

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних систем управління

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему:

**«Розробка веб-орієнтовної системи для організації
роботи спортивного залу»**

Виконав: студент 2-го курсу, групи
2АКІТР-23м спеціальності 174 –
Автоматизація, комп'ютерно-
інтегровані технології та
роботехніка

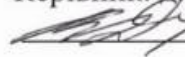
(шифр і назва напрямку підготовки, спеціальності)



Віталій КАРВАЦЬКИЙ

(ім'я та прізвище)

Керівник: д.т.н., проф каф. КСУ



Володимир ДУБОВОЙ

(ім'я та прізвище)

« 4 » 12 2024 р.

Опонент: к.т.н., професор каф. АІТ



Євген ПАЛАМАРЧУК

(ім'я та прізвище)

« 4 » 12 2024 р.

Допущено до захисту

Зав. кафедри КСУ, д.т.н., проф.

В'ячеслав КОВТУН

(ім'я та прізвище)

« 4 » 12 2024 р.

Вінниця ВНТУ - 2024 рік

Вінницький національний технічний університет
Факультет інтелектуальних інформаційних технологій та автоматизації
Кафедра комп'ютерних систем управління
Рівень вищої освіти другий (магістерський)
Галузь знань – 17 – Електроніка, автоматизація та електронні комунікації
Спеціальність – 174 – Автоматизація, комп'ютерно-інтегровані технології та
робототехніка
Освітньо-професійна програма – Інтелектуальні комп'ютерні системи

ЗАТВЕРДЖУЮ
Зав. кафедри КСУ

 В'ячеслав КОВТУН

“14” жовтня 2024
року

ЗАВДАННЯ
НА МАГІСТЕРСЬКУ КВАЛІФІКАЦІЙНУ РОБОТУ
студенту Карвацькому Віталію Юрійовичу.
(прізвище, ім'я, по батькові)

1. Тема роботи. Розробка веб-орієнтовної системи для організації роботи спортивного залу

керівник роботи д.т.н., професор Дубовой В.М

затверджені наказом ВНТУ від “17” вересня 2024 року №310

2. Термін подання студентом роботи “1” грудня 2024 року

3. Вихідні дані до роботи: реєстрація та авторизація користувачів, бронювання послуг, відстеження розкладу занять, управління персоналом, ведення обліку клієнтів та їх підписок.

1. Кашуба В. О. Застосування сучасних інформаційних технологій у період проведення та завершення спортивних змагань. Фізичне виховання, спорт і культура здоров'я у сучасному суспільстві: зб. наук. пр. Волин. нац. університету ім. Лесі Українки, Луцьк. 2012. №7. С. 119 – 126.

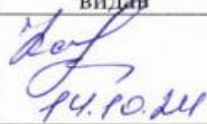
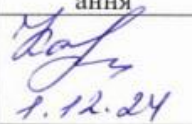
2. Ладика П. Сучасні комп'ютерні технології у фізичному вихованні і спорті. Актуальні аспекти фізичного виховання, спорту і здоров'я людини: зб. наук. пр. Тернопіль, 2013. С. 128 – 134.

3. Лядська О. Ю. Застосування комп'ютерної програми “Fitball training” для удосконалення організації фізкультурно-оздоровчих занять з жінками першого зрілого віку із застосуванням футболу / О. Ю. Лядська // Педагогіка, психологія та медико-біологічні проблеми фізичного виховання і спорту. 2014. № 12. С. 76–80.

4. Зміст текстової частини: вступ, огляд і аналіз існуючих аналогів, специфікація вимог до системи, стан розв'язання технічної задачі, розроблення архітектури програмної системи, проєктування структури бази даних та тестування веб-орієнтованої системи для організації роботи спортивного залу.

5. Перелік ілюстративного матеріалу (з точним зазначенням обов'язкових креслень), об'єкт, предмет та мета дослідження, DFD-моделі предметної області, ER-діаграма розробленої веб-орієнтованої системи для користування спортивним залом та інші блок-схеми.

1. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	виконання
4	Кавецький В.В. – доцент кафедри економіки підприємства і виробничого менеджменту	 24.10.24	 1.12.24

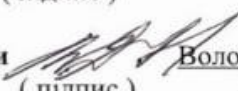
2. Дата видачі завдання “14” жовтня 2024 року

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва та зміст етапу	Термін виконання		Примітка
		початок	закінчення	
1	Дослідження стану автоматизації і проєктування веб-орієнтованої інформаційної системи роботи спортивного залу	20.10.2024	30.10.2024	
2	Дослідження стану розв’язання технічної задачі	25.10.2024	10.11.2024	
3	Розробка та тестування веб-орієнтованої системи	30.10.2024	25.11.2024	
4	Розрахунок економічної частини	01.11.2024	30.11.2024	
5	Оформлення матеріалів до захисту	25.11.2024	30.11.2024	

Студент 
(підпис)

Віталій КАРВАЦЬКИЙ
(Ім'я ПРІЗВИЩЕ)

Керівник роботи 
(підпис)

Володимир ДУБОВОЙ
(Ім'я ПРІЗВИЩЕ)

АНОТАЦІЯ

Карвацький В.Ю. Розробка веб-орієнтовної системи для організації роботи спортивного залу. Магістерська кваліфікаційна робота зі спеціальності 174 – Автоматизація, комп'ютерно-інтегровані технології та робототехніка. Вінниця. ВНТУ. 2024. -113 с.

На укр. мові: Бібліограф. – 30 назв., рис. – 44, табл. – 33.

У магістерській кваліфікаційній роботі розглядаються питання проєктування, розробки та впровадження веб-орієнтованої інформаційної системи для управління роботою спортивного залу.

Метою магістерської кваліфікаційної роботи є розробка веб-орієнтованої системи для організації роботи спортивного залу, яка дозволить користувачам ефективно планувати та управляти графіками тренувань, розкладом заходів та взаємодією між клієнтами, тренерами та адміністрацією.

У ході виконання роботи були проаналізовані сучасні аналоги програмного забезпечення, їхні переваги та недоліки, що дозволило сформулювати вимоги до майбутньої системи.

Розроблена система має інтуїтивний інтерфейс, який дозволяє працівникам ефективно працювати з розкладами, ресурсами та фінансовими операціями, а клієнтам – легко записуватися на тренування, оплачувати послуги та отримувати актуальну інформацію про доступність занять.

У роботі також описано процес тестування системи, результати якого підтвердили її відповідність заданим вимогам. Окремий розділ присвячено економічному обґрунтуванню впровадження системи та аналізу її ефективності для підвищення продуктивності роботи спортивного залу.

Графічна частина складається з 10 плакатів із результатами моделювання.

Ключові слова: веб-орієнтована інформаційна система; автоматизація бізнес-процесів; спортивний зал; управління розкладом.

ABSTRACT

Karvatskyi V.Yu. Development of a Web-Oriented System for Organizing the Operation of a Gym. Master's Qualification Thesis in Specialty 174 – Automation, Computer-Integrated Technologies, and Robotics. Vinnytsia, VNTU, 2024. - 113 p.

In Ukrainian language: Bibliography – 30 titles, fig. – 44, tabl. – 33.

The master's qualification thesis addresses the design, development, and implementation of a web-oriented information system for managing the operations of a gym.

The goal of the master's qualification thesis is to develop a web-oriented system for organizing gym operations, enabling users to efficiently plan and manage training schedules, event timetables, and interactions between clients, trainers, and the administration.

The developed system features an intuitive interface that enables staff to efficiently manage schedules, resources, and financial transactions, while allowing customers to easily book training sessions, make payments, and access up-to-date information on available classes.

The thesis also describes the testing process of the system, with results confirming its compliance with the specified requirements. A separate section is devoted to the economic justification for the system's implementation and the analysis of its effectiveness in improving the productivity of gym operations.

The graphical part consists of 10 posters showcasing the modeling results.

