МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВСП «ЕКОНОМІКО-ПРАВНИЧИЙ ФАХОВИЙ КОЛЕДЖ
ЗАПОРІЗЬКОГО НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ»

Циклова комісія математичних дисциплін та інформаційних технологій

КВАЛІФІКАЦІЙНА РОБОТА

на тему: «РОЗРОБКА ІНФОРМАЦІЙНОЇ СИСТЕМИ З ПРОДАЖУ КВИТКІВ НА СПОРТИВНІ ПОДІЇ»

Виконав: здобувач освіти 4 курсу, групи К121-22
Спеціальність 121 Інженерія
програмного забезпечення
(шифр і спеціальність)
Михайло ЛЕБЕДЕНКО
(ім'я та ПРІЗВИЩЕ)
Керівник Юлія БОРИСОВСЬКА
(ім'я та ПРІЗВИЩЕ)
Рецензент доцент каф. програмної інженерії ЗНУ, к.т.н., доцент Юлія ЛИМАРЕНКО
(посада, вчене звання, науковий ступінь, ім'я та ПРІЗВИЩЕ)

Запоріжжя
2026

ЗАПОРІЗЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ

ВСП «Економіко-правничий фаховий коледж ЗНУ»

Освітньо–кваліфікаційний ступінь фаховий молодший бакалавр

Спеціальність 121 – Інженерія програмного забезпечення
(шифр і назва)

ЗАТВЕРДЖУЮ

Голова циклової комісії математичних дисциплін та інформаційних технологій

_____ Т.М. Смолянкова
(підпис)

“ 05 ” червня 2026 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ СТУДЕНТУ

ЛЕБЕДЕНКУ Михайлу Володимировичу
(ПРІЗВИЩЕ, ім’я та по– батькові)

1. Тема роботи «Розробка інформаційної системи з продажу квитків на спортивні події»

Керівник роботи БОРИСОВСЬКА Юлія Олександрівна
(науковий ступінь, вчене звання, прізвище, ім’я, по батькові)

Затверджені наказом ВСП ЕПФК ЗНУ від «01» грудня 2025 р. № 1728-с

2. Строк подання студентом роботи 05.06.2026

3. Вихідні дані до роботи 1. Постановка задачі.
2. Перелік літератури.

4. Зміст розрахунково–пояснювальної записки (перелік питань, які потрібно розробити)
1. Сучасні інформаційні системи
2. Розробка проєкту інформаційної системи
3. Програмна реалізація інформаційної системи

5. Перелік графічного матеріалу (з точним зазначенням обов’язкових креслень)
презентація до захисту

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		Завдання видав	Завдання прийняв

7. Дата видачі завдання 01.12.2025

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів кваліфікаційної роботи	Строк виконання етапів роботи	Примітка
1.	Розробка плану роботи:		
	постановка задачі	Грудень 2025	виконано
2.	Збір вихідних даних, обробка методичних		
	та теоретичних джерел	Січень 2026	виконано
3.	Розробка першого розділу:		
	огляд інформаційних систем та технологій для їхнього створення	Лютий 2026	виконано
4.	Розробка другого розділу:		
	розробка проєкту інформаційної системи та вибір технологій	Квітень 2026	виконано
5.	Розробка третього розділу:		
	розробка застосунку	Травень 2026	виконано
6.	Оформлення і нормоконтроль кваліфікаційної роботи та перевірка на плагіат	Червень 2026	виконано
7.	Захист кваліфікаційної роботи	16.06.2026	виконано

Здобувач освіти _____ Михайло ЛЕБЕДЕНКО
(підпис) (ім’я ПРІЗВИЩЕ)

Керівник роботи _____ Юлія БОРИСОВСЬКА
(підпис) (ім’я ПРІЗВИЩЕ)

Нормоконтроль пройдено

Нормоконтролер _____ Юлія БОРИСОВСЬКА
(підпис) (ім’я ПРІЗВИЩЕ)

РЕФЕРАТ

Кваліфікаційна робота: 50 сторінок, 12 рисунків, 3 таблиці, 14 джерел.

Об’єктом дослідження є процеси автоматизації операційного планування спортивних заходів, централізованого обліку вільних місць та електронної дистрибуції квитків у сучасному вебсередовищі.

Предметом дослідження є вебтехнології, архітектурні рішення, інструменти веброзробки та програмні алгоритми захисту даних при створенні інформаційних систем для реалізації квитків та менеджменту подій.

Метою роботи є проєктування, програмна реалізація та верифікація багатофункціональної інформаційної системи, що забезпечує повний автоматизований цикл продажу квитків, управління контентом заходів та облік фінансових транзакцій із високим рівнем безпеки.

Кваліфікаційну роботу виконано із використанням трирівневої клієнт-серверної архітектури. У процесі розробки застосовано методи системного аналізу предметної області, об'єктно-орієнтованого проєктування та моделювання реляційних баз даних. Серверна логіка вебзастосунку реалізована на базі мови програмування PHP, а рівень збереження інформації забезпечено реляційною СУБД MySQL. Графічний інтерфейс користувача побудовано на основі стандартів HTML5 та CSS3 із інтеграцією клієнтських сценаріїв JavaScript для підвищення інтерактивності екранних форм.

У ході виконання кваліфікаційної роботи розроблено цілісну веборієнтовану інформаційну систему. Спроєктовано оптимальну структуру бази даних sport_events_db, що містить взаємопов'язані таблиці для обліку користувачів, подій, локацій трибун, квитків та фінансових транзакцій. Програмно реалізовано механізми безпечної автентифікації на основі криптографічного хешування BCRYPT та захисту від SQL-ін’єкцій через обов'язкове використання підготовлених запитів. Створено динамічну афішу заходів, модуль онлайн-замовлення з транзакційною логікою на стороні сервера

та захищену адміністративну панель для керування контентом сайту в режимі реального часу.

Розроблена інформаційна система враховує вимоги безпеки міжнародних платіжних стандартів PCI DSS та має ізольовану модульну структуру, що дозволяє легко інтегрувати реальні платіжні шлюзи. Програмний продукт може бути успішно використаний локальними спортивними федераціями, комплексами та клубами для повної цифровізації квиткового господарства без залучення сторонніх комерційних сервісів-агрегаторів.

ІНФОРМАЦІЙНА СИСТЕМА, ВЕБЗАСТОСУНОК, PHP, MYSQL, КЛІЄНТ-СЕРВЕРНА АРХІТЕКТУРА, КВИТКОВА ДИСТРИБУЦІЯ, БЕЗПЕКА ДАНИХ, PREPARED STATEMENTS, BCRYPT

SUMMARY

The qualification work consists of 50 pages, 12 figures, 3 tables, and 14 sources.

The object of research is the automation processes of operational planning of sports events, centralized accounting of available seats, and electronic ticket distribution in a modern web environment.

The subject of research consists of web technologies, architectural solutions, web development tools, and software data protection algorithms used in creating information systems for ticket sales and event management.

The goal of the work is the design, software implementation, and verification of a multi-functional information system that provides a full automated cycle of ticket sales, event content management, and financial transaction accounting with a high level of security.

The qualification work is executed using a three-tier client-server architecture. In the process of development, methods of system analysis of the subject area, object-oriented design, and relational database modeling were applied. The backend logic of the web application is implemented based on the PHP programming language, and the data storage layer is provided by the MySQL relational DBMS. The graphical user interface is built based on HTML5 and CSS3 standards with the integration of client-side JavaScript scenarios to increase the interactivity of screen forms.

In the course of the qualification work, a cohesive web-oriented information system was developed. The optimal structure of the sport_events_db database was designed, containing interconnected tables for accounting users, events, stand locations, tickets, and financial transactions. Mechanisms for secure authentication based on BCRYPT cryptographic hashing and protection against SQL injections through the mandatory use of prepared statements were implemented. A dynamic event poster, an online ordering module with server-side transactional logic, and a secure administrative panel for real-time site content management were created.

The developed information system takes into account the security requirements of international payment standards PCI DSS and has an isolated modular structure, which allows for easy integration of real payment gateways. The software product can be successfully used by local sports federations, complexes, and clubs for full digitalization of ticket management without involving third-party commercial aggregator services.

KEYWORDS: INFORMATION SYSTEM, WEB APPLICATION, PHP, MYSQL, CLIENT-SERVER ARCHITECTURE, TICKET DISTRIBUTION, DATA SECURITY, PREPARED STATEMENTS, BCRYPT

ЗМІСТ

Завдання на кваліфікаційну роботу студенту 2
Реферат 4
Summary 6
Вступ 9
1 Огляд предметної області 11
1.1 Аналіз сучасних систем продажу квитків 11
1.2 Аналіз предметної області 15
1.3 Аналіз технологій розробки 16
1.4 Постановка задачі 17
2 Проєктування інформаційної системи 19
2.1 Архітектура системи 19
2.2 Діаграма прецедентів інформаційної системи 20
2.3 Об'єктно-орієнтоване проєктування та діаграма класів системи 22
2.4 Моделювання динамічної взаємодії та діаграма послідовності 23
2.5 Проєктування фізичної архітектури та діаграма розгортання 25
2.6 Проєктування реляційної бази даних та ER-модель 26
2.7 Проєктування графічного інтерфейсу користувача 28
3 Програмна реалізація 32
3.1 Опис середовища розробки та реалізація серверної логіки 32
3.2 Реалізація роботи з БД 33
3.3 Реалізація інтерфейсу 34
3.4 Забезпечення безпеки 40
3.5 Тестування системи 42
3.6 Перспективи розвитку 44
Висновки 48
Перелік використаних джерел 50

ВСТУП

Цифрова трансформація спортивної індустрії та стрімкий розвиток веборієнтованих технологій вимагають впровадження ефективних автоматизованих інструментів для організації взаємодії між організаторами масових заходів та вболівальниками. Сучасні системи електронної дистрибуції мають забезпечувати не лише високу швидкість обробки інформаційних потоків, а й забезпечувати транзакційну цілісність, захищеність персональних даних, інтуїтивність графічного інтерфейсу та прозорість операційного обліку. Розробка та розгортання власної автономної інформаційної системи дозволяє локальним спортивним організаціям і клубам суттєво оптимізувати операційні витрати, повністю уникаючи високих комісійних зборів з боку сторонніх сервісів-агрегаторів, а також отримати повний прямий контроль над базами даних, аналітикою та статистикою продажів у режимі реального часу.

Метою кваліфікаційної роботи є проєктування, програмна реалізація та верифікація інформаційної системи для планування спортивних подій та автоматизації процесів реалізації електронних квитків. Об’єктом дослідження виступає процес організації спортивних заходів, менеджменту контенту та дистрибуції квитків у сучасному вебсередовищі. Предметом дослідження є методи, архітектурні рішення, програмні засоби та інструменти розробки динамічних вебзастосунків на основі трирівневої клієнт-серверної моделі.

Для досягнення поставленої мети в межах роботи було визначено та вирішено комплекс послідовних інженерних завдань. Першочергово було проведено ґрунтовний аналіз предметної області та існуючих ринкових аналогів систем продажу квитків, що дозволило сформувати вимоги до майбутнього продукту. На основі цих вимог було спроєктовано логічну й фізичну структуру реляційної бази даних із забезпеченням умов третьої нормальної форми, цілісності зв'язків та збереження інформації. Наступним кроком стало обґрунтування технологічного стеку та реалізація серверної логіки системи

засобами мови PHP, що дозволило розробити модулі автентифікації користувачів, динамічної фільтрації подій та атомарного оформлення замовлень. Також було реалізовано адаптивний графічний інтерфейс користувача та спеціалізований модуль адміністрування для виконання CRUD-операцій над контентом. Кінцевим завданням став аналіз безпеки розробленого програмного продукту та комплексне функціональне тестування системи на відповідність вимогам надійності.

У процесі програмної реалізації інформаційної системи було використано сучасний стек технологій, що включає мову програмування PHP, СКБД MySQL у локальному середовищі інструментального пакету XAMPP, а також стандартизовані мови розмітки контенту та каскадних стилів HTML5 і CSS3 [9]. Архітектура розробленого вебзастосунку базується на обов'язковому використанні механізмів підготовлених запитів та транзакційного управління на рівні бази даних, що гарантує надійний захист від виникнення SQL-ін’єкцій та атомарність фінансових операцій користувачів. Результатом роботи є готова до експлуатації функціональна інформаційна система, яка забезпечує повний цикл обслуговування клієнта від перегляду актуальної спортивної афіші до отримання персонального електронного квитка з унікальним ідентифікатором транзакції.

