatval($_POST['price']);

            $field_size = trim($_POST['field_size']);

            $max_people = intval($_POST['max_people']);

            $schedule = trim($_POST['schedule']);
```

```

$booking_phone = trim($_POST['booking_phone']);

$image_names = [];

if (isset($_FILES["images"]) && $_FILES["images"]["error"][0] == 0) {

    $upload_dir = __DIR__ . "/images/images_category/";

    if (!is_dir($upload_dir)) {

        mkdir($upload_dir, 0777, true);

    }

    $allowed_extensions = ['jpg', 'jpeg', 'png', 'gif'];

    foreach ($_FILES["images"]["name"] as $key => $filename) {

        $extension = strtolower(pathinfo($filename, PATHINFO_EXTENSION));

        if (!in_array($extension, $allowed_extensions)) {

            die("Помилка: недозволеній тип файлу!");

        }

        if ($_FILES["images"]["size"][$key] > 2 * 1024 * 1024) {

            die("Помилка: один із файлів занадто великий!");

        }

        $image_name = md5($filename) . time() . $key . '.' . $extension;

        $upload_path = $upload_dir . $image_name;

        if (!move_uploaded_file($_FILES["images"]["tmp_name"][$key], $upload_path)) {

            die("Помилка при завантаженні файлу $filename!");

        }

        $image_names[] = $image_name;

    }

} else {

    die("Помилка: Файли не вибрано!");
}

```

```

}

$image_names_str = implode(",", $image_names);

$sql = "INSERT INTO tblclasses (category_id, sub_category_name, sub_category_images,
sub_category_description,
        sub_category_trainer, price, field_size, max_people, booking_phone, schedule)
        VALUES (:category_id, :sub_category_name, :image, :description, :ytrainer, :price, :field_size,
:max_people,
        :booking_phone, :schedule)";

$query = $dbh->prepare($sql);

$query->bindParam(':category_id', $category_id, PDO::PARAM_INT);

$query->bindParam(':image', $image_names_str, PDO::PARAM_STR);

$query->bindParam(':description', $description, PDO::PARAM_STR);

$query->bindParam(':sub_category_name', $sub_category_name, PDO::PARAM_STR);

$query->bindParam(':ytrainer', $ytrainer, PDO::PARAM_STR);

$query->bindParam(':price', $price, PDO::PARAM_STR);

$query->bindParam(':field_size', $field_size, PDO::PARAM_STR);

$query->bindParam(':max_people', $max_people, PDO::PARAM_INT);

$query->bindParam(':booking_phone', $booking_phone, PDO::PARAM_STR);

$query->bindParam(':schedule', $schedule, PDO::PARAM_STR);

if ($query->execute()) {

    echo "☐ Секцію успішно додано!";

    echo "<script>window.location.href ='classes.php'</script>";

} else {

    $error = $query->errorInfo();

    echo "☐ Помилка при додаванні секції: " . $error[2];

}

```

```

    } catch (PDOException $e) {

        die("❑ Помилка бази даних: " . $e->getMessage());

    }

}

?>

<!DOCTYPE html>

<html>

<head>

    <title>Category Classes Registration System | Classes</title>

    <!-- Font Awesome -->

    <link rel="stylesheet" href="plugins/fontawesome-free/css/all.min.css">

    <!-- Ionicons -->

    <link rel="stylesheet" href="https://code.ionicframework.com/ionicons/2.0.1/css/ionicons.min.css">

    <!-- Theme style -->

    <link rel="stylesheet" href="dist/css/adminlte.min.css">

    <!-- Google Font: Source Sans Pro -->

    <link href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:300,400,400i,700" rel="stylesheet">

    <script src="http://js.nicedit.com/nicEdit-latest.js" type="text/javascript"></script>

    <script type="text/javascript">bkLib.onDomLoaded(nicEditors.allTextAreas);</script>