Keywords: web-oriented information system; business process automation; gym; schedule management.

ЗМІСТ

ВСТУП.....	4
1. ДОСЛІДЖЕННЯ СТАНУ АВТОМАТИЗАЦІЇ І ПРОЄКТУВАННЯ ВЕБ-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ РОБОТИ СПОРТИВНОГО ЗАЛАЗУ	7
1.1. Характеристика об'єкту дослідження, аналіз стану автоматизації	7
1.2. Опис предметної області.....	12
1.3. Огляд і аналіз існуючих аналогів.....	21
1.4. Специфікація вимог до системи.....	30
Висновки до розділу 1	38
2. РОЗРОБЛЕННЯ АРХІТЕКТУРИ ПРОГРАМНОЇ СИСТЕМИ	40
2.1. Архітектура веб-орієнтованої інформаційної системи та технології розв'язання поставлених задач.....	40
2.2. Проєктування структури бази даних	44
2.3. Функціональна структура задачі та підсистеми забезпечення функціональної частини.....	48
Висновки до розділу 2.....	51
3. ПРОЄКТУВАННЯ ТА ТЕСТУВАННЯ ВЕБ-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ РОБОТИ СПОРТИВНОГО ЗАЛУ	52
3.1. Програмна реалізація проєкту.....	52
3.1.1. Вибір програмних засобів для реалізації проєкту	52
3.1.2. Організація інтерфейсу з користувачем	57
3.2. Створення бази даних системи.....	61
3.3. Тестування.....	66
Висновки до розділу 3.....	69
4. ЕКОНОМІЧНА ЧАСТИНА	71
4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки.....	71
4.2 Розрахунок узагальненого коефіцієнта якості розробки	74
4.3 Розрахунок витрат на проведення науково-дослідної роботи	76

	3
4.3.1 Витрати на оплату праці	76
4.3.2 Відрахування на соціальні заходи.....	79
4.3.3 Сировина та матеріали	79
4.3.4 Розрахунок витрат на комплектуючі	80
4.3.5 Спецустаткування для наукових (експериментальних) робіт	81
4.3.6 Програмне забезпечення для наукових (експериментальних) робіт	82
4.3.7 Амортизація обладнання, програмних засобів та приміщень	82
4.3.8 Паливо та енергія для науково-виробничих цілей	84
4.3.9 Службові відрядження	85
4.3.10 Витрати на роботи, які виконують сторонні підприємства, установи і організації.....	85
4.3.11 Інші витрати.....	85
4.3.12 Накладні (загальновиробничі) витрати	86
Висновки до розділу 4.....	91
ВИСНОВКИ	92
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	93
ДОДАТКИ	96
Додаток А (обов'язковий) – Протокол перевірки на наявність запозичень ...	97
Додаток Б (обов'язковий) – Технічне завдання	98
Додаток В (обов'язковий) - Ілюстративна частина	103

ВСТУП

Актуальність теми. У сучасних умовах значна кількість людей приділяє увагу фізичній активності та відвідуванню спортивних залів. Ведення активного способу життя та заняття спортом є запорукою здоров'я і довголіття. Спорт не лише сприяє фізичному розвитку, але й формує дисципліну, зміцнює організм та покращує самопочуття. У сучасних умовах все більше людей відвідують спортивні зали, фітнес-центри, тренажерні зали та басейни з метою покращення фізичної форми та загального самопочуття.

Однак зростання кількості таких закладів, а також популярність активного відпочинку ставлять перед організаціями нові виклики у плануванні та координації тренувальних процесів. Організація тренувань та оптимальне використання часу є складним завданням, особливо при взаємодії різних груп клієнтів і тренерів. Проблеми, пов'язані з плануванням графіків та керуванням ресурсами, вимагають ефективного вирішення, яке допоможе спростити процес підготовки до тренувань, змагань та інших спортивних заходів.