1 ОГЛЯД ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналіз сучасних систем продажу квитків

На сьогодні сфера онлайн-маркетингу та дистрибуції електронних квитків на масові спортивні й культурні заходи представлена автоматизованими платформами, які задають високі міжнародні стандарти якості, безпеки та транзакційної швидкодії. Для проведення детального аналізу архітектурних переваг, функціональних можливостей та виявлення системних недоліків існуючих рішень було обрано три ключові платформи, що домінують на світовому та вітчизняному ринках: глобальний сервіс Ticketmaster, а також провідні українські квиткові оператори Concert.ua та Karabas.

Глобальним технологічним лідером у досліджуваній сфері є американська інформаційна система Ticketmaster, яка адмініструє масштабні потоки даних та обслуговує мільйонні аудиторії в реальному часі [1]. Головною архітектурною перевагою даної платформи є інтеграція високонавантажених модулів інтерактивного тривимірного моделювання арен, що дозволяє користувачеві перед покупкою оцінити реальний вигляд на спортивний майданчик безпосередньо з обраного посадкового місця. Також система впроваджує інтелектуальні алгоритми верифікації фанатів для боротьби з вторинним ринком перекупників. Загальний інтерфейс користувача та структура відкритої вітрини подій сервісу Ticketmaster представлені на рисунку 1.1.



Рисунок 1.1 – Інтерфейс головної сторінки системи Ticketmaster

Основними технологічними перевагами Ticketmaster є масштабованість, наявність інтерактивних планів залів та високий рівень криптографічного захисту транзакцій. Проте суттєвими недоліками системи є надзвичайна складність її програмного ядра, орієнтація суто на глобальні корпорації, а також критично висока комісійна ставка (сервісний збір), яка суттєво здорожує фінальну вартість квитків для кінцевого вболівальника.

Серед вітчизняних рішень значну частку українського ринку охоплює інформаційна система Concert.ua. Дана платформа орієнтована на забезпечення максимальної простоти взаємодії із кінцевим споживачем та відрізняється високою швидкістю адаптації інтерфейсу під мобільні пристрої. Система має гнучкий модуль фільтрації подій за категоріями, датами та локаціями, а також підтримує швидку інтеграцію з поширеними національними платіжними шлюзами [2]. Візуальне компонування картки заходу та модулів вибору місць на платформі Concert.ua продемонстровано на рисунку 1.2.

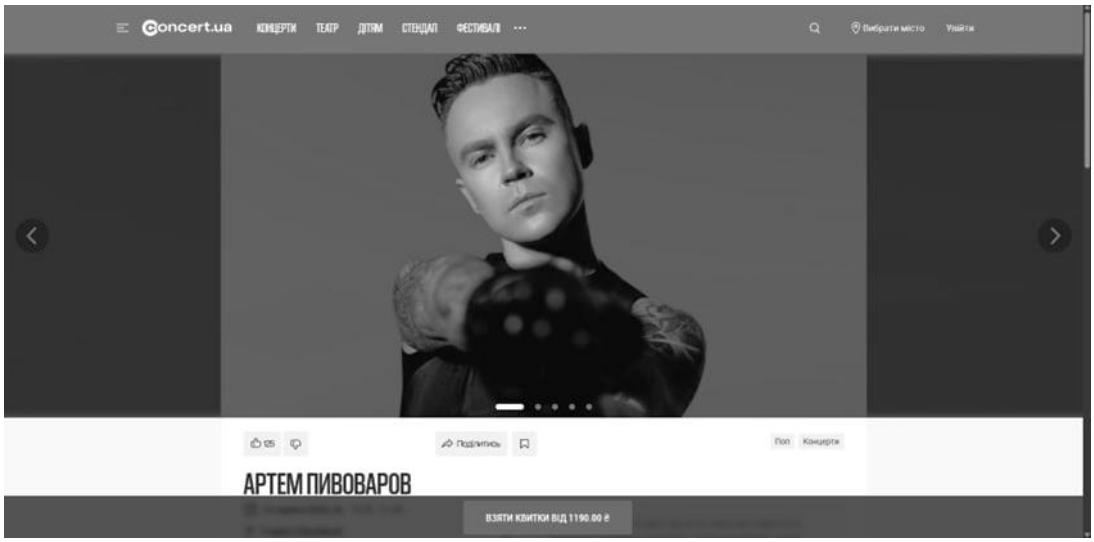


Рисунок 1.2 – Візуальне представлення картки події на платформі Concert.ua

До позитивних сторін сервісу Concert.ua варто віднести лаконічний UX/UI-дизайн, стабільну роботу системи автентифікації та зручну інтеграцію з цифровими гаманцями. Втім, головним мінусом для локальних спортивних організацій є її закрита архітектура комерційного агрегатора. Платформа стягує великий відсоток за обслуговування кожного заходу, не надає організаторам повного прямого доступу до сирих баз даних клієнтів та аналітичних масивів для внутрішнього маркетингу.

Ще одним репрезентативним представником національного сегменту є сервіс Karabas, який спеціалізується на дистрибуції електронних квитків через розгалужену мережу інтерактивних віджетів та власних мобільних застосунків. Платформа використовує централізовану реляційну базу даних для оперативного моніторингу наявності вільних місць, що мінімізує затримки при масових паралельних запитах користувачів [3]. Приклад організації інтерфейсу списку подій у системі Karabas наведено на рисунку 1.3.

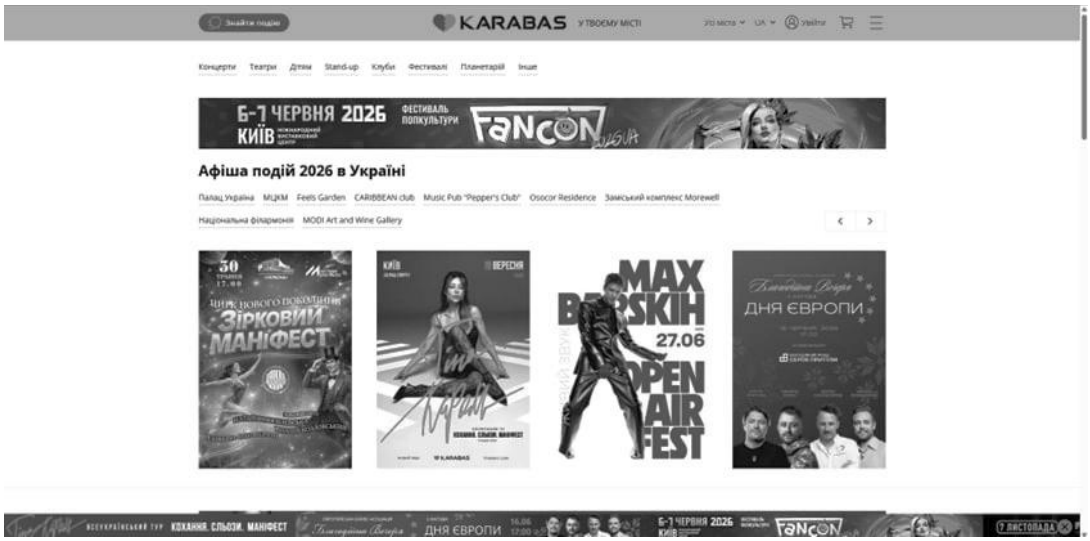


Рисунок 1.3 – Структура інтерфейсу перегляду подій у системі Karabas

Основним плюсом системи Karabas є стійкість до пікових навантажень та наявність розвиненого API для синхронізації з партнерськими сервісами. Головні мінуси аналогічні до попереднього конкурента: висока фінансова залежність організатора від посередника, жорсткі рамки кастомізації інтерфейсу під специфічні потреби конкретного спортивного клубу та відсутність автономного локального модуля адміністрування для внутрішнього персоналу стадіону.

Проведений порівняльний аналіз існуючих ринкових рішень дозволив визначити базовий набір вимог до власної розроблюваної системи. Оскільки розглянуті сервіси функціонують виключно як зовнішні комерційні агрегатори з високими комісійними зборами та закритими базами даних, постає гостра інженерна актуальність створення автономної, локальної інформаційної системи. Розробка власного програмного продукту на базі клієнт-серверної архітектури дозволить локальним спортивним федераціям та клубам повністю ліквідувати витрати на сторонніх посередників, забезпечити прямий зв'язок із вболівальниками через персоналізований вебінтерфейс, реалізувати гнучкі механізми транзакційного продажу квитків без ризику овербукінгу, а також отримати повний і захищений контроль над операційною статистикою та аналітикою відвідуваності об'єкта в режимі реального часу.

1.2 Аналіз предметної області

Специфіка досліджуваної предметної області характеризується високою динамічністю інформаційних потоків, необхідністю обробки значних масивів даних у пікові періоди перед початком заходів та підвищеними вимогами до надійності й безпеки фінансових транзакцій. Об’єктом автоматизації у кваліфікаційній роботі виступають процеси операційного планування спортивних заходів, централізованого обліку доступності вільних місць на трибунах, а також електронної дистрибуції та верифікації квитків. Основними суб’єктами та об'єктами взаємодії в межах проєктованої системи є організатори заходів (адміністратори), кінцеві споживачі (вболівальники), спортивні події різного масштабу (матчі, турніри), локації їх проведення (стадіони, спортивні комплекси з обмеженою місткістю трибун), а також документи суворого обліку у вигляді електронних квитків та пов'язані з ними транзакції.

Проведений аналіз поточної діяльності та операційних процесів на базі локальних спортивних майданчиків виявив низку критичних недоліків та системних проблем, які суттєво знижують загальну ефективність організації змагань. Першою вагомою проблемою є низька швидкість обслуговування відвідувачів та обмеженість каналів реалізації. Традиційний продаж квитків через фізичні каси або лімітовані за часом точки продажу створює значні черги в передматчеві дні, що критично знижує лояльність вболівальників та штучно обмежує географію потенційних покупців. Другим деструктивним фактором виступає високий ризик виникнення помилок через людський фактор при ручному обліку. Ведення обліку вільних і зайнятих місць у паперовому вигляді або за допомогою локальних електронних таблиць на кшталт MS Excel часто призводить до виникнення небезпечних аномалій даних, зокрема дублювання місць і овербукінгу, втрати інформації про транзакції та критичного ускладнення процесу швидкої верифікації квитків на вході до стадіону.

Окремим критичним недоліком є повна відсутність інструментів оперативної аналітики та моніторингу завантаженості в режимі реального часу.

За умов відсутності спеціалізованої автоматизованої системи менеджменту організаторам вкрай важко відстежувати динаміку та темпи продажів, що унеможливлює гнучке управління ціноутворенням, оперативне прогнозування заповнюваності трибун та ефективне планування маркетингових кампаній. Таким чином, розробка та впровадження проєктованої веборієнтованої інформаційної системи дозволить повністю цифровізувати зазначені процеси, централізувати збереження інформації в реляційній базі даних, мінімізувати операційні витрати організації та забезпечити цілодобовий захищений доступ користувачів до сервісу через зручний інтерфейс.

1.3 Аналіз технологій розробки

Для реалізації інформаційної системи планування спортивних подій та продажу квитків було проведено аналіз сучасних технологій створення динамічних вебзастосунків. Вибір технологічного стеку здійснювався на основі критеріїв швидкості обробки транзакцій, надійності збереження даних, простоти масштабування архітектури та доступності інструментів для розгортання серверної частини.

Основним інструментом розробки серверної логіки було обрано скриптову мову програмування PHP 8.2. Дане рішення зумовлене її глибокою спеціалізацією на веброзробці, високою швидкістю виконання сценаріїв та наявністю ефективних вбудованих модулів для безпечної взаємодії з реляційними базами даних. Використання PHP дозволяє гнучко реалізувати механізми сесій користувачів, валідацію вхідних параметрів та динамічне формування контенту сторінок залежно від наявних у системі спортивних заходів.

Для організації збереження інформації, керування обліковими записами користувачів, обліку доступних місць на стадіонах та фіксації квиткових транзакцій було обрано реляційну систему керування базами даних MySQL.

Вибір цієї СКБД обґрунтований її високою продуктивністю при використанні складних SQL-запитів із об'єднанням таблиць, підтримкою транзакційних механізмів для запобігання аномалій даних (зокрема, овербукінгу квитків) та повною сумісністю із мовою програмування PHP через розширення mysqli.

Розробка клієнтської частини інформаційної системи базується на класичному комерційному стеку вебтехнологій, що включає стандарти HTML5, CSS3 та мову сценаріїв JavaScript. Мова розмітки HTML5 використана для проєктування логічної структури вебсторінок, розмітки інтерактивних форм бронювання та виведення списків подій. Каскадні таблиці стилів CSS3 застосовані для реалізації адаптивного інтерфейсу користувача за допомогою технології Flexbox, що забезпечує коректне відображення сайту на мобільних пристроях та персональних комп'ютерах. Мова JavaScript інтегрована для підвищення динамічності інтерфейсу, зокрема для виконання первинної перевірки коректності введення даних користувачем на стороні клієнта перед відправкою HTTP-запитів на сервер.