</head>

<body class="hold-transition sidebar-mini">

<div class="wrapper">

    <!-- Navbar -->

    <?php include_once('includes/header.php'); ?>

    <!-- /.navbar -->

```

```

<!-- Main Sidebar Container -->

<?php include_once('includes/sidebar.php'); ?>

<!-- Content Wrapper. Contains page content -->

<div class="content-wrapper">

    <!-- Content Header (Page header) -->

    <section class="content-header">

        <div class="container-fluid">

            <div class="row mb-2">

                <div class="col-sm-6">

                    <h1>Add Section</h1>

                </div>

                <div class="col-sm-6">

                    <ol class="breadcrumb float-sm-right">

                        <li class="breadcrumb-item"><a href="dashboard.php">Home</a></li>

                        <li class="breadcrumb-item active">Section</li>

                    </ol>

                </div>

            </div>

        </div><!-- /.container-fluid -->

    </section>

<!-- Main content -->

<section class="content">

    <div class="container-fluid">

        <div class="row">

            <!-- left column -->

            <div class="col-md-12">

```

```

<!-- general form elements -->

<div class="card card-primary">

  <div class="card-header">

    <h3 class="card-title">Classes</h3>

  </div>

  <!-- /.card-header -->

  <!-- form start -->

  <form role="form" method="post" enctype="multipart/form-data">

    <div class="card-body">

      <div class="form-group">

        <label for="category">Select Category</label>

        <select class="form-control select2" name="category_id" required>

          <option value="">Choose a Category</option>

          <?php

            $stmt = $dbh->query("SELECT id, categoryName FROM tblcategory ORDER BY
categoryName ASC");

            while ($row = $stmt->fetch(PDO::FETCH_ASSOC)) {

              echo '<option value="' . $row['id'] . '">' . htmlspecialchars($row['categoryName']) .
'</option>';

            }

          ?>

        </select>

      </div>

      <div class="form-group">

        <label for="ytrainer">Category Name</label>

        <input type="text" class="form-control" name="sub_category_name" required>

```

```
</div>
```

```
<div class="form-group">
```

```
  <label for="categoryImage">Category Images</label>
```

```
  <input type="file" name="images[]" multiple>
```

```
</div>
```

```
<div class="form-group">
```

```
  <label for="desc">Description</label>
```

```
  <textarea class="form-control" name="desc"></textarea>
```

```
</div>
```

```
<div class="form-group">
```

```
  <label for="ytrainer">Category Trainer</label>
```

```
  <input type="text" class="form-control" name="ytrainer">
```

```
</div>
```

```
<div class="form-group">
```

```
  <label for="price">Section Price</label>
```

```
  <input type="number" class="form-control" name="price" required>
```

```
</div>
```

```
<div class="form-group">
```

```
  <label for="field_size">Field Size</label>
```

```
  <textarea class="form-control" name="field_size"></textarea>
```

```
</div>
```

```
<div class="form-group">
```

```
  <label for="max_people">Максимально людей</label>
```

```
  <input type="text" class="form-control" name="max_people">
```

```
</div>
```

```

<div class="form-group">
    <label for="schedule">Days & Time</label>
    <textarea class="form-control" name="schedule"></textarea>
</div>
<div class="form-group">
    <label for="booking_phone">Booking Phone</label>
    <input type="text" class="form-control" name="booking_phone" required>
</div>
<div class="card-footer">
    <button type="submit" class="btn btn-primary" name="submit">Submit</button>
</div>
</div>
</form>
</div>
</div>
<!-- /.col (left) -->
<!-- /.col (right) -->
</div>
<!-- /.row -->
</div><!-- /.container-fluid -->
</section>
<!-- /.content -->
</div>
<!-- /.content-wrapper -->
<?php include_once('includes/footer.php'); ?>

```

```

<!-- Control Sidebar -->

<aside class="control-sidebar control-sidebar-dark">

    <!-- Control sidebar content goes here -->

</aside>

<!-- /.control-sidebar -->

</div>

<!-- ./wrapper -->

<!-- jQuery -->

<script src="plugins/jquery/jquery.min.js"></script>

<!-- Bootstrap 4 -->

<script src="plugins/bootstrap/js/bootstrap.bundle.min.js"></script>

<!-- bs-custom-file-input -->

<script src="plugins/bs-custom-file-input/bs-custom-file-input.min.js"></script>

<!-- AdminLTE App -->

<script src="dist/js/adminlte.min.js"></script>

<!-- AdminLTE for demo purposes -->

<script src="dist/js/demo.js"></script>

<script type="text/javascript">

    $(document).ready(function () {

        bsCustomFileInput.init();

    });

</script>

</body>

</html><?php } ?>

```