Веб-орієнтовані системи є актуальними інструментами для автоматизації та оптимізації управління роботою спортивних закладів. Вони дозволяють ефективніше розпоряджатися ресурсами, зручніше організовувати графіки тренувань та підвищувати рівень обслуговування клієнтів. Розробка та впровадження такої системи здатні вирішити проблеми, які виникають під час взаємодії між клієнтами, тренерами та адміністрацією, сприяючи підвищенню продуктивності роботи спортивних залів.

Таким чином, розробка веб-орієнтованої системи для організації роботи спортивного залу є актуальною задачею, що відповідає потребам сучасного суспільства та ринку спортивних послуг

Мета - розробити веб-орієнтовану систему для організації роботи спортивного залу, яка дозволить користувачам ефективно планувати та управляти графіками тренувань, розкладом заходів та взаємодією між клієнтами, тренерами та адміністрацією. Для досягнення поставленої мети необхідно виконати наступні **завдання**:

1. Дослідити існуючі методи організації роботи спортивних залів і виявити основні проблеми та потреби користувачів.
2. Розробити функціонал системи, який включатиме можливість керування розкладом тренувань, змагань та бронюванням ресурсів.
3. Забезпечити зручний інтерфейс для клієнтів, тренерів та адміністрації.
4. Впровадити механізми автоматизації процесів планування та моніторингу ефективності використання часу.
5. Забезпечити безпеку даних та конфіденційність користувачів.

Об'єкт дослідження – веб-орієнтована система для організації роботи спортивного залу, яка спрямована на покращення ефективності планування тренувань та управління ресурсами.

Предмет дослідження – процеси організації та управління тренувальним процесом в спортивних залах.

Практична цінність. Розроблена веб-система дозволить спростити управління спортивним залом, автоматизувати процеси планування тренувань, розподілу ресурсів та взаємодії між тренерами та клієнтами. Це призведе до підвищення ефективності використання ресурсів, покращення якості обслуговування клієнтів та збільшення лояльності відвідувачів. Система також допоможе уникнути конфліктів у розкладі та оптимізувати роботу персоналу.

Інноваційність розробки. Розробка веб-орієнтованої системи є інноваційною через інтеграцію різних функцій керування графіками тренувань, автоматизацію бронювання ресурсів та забезпечення зручної взаємодії між користувачами. Система дозволить значно зменшити ручні процеси, підвищити ефективність роботи спортивних закладів та створити кращі умови для їх відвідувачів.

Апробація. Результати дипломної роботи були представлені на LIV Всеукраїнській науково-технічній конференції підрозділів Вінницького національного технічного університету у 2025 році на тему: «Розробка веб-

орієнтовної системи для організації роботи спортивного залу», науковий керівник: д.т.н., професор, В.М. Дубовой.

Публікація. Дубовой В.М., Карвацький В.Ю. Розробка веб-орієнтовної системи для організації роботи спортивного залу. *Матеріали LIV Всеукраїнської науково-технічної конференції підрозділів Вінницького національного технічного університету. Вінниця. ВНТУ, 2025.*

1. ДОСЛІДЖЕННЯ СТАНУ АВТОМАТИЗАЦІЇ І ПРОЄКТУВАННЯ ВЕБ-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ РОБОТИ СПОРТИВНОГО ЗАЛАЗУ

1.1. Характеристика об'єкту дослідження, аналіз стану автоматизації

Активний спосіб життя є ключем до збереження здоров'я та підтримки гарної фізичної форми протягом багатьох років. Заняття спортом включають виконання фізичних вправ та використання методичних підходів, які позитивно впливають на організм людини.

Сучасна спортивна інфраструктура пропонує широкий вибір можливостей як для любителів, так і для професійних спортсменів. Багато колишніх професіоналів продовжують свою кар'єру у спорті, працюючи тренерами для молоді. Різноманітні спортивні секції та спеціалізовані школи спрямовані на популяризацію спорту, починаючи від шахів і закінчуючи культуризмом. Основним завданням спортивних організацій є організація змагань як на всеукраїнському рівні, так і на рівні окремих закладів, наприклад, університетів.