Локальне інструментальне середовище для розробки, налагодження та комплексного тестування модулів інформаційної системи побудоване на базі кросплатформного програмного пакету XAMPP під керуванням вебсервера Apache. Даний вибір дозволив об'єднати інтерпретатор мови PHP, вебсервер та сервер баз даних у єдиний локальний контур розробки, а використання інтегрованої графічної утиліти phpMyAdmin суттєво спростило процеси адміністрування, проєктування схем таблиць та первинного наповнення бази даних тестовими записами.

1.4 Постановка задачі

На основі проведеного комплексного аналізу предметної області, специфіки бізнес-процесів дистрибуції квитків та існуючих технологічних рішень, метою даної кваліфікаційної роботи є розробка та впровадження

веборієнтованої інформаційної системи для автоматизації процесів планування спортивних заходів та реалізації електронних квитків.

Для досягнення поставленої мети та розгортання працездатного програмного продукту в межах дослідження необхідно вирішити комплекс взаємопов'язаних інженерних, архітектурних та програмних завдань. Першочергово потрібно спроєктувати оптимальну логічну й фізичну структуру реляційної бази даних sport_events_db, забезпечивши її відповідність вимогам третьої нормальної форми, цілісність зв'язків та надійне збереження даних про користувачів, події, локації, квитки й фінансові транзакції. Наступним кроком є розробка архітектури серверної логіки вебзастосунку на базі мови програмування PHP для надійного забезпечення таких ключових процесів, як багаторівнева автентифікація користувачів, динамічна фільтрація афіші та атомарне оформлення замовлень за допомогою транзакційних механізмів.

Також необхідно реалізувати адаптивний графічний вебінтерфейс користувача з використанням стандартів HTML5, CSS3 та елементів клієнтських сценаріїв JavaScript, гарантувавши коректне кросбраузерне відображення екранних форм на різних типах пристроїв, включаючи мобільні телефони, планшети та персональні комп'ютери. Для менеджерів системи вимагається створити інтегрований модуль адміністрування, який дозволить здійснювати повний спектр CRUD-операцій над контентом подій і локацій, а також проводити моніторинг фінансових транзакцій у режимі реального часу. Фінальним завданням є забезпечення високого рівня безпеки й стабільності роботи платформи шляхом впровадження програмних механізмів захисту від типових вебвразливостей, зокрема SQL-ін’єкцій, та проведення верифікації працездатності системи за допомогою функціонального тестування.

Очікуваним результатом роботи є готова до повноцінної експлуатації автономна інформаційна система, що дозволяє вболівальникам самостійно купувати квитки, а організаторам – ефективно керувати спортивними заходами без залучення сторонніх сервісів-агрегаторів.

2 ПРОЄКТУВАННЯ ІНФОРМАЦІЙНОЇ СИСТЕМИ

2.1 Архітектура системи

Для програмної реалізації інформаційної системи планування спортивних подій та продажу квитків було обрано класичну клієнт-серверну архітектуру, що базується на трирівневій моделі. Зазначений архітектурний підхід дозволяє логічно та функціонально відокремити графічний інтерфейс користувача від безпосередньої бізнес-логіки обробки запитів та рівня фізичного зберігання інформації, що забезпечує високу стабільність, захищеність і потенціал для подальшого масштабування програмного продукту [4].

Першим конструктивним рівнем виступає рівень представлення даних, який виконує роль клієнтської частини вебзастосунку, з якою безпосередньо взаємодіє відвідувач через екранні форми веббраузера. Цей компонент реалізовано за допомогою стандартного кросплатформного стеку вебтехнологій, де мова розмітки HTML5 використовується для структурування контенту, каскадні таблиці стилів CSS3 забезпечують візуальну стилізацію інтерфейсу та його адаптивність під мобільні пристрої, а мова програмування JavaScript відповідає за інтерактивність користувацького досвіду, включаючи первинну валідацію полів та обробку динамічних подій без перезавантаження сторінок [5].

Другим рівнем системи є рівень логіки, реалізований на базі серверної мови програмування PHP. Це центральний обчислювальний компонент системи, що функціонує на стороні вебсервера Apache у складі інструментального пакету локальної розробки XAMPP. Серверна частина повноцінно забезпечує приймання та обробку вхідних HTTP-запитів від клієнта, реалізацію бізнес-логіки перевірки доступності місць, розрахунку вартості квитків, керування суперглобальними масивами сесій користувачів для авторизації й розмежування прав доступу, а також організовує безпечну транзакційну взаємодію з базою даних [6].

Фінальним елементом архітектури виступає рівень даних, представлений реляційною системою керування базами даних MySQL. Цей рівень відповідає за надійне та структуроване зберігання, захист та оперативну вибірку інформації за запитами серверних PHP-скриптів [7]. На рівні бази даних забезпечується персистентність усієї системної інформації, зокрема даних про облікові записи користувачів, спортивні події, характеристики локацій, поточний статус квитків та історію фінансових транзакцій. Комунікація між рівнем логіки та рівнем даних здійснюється за допомогою оптимізованих запитів мови SQL. Використання такої моделі дозволяє централізовано керувати інформаційними потоками та гарантує стабільний доступ до системи з будь-якого пристрою.

2.2 Діаграма прецедентів інформаційної системи

Для формалізації функціональних вимог до розроблюваної інформаційної системи, визначення її меж та візуалізації процесів взаємодії користувачів із ключовими модулями було використано мову графічного моделювання UML [8]. Основним інструментом на етапі концептуального аналізу бізнес-логіки платформи стала діаграма варіантів використання, яка дозволяє абстрагуватися від програмної реалізації та чітко зафіксувати ролі користувачів і доступний їм інструментарій. Розроблена UML-діаграма прецедентів, що описує повний спектр взаємодії з вебресурсом, представлена нижче на рисунку 2.1.

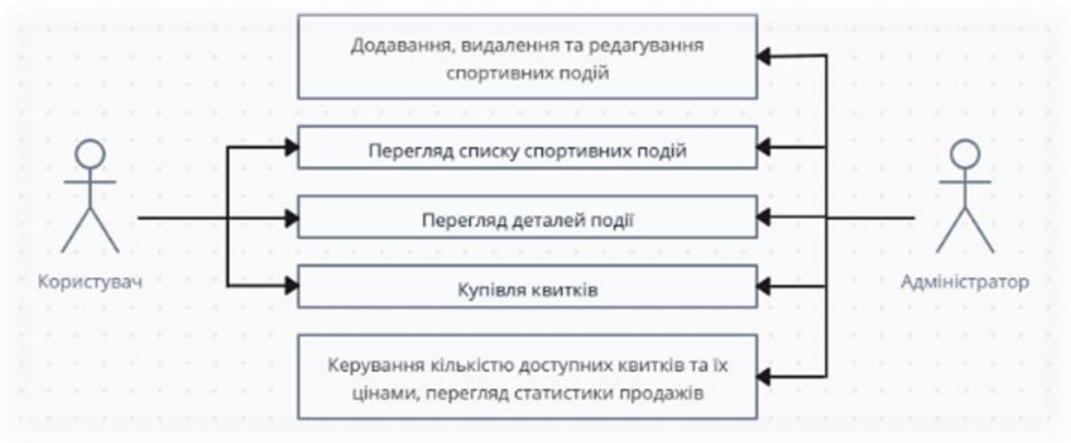


Рисунок 2.1 – UML-діаграма варіантів використання

Основними дійовими особами, тобто акторами, у межах спроєктованої архітектури виступають користувач та адміністратор системи. Актор користувача уособлює звичайного відвідувача вебресурсу, основною метою якого є ознайомлення з поточною спортивною афішею, пошук і фільтрація заходів, а також безпосереднє придбання електронних квитків. Актор адміністратора представляє привілейованого користувача з розширеними правами доступу, який відповідає за підтримання актуальності інформаційних масивів, менеджмент контенту та моніторинг операційної діяльності сервісу. Для глибшого аналізу логіки функціонування платформи доцільно деталізувати основні бізнес-сценарії взаємодії зазначених акторів із системою.

Ключовим прецедентом клієнтської частини є сценарій купівлі квитків, який ініціюється користувачем після вибору конкретного спортивного заходу з динамічного списку вітрини подій. Система забезпечує відображення детальних параметрів івенту, включаючи назву, дату, час, адресу локації та поточний залишок вільних місць. Після підтвердження операції користувачем платформа виконує автоматичну перевірку стану авторизації, ініціює безпечне оформлення замовлення, динамічно перераховує баланс доступних квитків та генерує персональний документ із унікальним ідентифікатором транзакції.

Сценарій управління подіями реалізується актором адміністратора через спеціалізований захищений інтерфейс панелі керування. Даний прецедент охоплює повний спектр стандартних операцій менеджменту даних, зокрема створення нових записів про змагання, редагування існуючих параметрів, встановлення лімітів місткості стадіонів та видалення неактуальних івентів. При внесенні нової події адміністратор взаємодіє з інтегрованим довідником спортивних майданчиків, а система автоматично оновлює контент клієнтської частини, роблячи захід доступним для онлайн-бронювання. Крім того, адміністратору доступний прецедент аналізу статистики продажів, який дозволяє відстежувати фінансові потоки та оцінювати завантаженість локацій, що гарантує високу ефективність адміністрування комерційної системи.

2.3 Об'єктно-орієнтоване проєктування та діаграма класів системи

Наступним архітектурним етапом розробки проєкту інформаційної системи є перехід від концептуальних прецедентів взаємодії до об'єктно-орієнтованого моделювання внутрішньої структури програмного продукту. Для візуалізації статичної структури системи, визначення ключових програмних сутностей, їхніх внутрішніх атрибутів, методів обробки даних та типів логічного зв'язку між ними було побудовано UML-діаграму класів [4]. Цей інструмент дозволяє відобразити архітектуру коду як сукупність взаємопов'язаних модулів, що забезпечує гнучкість розширення функціоналу платформи без руйнування її ядра. Розроблена діаграма класів інформаційної системи представлена нижче на рисунку 2.2.

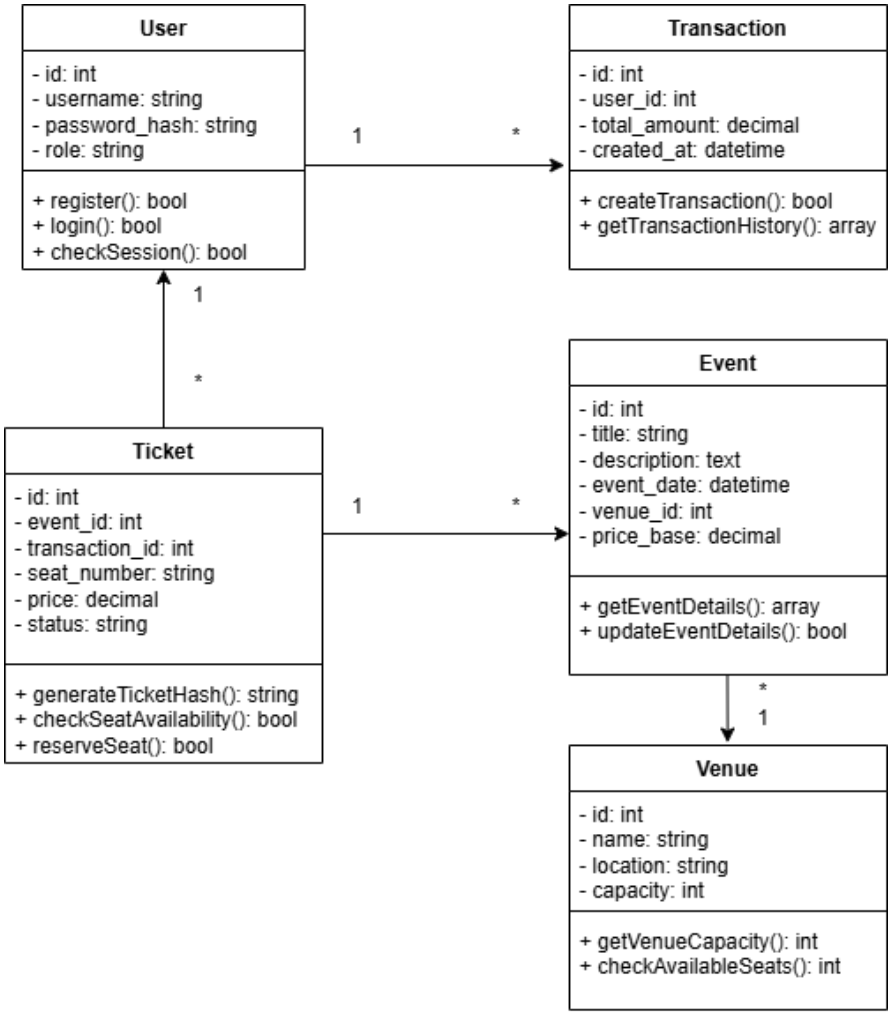


Рисунок 2.2 – UML-діаграма класів інформаційної системи

Спроєктована структура включає п'ять базових класів, які координують ключові бізнес-процеси платформи. Клас User інкапсулює в собі механізми автентифікації й реєстрації відвідувачів, а також зберігає сесійні параметри прав доступу ролей. Взаємодію з каталогом спортивних заходів координує клас Event, який містить атрибути часових обмежень та ідентифікатори прив'язки до локацій. Просторові характеристики стадіонів та параметри максимальної місткості трибун описуються методами класу Venue. Для безпосереднього керування процесами квиткової дистрибуції, генерації унікальних хеш-ідентифікаторів та моніторингу станів зайнятості місць призначено функціонал класу Ticket. Системний облік фінансових операцій, фіксацію купівельних дій у часовому вимірі та забезпечення атомарності операцій покладено на методи класу Transaction. Наведені класи пов'язані між собою логічними асоціативними зв'язками різної кратності, що гарантує коректну структурованість коду на етапі програмної реалізації backend-частини застосунку.

2.4 Моделювання динамічної взаємодії та діаграма послідовності

Для детального аналізу та документування динамічної поведінки інформаційної системи під час виконання найбільш критичних бізнес-процесів було застосовано UML-діаграму послідовності [8]. Побудова даної моделі дозволяє візуалізувати часову послідовність обміну інформаційними повідомленнями між ключовими об'єктами системи, починаючи від ініціації дії користувачем на клієнтській частині й закінчуючи остаточною фіксацією змін на рівні реляційної бази даних. Як репрезентативний сценарій для моделювання було обрано процес оформлення замовлення та придбання електронного квитка авторизованим користувачем. Графічне представлення розробленої діаграми послідовності наведено на рисунку 2.3.

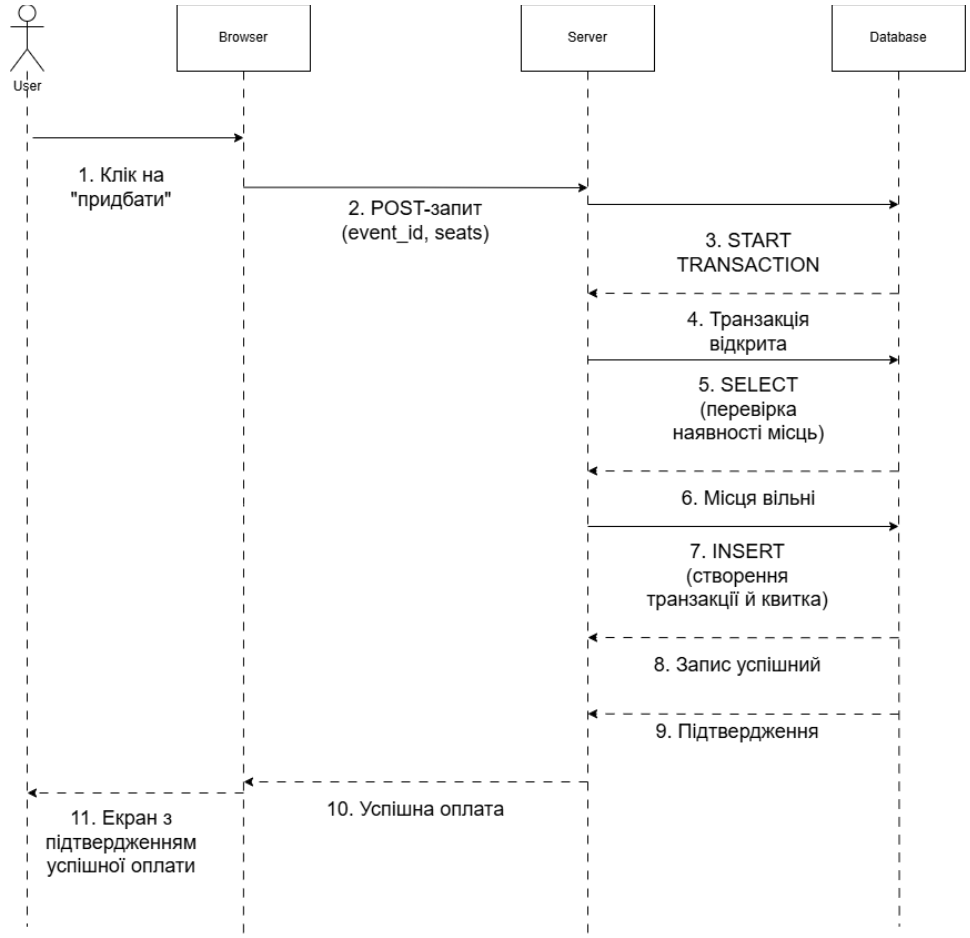


Рисунок 2.3 – UML-діаграма послідовності сценарію купівлі квитків

У межах побудованої динамічної моделі зафіксовано взаємодію між чотирма ключовими компонентами: актором користувача, клієнтським інтерфейсом веббраузера, серверним обробником PHP та реляційним сховищем MySQL. Процес ініціюється відправкою HTTP-запиту з параметрами форми покупки квитків до серверного скрипта. Серверний модуль виконує первинну сесійну валідацію, після чого звертається до рівня бази даних для перевірки реальної наявності вільних місць на обраний захід. У разі позитивної відповіді СКБД, сервер ініціює ініціалізацію захищеної транзакції, вносить відповідні записи до логу фінансових операцій та змінює статус доступності квитків. Фінальним етапом є повернення сервером успішного статусу відповіді до інтерфейсу браузера, який динамічно генерує для користувача екранну форму підтвердження оплати з унікальним номером транзакції, що підтверджує повну узгодженість динамічних процесів платформи.

2.5 Проєктування фізичної архітектури та діаграма розгортання

Завершальним етапом архітектурного проєктування платформи є визначення топології апаратних засобів та особливостей розміщення програмних компонентів у реальному мережевому середовищі. Для вирішення цього завдання було побудовано UML-діаграму розгортання, яка відображає фізичний розподіл обчислювальних ресурсів, конфігурацію серверних вузлів та мережеві протоколи, що забезпечують надійну комунікацію між ними [9]. Розроблена діаграма розгортання інформаційної системи представлена на рисунку 2.4.

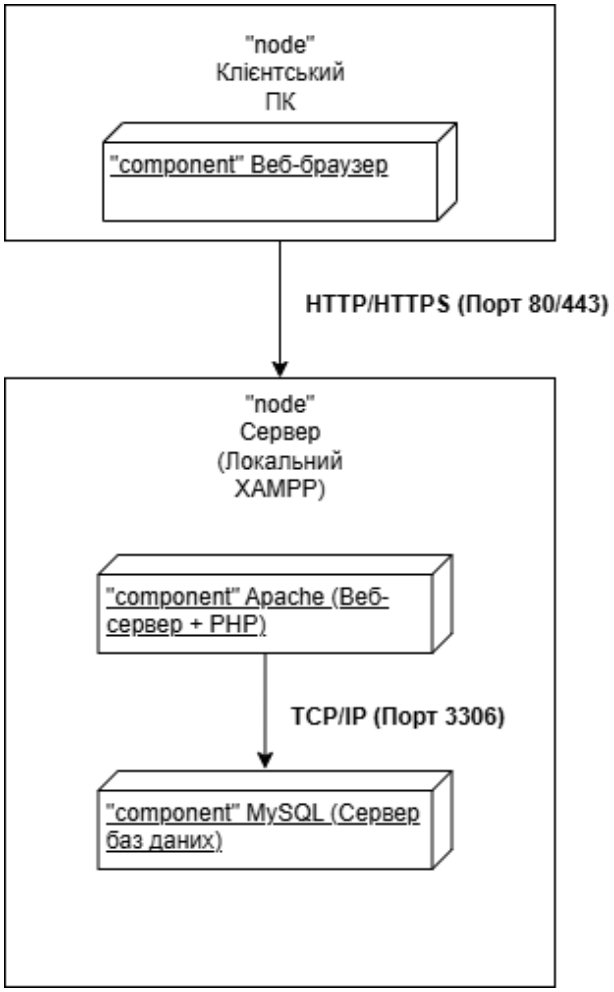


Рисунок 2.4 – UML-діаграма розгортання інформаційної системи

Спроєктована фізична архітектура базується на взаємодії двох основних обчислювальних вузлів (Nodes): клієнтського термінала та локального сервера

розробки. Вузол клієнтського пристрою уособлює персональний комп'ютер або смартфон вболівальника, де в якості програмного артефакту функціонує сучасний веббраузер, який відповідає за рендеринг адаптивних сторінок HTML5/CSS3 та виконання клієнтських скриптів JavaScript. Комунікаційний зв'язок між клієнтом та сервером організовано через мережевий протокол передачі даних HTTP/HTTPS. Вузол сервера розгортання базується на кросплатформному програмному пакеті XAMPP, у складі якого ізольовано функціонують два критичні компоненти: вебсервер Apache, відповідальний за інтерпретацію серверного PHP-коду й маршрутизацію запитів, та сервер баз даних MySQL, який здійснює персистентне збереження та захист інформаційних масивів, що гарантує стабільну експлуатацію вебзастосунку.

2.6 Проєктування реляційної бази даних та ER-модель

Для забезпечення стабільного функціонування, високої швидкодії та залізобетонної транзакційної цілісності розроблюваного вебзастосунку було спроєктовано реляційну базу даних sport_events_db. Розробка оптимальної архітектури та логічної структури сховища даних базується на дескриптивному методі «сутність–зв'язок» за допомогою побудови концептуальної ER-моделі (Entity-Relationship). Це архітектурне рішення дозволило наочно візуалізувати майбутні масиви інформації, чітко визначити ключові сутності предметної області, їхні унікальні змінні атрибути, а також логічні типи зв'язків між ними перед безпосереднім етапом фізичного створення таблиць у СКБД MySQL. Графічне відображення розробленої реляційної ER-моделі із детальними каскадними зв'язками представлено нижче на рисунку 2.5.

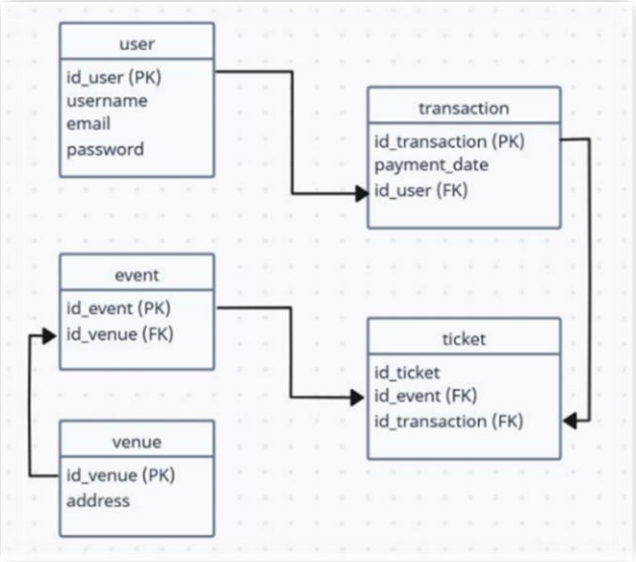


Рисунок 2.5 – ER-діаграма бази даних sport_events_db

Побудована концептуальна ER-модель даних дозволяє ефективно структурувати інформаційні потоки сервісу та повністю усунути аномалії надлишковості або дублювання параметрів за рахунок послідовного проведення процедури декомпозиції й нормалізації таблиць до умов третьої нормальної форми (3NF). Для забезпечення логічної цілісності даних на рівні реляційних зв'язків було чітко визначено кратність взаємодії між сутностями. Характер та логічна специфіка встановлених зв'язків між ключовими компонентами розробленої бази даних деталізовані нижче у таблиці 2.1.