Для тих, хто не займається спортом професійно, аматорські види спорту та активний відпочинок є доступними для кожного, хто хоче вести здоровий спосіб життя. Можна обрати будь-який напрямок фізичної активності: від екстремальних видів до щоденної ранкової зарядки. Спортсмени-аматори можуть відвідувати фітнес-клуби, тренажерні зали, басейни, займатися індивідуально з тренером або брати участь у групових заняттях.

Регулярні заняття спортом, як у спеціалізованих закладах (спортивних залах, басейнах), так і самотійно, сприяють формуванню самодисципліни, всебічному розвитку, зміцненню здоров'я та покращенню настрою. Фізична активність спрямована на підтримання життєвого тону, підвищення енергійності та забезпечення гарного самопочуття.

Створення веб-орієнтованих систем у сучасних умовах є складним і відповідальним процесом, який вимагає ретельного підходу та володіння багатьма технологіями. Інтернет-технології стрімко розвиваються, пропонуючи нові можливості для спрощення розробки, але успішне створення такої системи потребує компетенцій у кількох сферах: від розробки привабливого та функціонального дизайну, що зможе залучати користувачів, до якісного написання frontend і backend частин, що забезпечують зручність і зрозумілість у користуванні. Важливим етапом також є тестування, яке допомагає виявити недоліки і виправити помилки до офіційного запуску системи.

Сьогодні веб-орієнтовані системи стали важливим елементом сучасного бізнесу. Вони є не лише засобом представлення компанії в Інтернеті, але й важливим інструментом для успішного маркетингу. Якісно розроблена веб-система дозволяє не тільки привернути увагу клієнтів, але й забезпечити їх позитивний досвід користування, що сприяє формуванню гарної репутації компанії та її послуг.

Однак, попри значення вебзастосунків, деякі компанії все ще недостатньо приділяють увагу їх розробці, що може призвести до втрати клієнтів та погіршення репутації. Тому особливо важливо приділяти належну увагу кожному етапу розробки веб-орієнтованих систем, забезпечуючи їх якісне функціонування та задоволення потреб користувачів. Для успішного розвитку веб-орієнтованої системи важливо враховувати кілька ключових аспектів, які впливають на її ефективність і довговічність (рис. 1.1):



Рисунок 1.1 – Ключові аспекти, які впливають на ефективність і довговічність веб-орієнтованої системи

1. Адаптивність та мобільність. У сучасному світі, де більшість користувачів звертаються до Інтернету через мобільні пристрої, адаптивність системи є критично важливою. Веб-орієнтована система повинна коректно відображатися на різних типах пристроїв, будь то смартфони, планшети або десктопи, щоб користувачі могли отримати однаково зручний досвід взаємодії незалежно від платформи.

2. Безпека та захист даних. Безпека є основним фактором у розробці будь-якого веб застосунку. Для того щоб уникнути загроз, таких як витік даних або атаки на систему, необхідно впроваджувати сучасні методи захисту інформації. Використання протоколів шифрування, захищені сервери, регулярні оновлення безпеки та аудит коду повинні бути обов'язковими

елементами під час розробки.

3. Продуктивність та масштабованість. Ефективна веб-орієнтована система повинна швидко завантажуватися та працювати без збоїв, навіть при високих навантаженнях. Це забезпечується оптимізацією коду, мінімізацією кількості запитів до серверу, а також впровадженням системи кешування. Важливо також передбачити можливість масштабування системи в майбутньому, щоб вона могла обробляти більше користувачів або даних без втрати продуктивності.

4. Зручність користування (User Experience). Важливим фактором успіху будь-якої веб-системи є простота і зручність її використання. Інтерфейс має бути інтуїтивно зрозумілим для користувачів, з чіткою навігацією та мінімальними кроками для виконання основних завдань. Добре продуманий UX-дизайн дозволяє підвищити задоволеність користувачів, що, своєю чергою, сприяє утриманню клієнтів і збільшенню їхньої лояльності.