Таблиця 2.1 – Зв’язки між сутностями

Сутність (Entity)	Зв’язана з (Related to)	Тип зв’язку (Relationship Type)
User	Transaction	1 → N (один до багатьох)
Transaction	Ticket	1 → N (один до багатьох)
Ticket	Event	N → 1 (багато до одного)
Event	Venue (Місце)	N → 1 (багато до одного)

Проєктована фізична архітектура сховища складається з п'яти взаємопов'язаних таблиць, кожна з яких відповідає за ізольовану сутність

предметної області. Таблиця users призначена для надійного збереження персональних даних зареєстрованих відвідувачів та адміністраторів платформи, включаючи унікальний логін, рівень прав доступу та паролі, захищені алгоритмами криптографічного хешування. Інформація про спортивні заходи, їхній розширений текстовий опис, категорію змагань та часові обмеження акумулюється в таблиці events. Довідник місць проведення матчів та турнірів реалізовано через таблицю venues, яка фіксує назви стадіонів, точні географічні адреси та показники максимальної місткості трибун. Сутність конкретних генерованих квитків, їх поточний операційний статус (вільно чи продано) та каскадна прив'язка до відповідних івентів описується параметрами таблиці tickets. Фіксація фінансових операцій здійснюється в таблиці transactions, яка зв'язує унікальний ідентифікатор авторизованого користувача із придбаними ним квитками в межах однієї сесії покупки, що гарантує збереження повної історії замовлень.

2.7 Проєктування графічного інтерфейсу користувача

На етапі проєктування клієнтської частини вебзастосунку було детально розроблено загальну структуру, логіку компонування та візуальну ієрархію екранних форм інтерфейсу, яка забезпечує зручну взаємодію користувачів із серверною логікою інформаційної системи. Основний інженерний акцент при проєктуванні графічного середовища було зроблено на забезпеченні принципу динамічного формування контенту сторінок засобами серверних PHP-скриптів на основі оперативної вибірки даних із СУКБД MySQL. Для верифікації структури сайту перед кодингом було створено низькодеталізовані чорно-білі каркаси (Wireframes), які згодом послужили основою для фінальної збірки клієнтської частини.

Головна сторінка системи (файл index.php) виступає основною точкою входу та призначена для первинного ознайомлення відвідувачів із наявним

асортиментом заходів. Проєктування інтерфейсу цієї форми передбачає реалізацію динамічного виведення трьох найближчих спортивних подій за допомогою SQL-запиту з оператором обмеження ліміту, що дозволяє акцентувати увагу вболівальників на найбільш актуальних івентах. Логіка відображення інтерфейсу автоматично підбирає відповідні графічні обкладинки залежно від значення категорії у базі даних. Навігаційна панель містить умовні конструкції перевірки сесії користувача для динамічної зміни елементів керування. Розроблений структурний каркас головної сторінки у порівнянні з її фінальною програмною реалізацією представлено на рисунку 2.6.

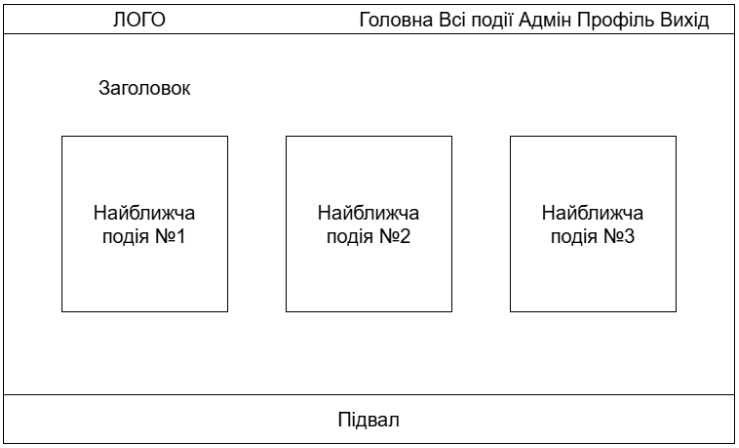


Рисунок 2.6 – Структурний каркас та реалізація головної сторінки index.php

Сторінка детальної інформації про захід та модуля оплати (файл event_details.php) є найбільш функціональним елементом клієнтського інтерфейсу, де реалізовано повний цикл оформлення замовлення. Система контролю доступу на рівні інтерфейсу блокує можливість фінансових операцій для неавторизованих гостей, перенаправляючи їх на сторінку логування. Інтерактивна форма замовлення дозволяє власноруч обирати кількість квитків у встановленому діапазоні, а для покращення користувацького досвіду інтегровано клієнтський скрипт на мові JavaScript, який візуалізує процес обробки платіжної транзакції за допомогою анімації завантаження на кнопці відправки форми. Схематичний макет інтерфейсу форми оформлення покупки квитків та відображення картки події в реальному браузері наведено на рисунку 2.7.

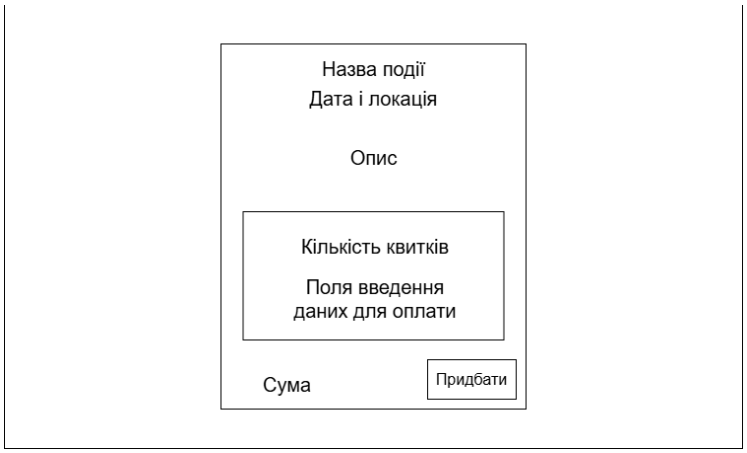


Рисунок 2.7 – Схематичний макет та інтерфейс форми замовлення квитків

Проєктування інтерфейсу адміністративної панелі (файл admin.php) виконувалося з урахуванням жорсткого обмеження прав доступу суто для привілейованих облікових записів. Робочий простір містить інструментарій табличного відображення наявних івентів із функціями швидкого видалення записів, що ініціюють відповідні операції на рівні бази даних. Форма додавання нових подій використовує випадаючі списки, які динамічно заповнюються актуальними назвами локацій із пов'язаних таблиць, що повністю мінімізує ризик введення некоректних або неіснуючих ідентифікаторів персоналом стадіону. Компонування елементів панелі керування контентом на рівні прототипу та готового робочого вікна адміністратора продемонстровано на рисунку 2.8.

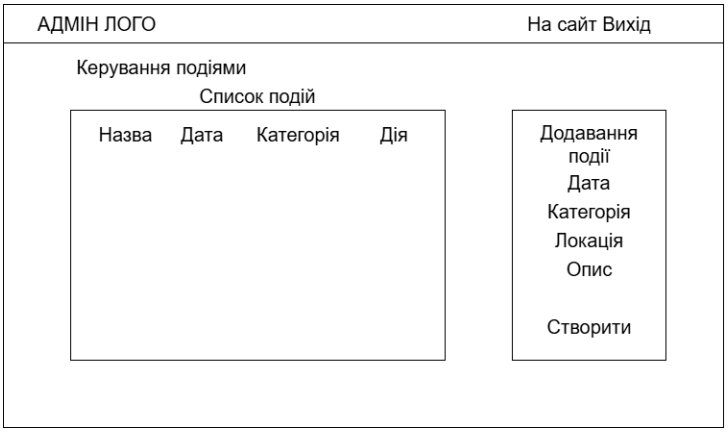


Рисунок 2.8 – Прототип та інтерфейс робочого простору адміністратора системи

Проєктування форми відображення повного переліку заходів (файл events.php) орієнтоване на забезпечення розширених можливостей пошуку та зручної навігації. Інтерфейс включає систему фільтрації, яка дозволяє користувачам відсіювати записи за категоріями спорту або виконувати пошук за ключовими словами через текстове поле. Серверна обробка цих дій вимагає динамічного формування умовних SQL-запитів із обов'язковим очищенням вхідних параметрів для захисту від потенційних вразливостей. Кожна інформаційна картка події містить назву, дату у форматі локального календаря та назву спортивного майданчика, яка підтягується через механізм об'єднання таблиць JOIN, що забезпечує цілісність сприйняття інформації.

3 ПРОГРАМНА РЕАЛІЗАЦІЯ

3.1 Опис середовища розробки та реалізація серверної логіки

Програмна реалізація інформаційної системи планування спортивних подій та продажу квитків здійснювалася з використанням локального контуру веброзробки, розгорнутого за допомогою кросплатформного пакету інструментів XAMPP. Основним елементом серверної інфраструктури виступає вебсервер Apache, який виконує функції обробки вхідних HTTP-запитів, керування маршрутизацією та передачі керування інтерпретатору мови програмування PHP 8.2 для динамічної генерації контенту. Організація збереження даних, забезпечення їх транзакційної цілісності та оперативне виконання аналітичних вибірок реалізовано на базі реляційної системи керування базами даних MySQL. Для адміністрування схем таблиць, проєктування зв'язків між сутностями та первинного наповнення бази даних тестовими записами застосовувався інтегрований вебграфічний інтерфейс утиліти phpMyAdmin. Безпосереднє написання програмного коду модулів системи, верстка користувацьких інтерфейсів та налагодження скриптів виконувалися в інтегрованому середовищі розробки Visual Studio Code з використанням інструментів розробника сучасних веббраузерів для моніторингу мережевої активності та валідації клієнтських сценаріїв.

Функціональний базис серверної частини розроблюваного вебзастосунку орієнтований на забезпечення надійної взаємодії з базою даних та реалізацію механізмів контролю доступу користувачів до ресурсів платформи. Взаємодія серверних скриптів із реляційним сховищем MySQL організована за допомогою об'єктно-орієнтованого інтерфейсу mysqli, що дозволяє мінімізувати затримки при виконанні складних SQL-запитів та гарантує захист від ін'єкцій. Підсистема реєстрації та авторизації відвідувачів базується на обов'язковій серверній перевірці унікальності ідентифікаторів користувача у базі даних та ініціалізації

сесій через виклик вбудованої функції роботи з сесіями. Після успішної автентифікації ключові параметри поточного користувача, зокрема його унікальний ідентифікатор та рівень прав доступу, фіксуються у суперглобальному масиві сесій, що дозволяє виконувати розмежування доступу до сторінок клієнтської частини та закритих адміністративних модулів застосунку.

Найбільш критичним вузлом серверної логіки є модуль оформлення замовлень та купівлі електронних квитків, який реалізовано у функціональному файлі обробки транзакцій. Даний скрипт у реальному часі приймає параметри сформованої користувачем форми, виконує додаткову перевірку наявності вільних місць на обраний спортивний захід безпосередньо в базі даних та розраховує фінальну вартість замовлення. Для запобігання виникненню помилок дублювання місць або незавершених фінансових операцій у випадку одночасних запитів від різних користувачів, обробка даних на рівні СКБД виконується із застосуванням механізму вбудованих транзакцій, що гарантує атомарність та узгодженість інформації.

3.2 Реалізація роботи з БД

Взаємодія серверних PHP-скриптів із реляційною базою даних sport_events_db, адміністрування та налаштування структурних таблиць якої здійснювалося через графічний веб-інтерфейс утиліти phpMyAdmin [10], реалізована за допомогою об'єктно-орієнтованого розширення mysqli, яке забезпечує високу швидкість виконання запитів та безпеку передачі параметрів. З метою орієнтації на максимальну швидкодію системи та забезпечення цілісності даних замість множинних поодиноких звернень було впроваджено складні SQL-запити з механізмом об'єднання таблиць. Основним запитом, що забезпечує коректну роботу головної сторінки та динамічного розділу подій, виступає вибірка даних із використанням оператора JOIN. Це архітектурне

рішення дозволяє в межах одного запиту одночасно отримати повну назву події, її категорію, дату проведення, а також оперативно підтягнути необхідні дані про спортивну локацію з іншої пов'язаної таблиці майданчиків:

```
SELECT e.id_event, e.name AS event_name, e.event_date, e.category, v.name AS venue_name
FROM event e
JOIN venue v ON e.id_venue = v.id_venue
WHERE e.event_date >= NOW()
ORDER BY e.event_date ASC
LIMIT 3;
```

Для надійної фіксації фінансових операцій та обліку проданих місць у структурі обробника process_purchase.php використовується запит типу INSERT, який функціонує в обов'язковій парі з механізмом вбудованих транзакцій СКБД. Такий підхід є критично важливим для інформаційних систем спортивного призначення, оскільки він повністю виключає виникнення аномалій даних та неприпустимих ситуацій, за яких платіж користувача успішно оброблено, але запис про генерацію квитка не був зафіксований у базі даних через раптовий збій зв'язку. Перед кожним внесенням даних серверний скрипт ініціює блокування відповідних рядків таблиці, що гарантує атомарність операції та надійно захищає систему від овербукінгу при паралельних запитах на одні й ті самі місця.

3.3 Реалізація інтерфейсу

Клієнтська частина розробленої інформаційної системи повністю реалізована та скомпонована з урахуванням сучасних стандартів UX/UI-дизайну для забезпечення максимальної адаптивності, кросбраузерності та інтуїтивності вебінтерфейсу. Для візуального оформлення платформи було обрано ергономічну темну кольорову гаму, що суттєво знижує навантаження на зір

користувача в умовах тривалої взаємодії та забезпечує високий рівень контрасту текстових елементів. Ключовою технологічною особливістю реалізованого інтерфейсу є динамічний рендеринг елементів навігації на стороні фронтенду, який виконується сервером на основі аналізу поточного стану глобальних сесій та прав доступу користувача.

Головна сторінка вебзастосунку (файл index.php) функціонує як динамічна інтерактивна вітрина актуальних спортивних подій. Кожен інформаційний блок заходу програмно представлений у вигляді окремої модульної картки. Серверний PHP-скрипт в автоматичному режимі аналізує текстове значення категорії змагань із бази даних і динамічно підставляє відповідне фонове графічне зображення для футболу, баскетболу чи боксу, забезпечуючи високу інформативність інтерфейсу. Верхня навігаційна панель (Header) містить інтегровані конструкції перевірки автентифікації: для неавторизованих гостей відображаються елементи переходу до форм входу, а для авторизованих користувачів сервер виводить поточне ім'я профілю та посилання на закриті розділи сайту або адмін-панель. Реалізований та наповнений тестовими даними графічний інтерфейс головної сторінки інформаційної системи представлено на рисунку 3.1.

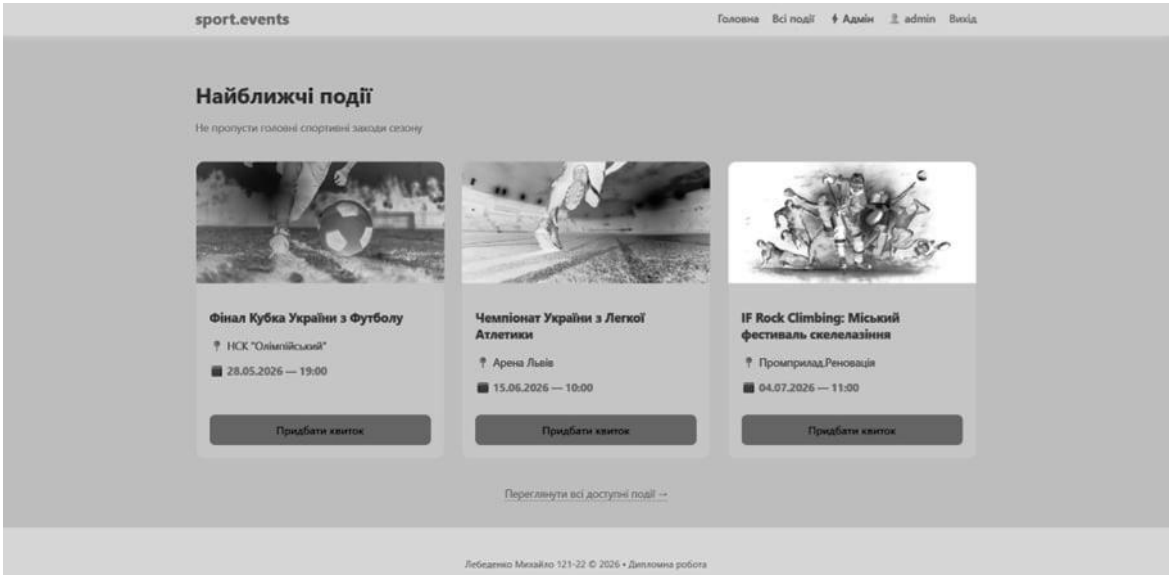


Рисунок 3.1 – Графічний інтерфейс головної сторінки

Інтерфейсна частина сторінки списку всіх подій (events.php) оптимізована для коректної роботи з великими обсягами динамічних даних. Пошуковий модуль містить інтерактивне текстове поле для фільтрації за назвою заходу та спадний список для сортування контенту за категоріями спорту. Для забезпечення якісного користувацького UX-відгуку в системі передбачено логічний вивід інформаційного повідомлення у випадку відсутності записів у базі даних за критеріями користувача, що дозволяє уникнути відображення порожніх областей екрана.

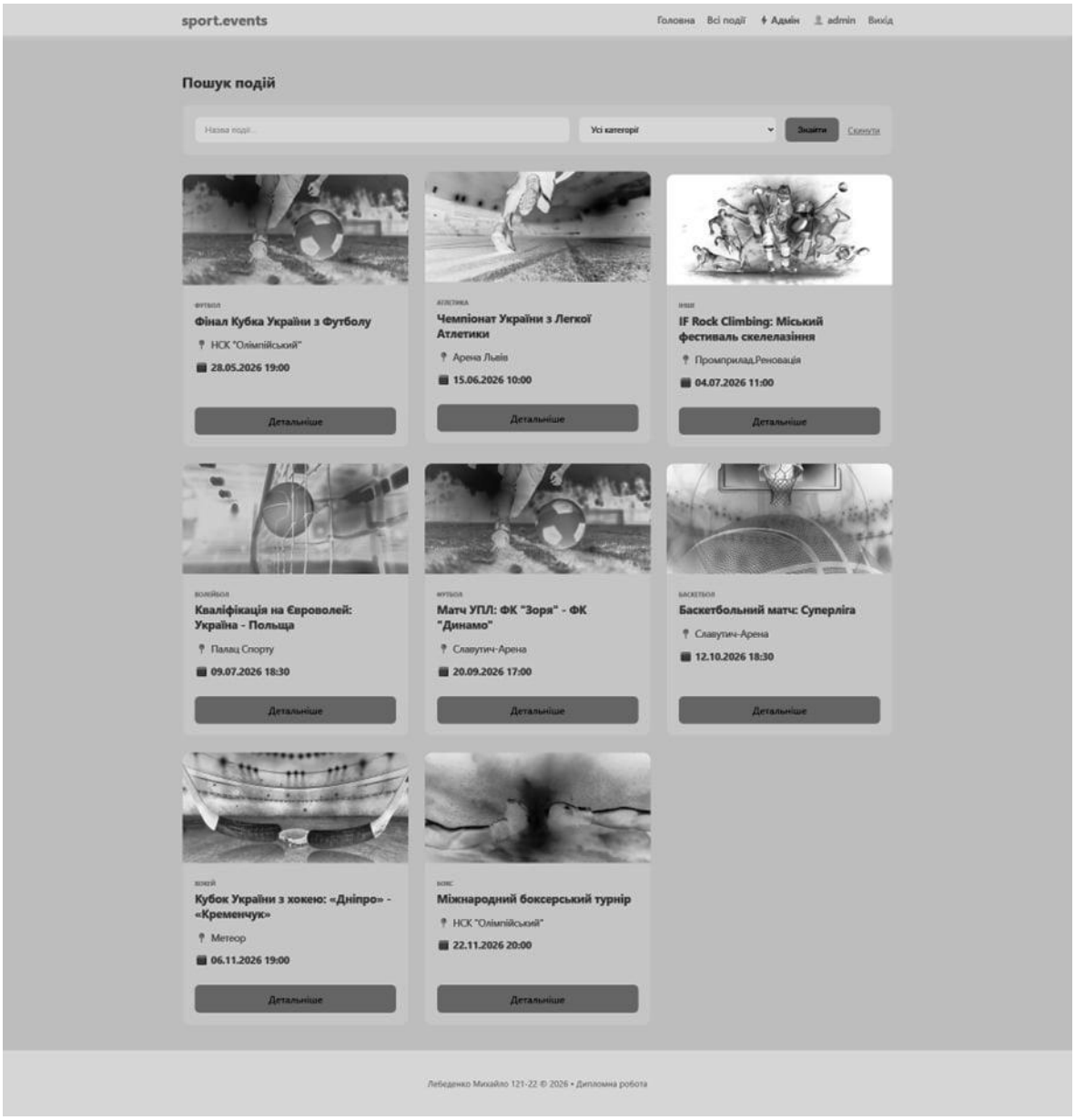


Рисунок 3.2 – Інтерфейс сторінки всіх подій з фільтрами пошуку

Сторінка детальної інформації про захід та модуля оплати (event_details.php) виступає ключовим вузлом інтерактивної взаємодії користувача з платформою. Функціональний блок виведення даних відображає повний опис спортивної події, дату, точний час проведення та адресу локації, яка підтягується з бази даних за допомогою об'єднання з таблицею venue. Інтерфейс форми покупки містить інтерактивне поле для вибору кількості квитків, а також інтегрований клієнтський JavaScript-сценарій. Цей скрипт динамічно керує станом форми під час оплати, змінюючи текст кнопки на повідомлення про обробку та активуючи анімований елемент завантаження, що покращує користувацький UX-відгук та забезпечує візуальне підтвердження старту транзакції.

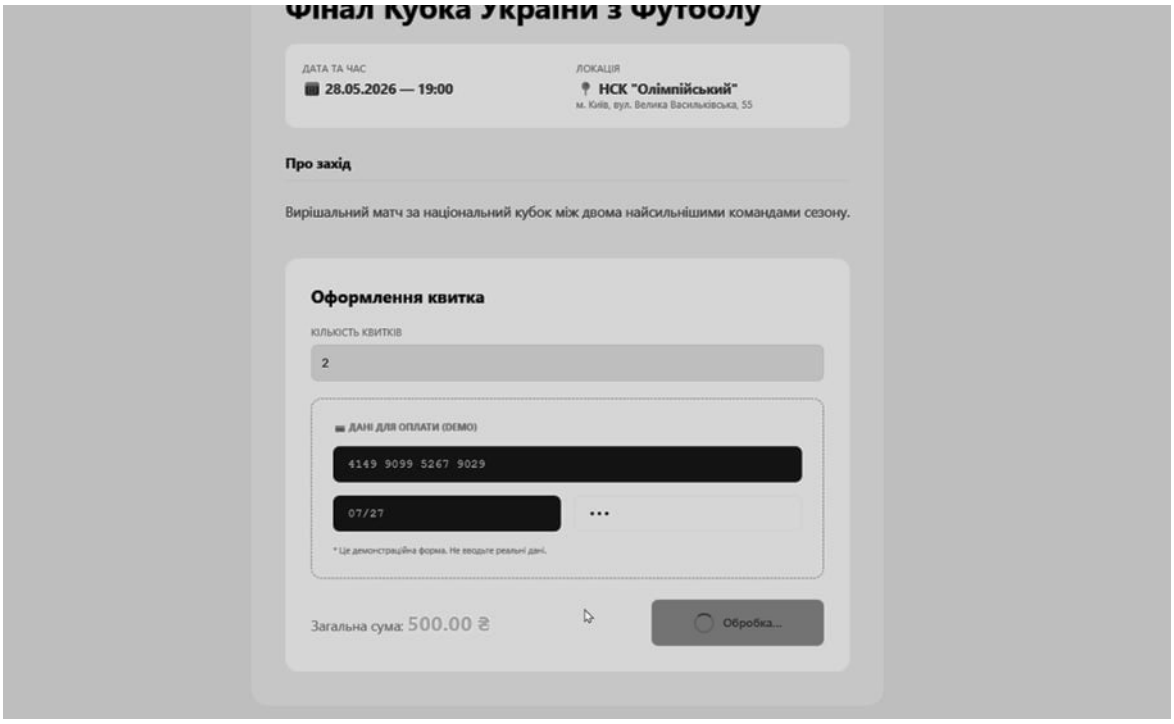


Рисунок 3.3 – Сторінка оформлення замовлення з ефектом завантаження

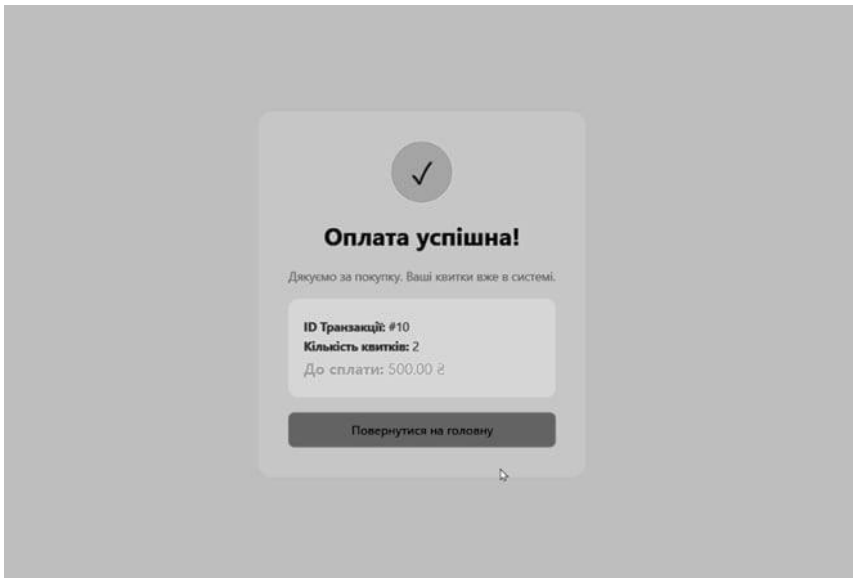


Рисунок 3.4 – Сторінка підтвердження оплати замовлення

Адміністративна панель вебзастосунку (admin.php) розроблена як закритий інтерфейс для моніторингу та оперативного керування контентом. На сторінці реалізовано зведене табличне відображення всіх наявних у базі даних спортивних подій з інтегрованою можливістю їх миттєвого видалення та створення нових записів. Форма створення нових заходів програмно змонтована у правому блоці робочого простору, що дозволяє адміністратору швидко вносити актуальні івенти, обираючи доступні спортивні локації з динамічного випадаючого списку.

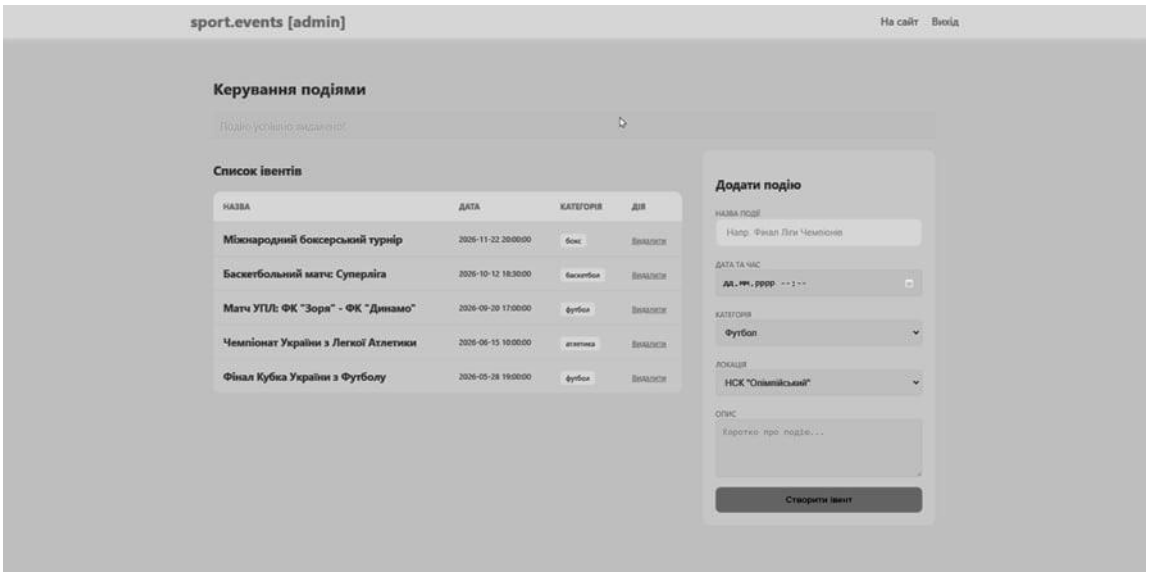


Рисунок 3.5 – Інтерфейс панелі адміністратора для управління контентом

Інтерфейсні форми сторінок особистого кабінету та авторизації користувачів (profile.php, login.php) реалізовані у мінімалістичному графічному стилі з центральним екранним розташуванням інтерактивних карток. У просторі особистого кабінету автентифікований користувач отримує повноцінний функціональний доступ до перегляду персональної історії раніше придбаних електронних квитків. Відображення цього списку динамічно формується на стороні сервера за допомогою оптимізованого SQL-запиту з об'єднанням даних із реляційних таблиць transaction та ticket за індивідуальним ідентифікатором сесії поточного користувача.

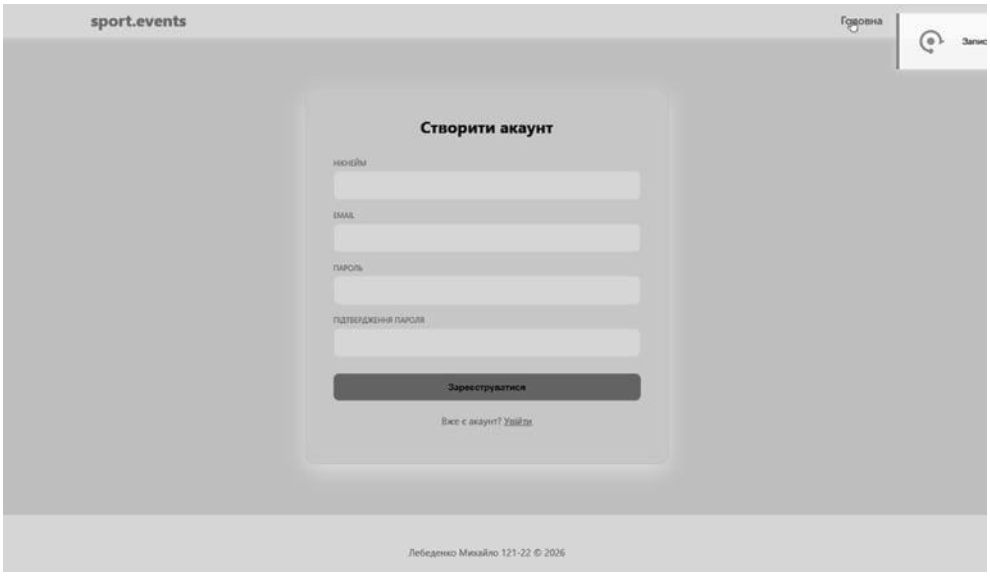


Рисунок 3.6 – Інтерфейс сторінки авторизації



Рисунок 3.7 – Інтерфейс сторінки входу до акаунту

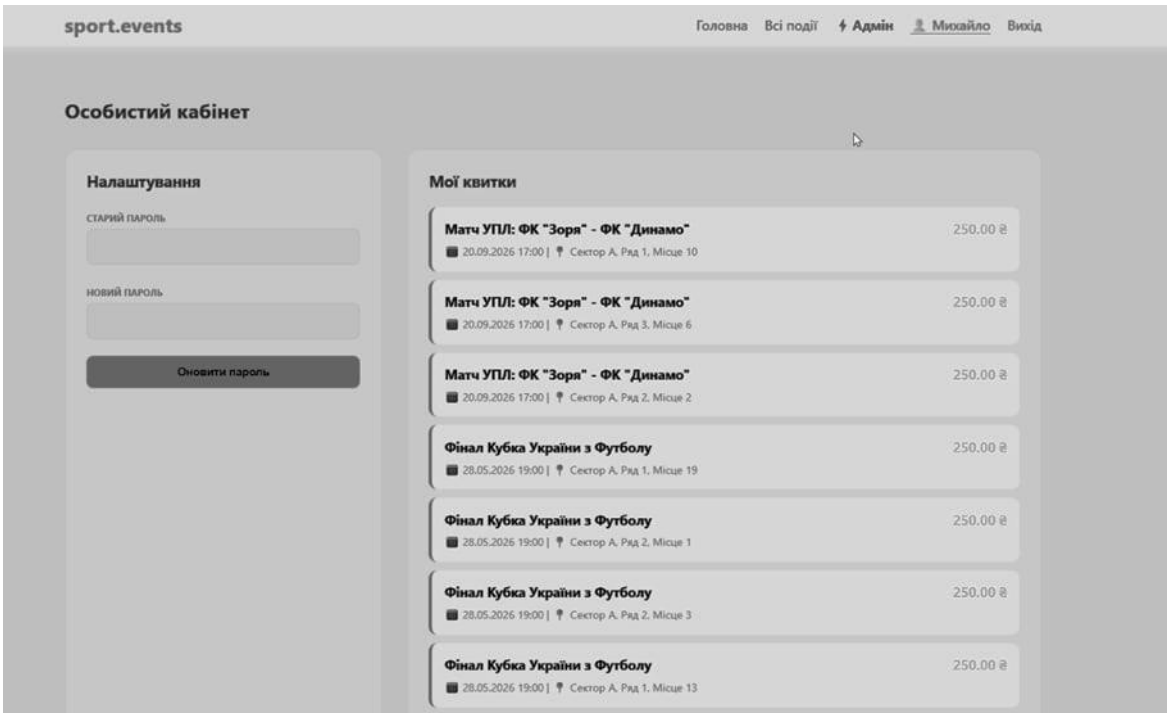


Рисунок 3.8 – Інтерфейс особистого кабінету

Усі елементи розробленого графічного інтерфейсу користувача є повністю адаптивними завдяки комплексному використанню технології CSS Flexbox та динамічних CSS-змінних. Такий підхід дозволяє забезпечити коректне та гнучке відображення структурних блоків інформаційної системи на екранах мобільних пристроїв, планшетів та персональних комп'ютерів із різною роздільною здатністю без втрати загальної функціональності сайту [11].

3.4 Забезпечення безпеки

У процесі проєктування та програмної реалізації інформаційної системи особлива увага була приділена захисту персональних даних користувачів, забезпеченню конфіденційності транзакцій та запобіганню поширеним типам зловмисних дій. Основні інструменти та механізми безпеки інтегровані безпосередньо на рівні серверної логіки мови PHP та підсистеми взаємодії з реляційною базою даних.

Для повного усунення ризику виникнення SQL-ін’єкцій, які є однією з найбільш критичних вразливостей сучасних вебзастосунків, у розробленій системі повністю виключено метод прямої вставки вхідних змінних у тіло запитів. Замість цього розробка базується на використанні механізму підготовлених виразів (Prepared Statements) об'єкта mysqli. Суть даного методу захисту полягає в тому, що загальна структура SQL-шаблону надсилається до сервера баз даних окремо від самих параметрів користувача, що унеможливлює модифікацію логіки виконання запиту навіть при введенні спеціальних або екранованих символів у текстові поля форм. Приклад практичної реалізації захищеного серверного запиту для виведення детальної інформації про захід у файлі event_details.php наведено нижче.

```
$stmt = $mysqli->prepare("SELECT * FROM event WHERE id_event = ?");
$stmt->bind_param("i", $event_id);
$stmt->execute();
```

З метою захисту облікових записів відвідувачів платформи та запобігання компрометації автентифікаційних даних у випадку гіпотетичного отримання зловмисниками повного доступу до бази даних, система не зберігає паролі користувачів у відкритому текстовому вигляді. Для криптографічної обробки паролів на етапі реєстрації застосовано сучасний стійкий алгоритм BCRYPT, реалізований через вбудовану функцію password_hash. Даний алгоритм в автоматичному режимі генерує та додає унікальну криптографічну «сіль» до кожного значення, забезпечуючи високий рівень стійкості до атак методом повного перебору. Подальша автентифікація та перевірка відповідності введеного пароля під час авторизації здійснюється в безпечному контурі за допомогою функції password_verify.

Додатковим бар'єром захисту виступає багаторівнева валідація та фільтрація всіх вхідних даних, які надходять від форм реєстрації, логування або модулів оформлення квитків. Клієнтська валідація реалізована за допомогою

стандартних атрибутів required, специфікації типів полів та обмежень діапазону безпосередньо в HTML-коді, що гарантує миттєвий зворотний зв'язок для користувача. Серверна фільтрація виконується скриптами PHP за допомогою функцій видалення зайвих пробілів та інструментів перевірки коректності поштових адрес filter_var, що повністю блокує передачу некоректних параметрів та запобігає виконанню міжсайтового скриптингу.

Для захисту від сесійних атак та мінімізації ризиків викрадення активних ідентифікаторів (Session Hijacking) у модулі авторизації після кожної успішної перевірки облікових даних викликається функція примусової регенерації сесії з видаленням старої мітки. Також у системі реалізовано жорстку перевірку сесійних змінних ролі адміністратора у файлі admin.php, що повністю виключає можливість несанкціонованого доступу сторонніх осіб до панелі керування через пряме введення URL-адреси в рядок браузера.

3.5 Тестування системи

Кінцевим етапом програмної реалізації інформаційної системи планування спортивних подій та продажу квитків є проведення комплексного тестування для перевірки відповідності розробленого програмного продукту висунутим вимогам технічного завдання та виявлення можливих логічних чи архітектурних помилок [12]. Для проведення верифікації було обрано метод функціонального тестування «чорної скриньки». Даний підхід дозволяє зосередити увагу на аналізі коректності обробки вхідних даних та відповідності отриманих результатів очікуваним станам системи без втручання безпосередньо у вихідний код під час виконання перевірочних сценаріїв.

Першочерговому аналізу було піддано підсистему реєстрації, автентифікації та розмежування прав доступу користувачів. Під час перевірки сценарію створення нового облікового запису з коректними параметрами було зафіксовано успішне внесення інформації до реляційної таблиці user, коректне

відпрацювання алгоритму хешування паролів та автоматичний редирект на форму логування. При спробі повторної реєстрації з використанням електронної пошти чи логіну, які вже наявні в базі даних, серверний скрипт успішно заблокував операцію та вивів інформаційне повідомлення про зайняті ідентифікатори, що підтверджує працездатність механізмів запобігання дублюванню записів. Перевірка форми входу з використанням хибних автентифікаційних даних продемонструвала, що вбудована функція password_verify повертає негативне значення, повністю блокуючи несанкціонований доступ до закритих розділів сайту.

Наступним критичним етапом було тестування процесу вибору посадкових місць та безпосереднього оформлення покупки квитків, що безпосередньо впливає на транзакційну цілісність бази даних. Тестування транзакційного модуля авторизованим користувачем підтвердило коректне ініціювання SQL-транзакції, успішну фіксацію фінансових параметрів у таблиці transaction, зміну внутрішнього статусу доступності місця та генерацію фінальної сторінки з унікальним ідентифікатором замовлення. У випадку спроби оформлення покупки неавторизованим гостем, система безпеки на основі перевірки глобального масиву сесій успішно заблокувала процес та виконала перенаправлення на сторінку login.php з вимогою попереднього логування.

Фінальний блок тестування охоплював перевірку адміністративних функцій та механізмів обробки виняткових ситуацій. При спробах надіслати форму покупки з некоректною, нульовою або від’ємною кількістю квитків, клієнтські обмеження атрибутів min у HTML-розмітці разом із дублюючою серверною валідацією у файлі process_purchase.php повністю заблокували передачу шкідливих параметрів. Перевірка надійності обмеження доступу до файлу admin.php показала, що спроби прямого переходу за URL-адресою без наявності прав адміністратора миттєво припиняються сесійною перевіркою, яка автоматично перенаправляє стороннього користувача на головну сторінку. Модуль видалення подій в адмін-панелі продемонстрував миттєве виконання каскадних операцій DELETE на рівні бази даних та синхронне оновлення

користувацької афіші. Також було підтверджено ефективність клієнтського JavaScript-сценарію, який при першому натисканні на кнопку оплати динамічно переводив її у стан disabled та запускав анімацію завантаження, що повністю виключило ризик повторного відправлення дублюючих запитів та виникнення помилок овербукінгу при випадковому подвійному кліку з боку користувача.

3.6 Перспективи розвитку

Розроблена інформаційна система планування спортивних подій та продажу квитків має гнучку модульну структуру, що дозволяє проводити подальшу модернізацію та нарощування функціональних можливостей без втручання в основне архітектурне ядро програми. На основі проведеного аналізу результатів тестування та консультацій із керівником проєкту було визначено три ключові пріоритетні напрямки подальшого технологічного розвитку програмного продукту.

Першим етапом модернізації є інтеграція реальних банківських платіжних шлюзів, таких як LiqPay або Stripe, за допомогою підключення офіційних API-інтерфейсів провайдерів. На поточному етапі розробки в системі функціонує надійна програмна імітація платіжного контуру, яка дозволяє повноцінно перевірити транзакційну логіку взаємодії з базою даних та коректність обробки фінансових статусів замовлень. Відмова від прямого збору, обробки та збереження конфіденційних даних платіжних карток відвідувачів у локальній базі даних є свідомим безпековим рішенням. Збереження такої інформації в реальних умовах функціонування комерційних систем потребує суворої відповідності міжнародному стандарту безпеки PCI DSS та проходження сертифікації [13]. Архітектура сервісу спроєктована із урахуванням ізоляції модулів, що дозволяє безперешкодно замінити поточний тестовий шматок коду на бойове інтеграційне рішення, забезпечуючи високий рівень безпеки

фінансових даних через повне делегування їхньої обробки сертифікованим платіжним системам.

Другим напрямком розвитку є розширення функціоналу підсистеми генерації документів шляхом впровадження динамічних QR-кодів для швидкої верифікації квитків. У структурі реляційній бази даних, зокрема в таблиці tickets, на етапі проєктування вже були закладені необхідні поля для збереження унікальних текстових ідентифікаторів та унікальних хеш-лінків замовлень. Наступним кроком модернізації є підключення профільної серверної бібліотеки phpqrcode, що дозволить повністю автоматизувати процес генерації графічних QR-кодів у форматі PNG на основі згенерованого ID транзакції. Це забезпечить миттєве зчитування та перевірку квитків за допомогою ручних сканерів або смартфонів контролерів безпосередньо на точках контролю доступу під час проведення спортивних заходів, трансформуючи вебзастосунок у повноцінний цифровий сервіс безпаперового документообігу.

Для забезпечення надійної комунікації між сервером інформаційної системи та зовнішніми сервісами месенджера в рамках модернізації було запроєктовано архітектурне рішення на базі REST API, яке дозволить обробляти вхідні запити від бота та повертати структуровані дані про розклад спортивних матчів у стандартизованому форматі JSON. Загальна концептуальна схема інтеграції розробленого бекенду із зовнішніми мобільними сервісами через API-шлюз представлена нижче на рисунку 3.9.



Рисунок 3.9 – Схема архітектурної інтеграції REST API з Telegram-ботом

З метою підтвердження технічної реалізованості та логічної узгодженості запропонованого інтеграційного функціоналу було сформовано концептуальний архітектурний прототип майбутнього ендпоінту розкладу подій (цільовий файл api/get_events.php). Даний серверний скрипт описує базову інженерну логіку інтерфейсу програмування додатків, демонструє механізм безпечного підключення до реляційного сховища, виконання оптимізованої вибірки актуальних івентів та їх подальшу трансформацію в об'єкт JSON для коректної інтерпретації на стороні Telegram-бота. Концептуальна програмна реалізація базової логіки ендпоінту REST API для майбутньої зовнішньої інтеграції наведена в лістингу коду нижче:

```
<?php
header("Content-Type: application/json; charset=UTF-8");
require_once '../config/db.php';

if ($_SERVER['REQUEST_METHOD'] === 'GET') {
    $query = "SELECT id_event, name, event_date FROM event WHERE event_date >= NOW() ORDER BY event_date ASC";
    $result = $mysqli->query($query);

    $events = [];
    while ($row = $result->fetch_assoc()) {
        $events[] = $row;
    }

    http_response_code(200);
    echo json_encode([
        "status" => "success",
        "data" => $events
    ], JSON_UNESCAPED_UNICODE);
} else {
    http_response_code(405);
    echo json_encode(["status" => "error", "message" => "Method Not Allowed"]);
}
?>
```

Впровадження запроєктованого інтеграційного інтерфейсу дозволить суттєво розширити цільову аудиторію платформи, автоматизувати маркетинг квиткової дистрибуції та забезпечити користувачам сучасний рівень мобільного сервісу.

ВИСНОВКИ

У процесі виконання кваліфікаційної роботи було повністю досягнуто поставлену мету та розроблено функціональну інформаційну систему планування спортивних подій і продажу квитків. У межах дослідження було забезпечено наскрізний цикл розробки програмного продукту, починаючи від системного аналізу предметної області та проєктування архітектурних рішень і закінчуючи практичною програмною реалізацією та верифікацією стабільності модулів вебзастосунку в локальному серверному середовищі.

На етапі системного аналізу та обґрунтування концепції проєкту було визначено ключові архітектурні вимоги до платформи, що дозволило обрати класичну трирівневу клієнт-серверну модель для чіткого розмежування інтерфейсу користувача, бізнес-логіки та рівня збереження даних. Спроєктована реляційна база даних пройшла процедуру нормалізації, що дозволило мінімізувати надлишковість інформації, оптимізувати швидкість виконання SQL-запитів та гарантувати цілісність даних на рівні сутностей. Програмна реалізація backend-частини виконана із застосуванням мови програмування PHP та СКБД MySQL, що забезпечило високу швидкість обробки HTTP-запитів та надійне функціонування критичних модулів, серед яких: багаторівнева автентифікація, розмежування прав доступу ролей користувачів, динамічна фільтрація афіші заходів та транзакційний механізм бронювання посадкових місць у реальному часі. Розроблений адаптивний інтерфейс адміністратора дозволив повністю автоматизувати процеси менеджменту контенту та знизити вплив людського фактора під час планування спортивних змагань.

Особлива увага в роботі була приділена програмним засобам забезпечення безпеки, захисту персональних даних та запобіганню поширеним вебвразливостям. Інтеграція підготовлених запитів на рівні взаємодії з базою даних дозволила повністю усунути загрозу виникнення SQL-ін’єкцій, а використання сучасного криптографічного алгоритму BCRYPT гарантує

стійкість збереження паролів користувачів до атак методом повного перебору. Проєктування модуля обробки фінансових транзакцій з використанням безпечного імітаційного контуру дозволило промоделювати логіку купівельного процесу без прямого збереження платіжних даних на локальному сервері, що відповідає вимогам міжнародного стандарту PCI DSS. Проведене комплексне функціональне тестування за методом «чорної скриньки» підтвердило стабільність, транзакційну цілісність, захищеність та високу продуктивність розробленої інформаційної системи, архітектурний потенціал якої дозволяє здійснювати подальшу модернізацію через підключення реальних платіжних API та інтеграцію модулів REST API для мобільних платформ.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ticketmaster : вебсайт. URL: https://www.ticketmaster.com (дата звернення: 26.03.2026).

2. Concert.ua : офіційний вебсайт. URL: https://concert.ua (дата звернення: 26.03.2026).

3. Karabas.com : офіційний вебсайт. URL: https://karabas.com (дата звернення: 26.03.2026).

4. Гуржій А.М., Поворознюк Н.І., Самсонов В.В. Інформатика та інформаційні технології: підручник. Харків : ТОВ «Компанія СМІТ», 2022. 352 с.

5. MDN Web Docs. Спільнота розробників Mozilla. URL: https://developer.mozilla.org/ (дата звернення: 18.04.2026).

6. Офіційна документація PHP. URL: https://www.php.net/manual/uk/ (дата звернення: 18.04.2026).

7. Вебсайт офіційної документації MySQL. URL: https://dev.mysql.com/doc/ (дата звернення: 15.04.2026).

8. Крамаренко М. В. Інформаційні системи в управлінні: практикум. Київ : КНЕУ, 2019. 196 с.

9. IT Education Center Blog : блог навчального центру DevOps. URL: https://itedu.center/ua/blog/ (дата звернення: 21.04.2026).

10. Документація phpMyAdmin. URL: https://docs.phpmyadmin.net/ (дата звернення: 20.04.2026).

11. W3Schools : Інтерактивний довідник з HTML, CSS, JS. URL: https://www.w3schools.com/ (дата звернення: 19.04.2026).

12. Stack Overflow : спільнота розробників. URL: https://stackoverflow.com/ (дата звернення: 15.04.2026).

13. PCI Security Standards Council. Official Site. URL: https://www.pcisecuritystandards.org/ (дата звернення: 20.04.2026).

Декларація

академічної доброчесності

здобувача освіти ВСП «Економіко-правничий фаховий коледж ЗНУ»

Я, Лебеденко Михайло Володимирович, здобувач освіти 4 курсу, спеціальності/освітньо-професійної програми 121 Інженерія програмного забезпечення, групи К121-22, адреса електронної пошти lebedenkomihail@ukr.net

- підтверджую, що написана мною кваліфікаційна робота на тему «Розробка інформаційної системи з продажу квитків на спортивні події» відповідає вимогам академічної доброчесності та не містить порушень, що визначені у ст. 42 Закону України «Про освіту», зі змістом яких ознайомлений;

- заявляю, що надана мною для перевірки електронна версія роботи є ідентичною її друкованій версії;

- згоден на перевірку моєї роботи на відповідність критеріям академічної доброчесності у будь-який спосіб, у тому числі за допомогою інтернет-системи, а також на архівування моєї роботи в базі даних цієї системи.

Дата _____ Підпис _____ Михайло ЛЕБЕДЕНКО

Дата 01.06.2026 Підпис  Юлія БОРИСОВСЬКА