5. Інтеграція з іншими сервісами. Сучасна веб-орієнтована система повинна мати можливість легко інтегруватися з іншими сервісами та платформами, такими як системи оплати, CRM, соціальні мережі або інші бізнес-інструменти. Це дозволяє створювати більш комплексні рішення, що спрощують управління бізнесом і розширюють функціонал системи.

6. Аналітика та моніторинг. Важливо також впроваджувати механізми для збирання та аналізу даних про використання системи. Це дозволяє розробникам і адміністраторам відслідковувати поведінку користувачів, виявляти можливі проблеми, а також робити інформовані рішення щодо подальшого розвитку та покращення системи.

7. Підтримка та оновлення. Після запуску веб застосунку робота не завершується. Система потребує регулярних оновлень, які не тільки впроваджують нові функції, але й усувають вразливості та помилки. Також важливо забезпечити користувачів своєчасною технічною підтримкою, яка допоможе вирішувати проблеми та підтримувати роботу на високому рівні.

Окрім оновлень і технічної підтримки, важливо здійснювати постійний

моніторинг роботи веб-застосунку. Це включає аналіз продуктивності системи, виявлення можливих точок збоїв та оптимізацію ресурсів для забезпечення стабільної роботи. Використання сучасних інструментів моніторингу, таких як Google Analytics, New Relic або Prometheus, дозволяє отримувати дані в реальному часі та оперативно реагувати на проблеми.

Ще одним важливим аспектом є зворотний зв'язок від користувачів. Регулярний збір відгуків допомагає ідентифікувати слабкі місця у функціоналі чи дизайні системи. Це дозволяє не тільки покращувати користувацький досвід, але й зберігати лояльність клієнтів, адаптуючи систему до їхніх потреб.

Також слід приділяти увагу безпеці даних. Веб-застосунки, особливо ті, що працюють із конфіденційною інформацією (наприклад, особисті дані користувачів або платіжні транзакції), повинні відповідати сучасним стандартам кібербезпеки. Регулярне проведення аудитів безпеки, оновлення SSL-сертифікатів, впровадження багаторівневої авторизації та шифрування даних є обов'язковими кроками для збереження довіри користувачів.

Не менш важливим є масштабування системи. Зі збільшенням кількості користувачів або навантаження на сервери, система повинна бути готовою до масштабування. Це може включати перехід на потужніші сервери, використання хмарних технологій або впровадження мікросервісної архітектури для розподілу навантаження.

Нарешті, розвиток веб-застосунку має відповідати сучасним технологічним тенденціям. Регулярний аналіз ринку, впровадження нових технологій та вдосконалення дизайну допоможуть зберегти конкурентоспроможність системи та забезпечити її успіх у довгостроковій перспективі.

Отже, розробка веб-орієнтованої системи – це багатоетапний процес, що вимагає комплексного підходу. Успішно реалізована система здатна не лише забезпечити ефективну роботу бізнесу, але й підвищити його конкурентоспроможність, задовольняючи потреби користувачів і підвищуючи їхню довіру до бренду.

1.2. Опис предметної області

У сучасному світі, де кожен аспект людської діяльності все більше переходить на цифрові рейки, комп'ютеризація є важливим кроком для оптимізації будь-якого процесу. Незалежно від масштабу або складності об'єкта, який необхідно автоматизувати, ключовою метою залишається створення єдиної інформаційної структури, яка дозволить користувачам легко вносити дані, зручно обмінюватися інформацією та отримувати необхідні аналітичні відомості для прийняття рішень.

Предметна область веб-орієнтованої інформаційної системи спортивного залу охоплює всі процеси, які відбуваються в межах спортивного закладу. Це включає взаємодію між адміністрацією, тренерами та клієнтами (спортсменами, аматорами, відвідувачами), а також управління ресурсами залу. Система покликана забезпечити ефективну організацію та автоматизацію цих процесів для підвищення зручності користувачів та оптимізації роботи самого закладу.

У контексті роботи спортивного залу комп'ютеризація має на меті забезпечити зручне управління такими процесами, як планування тренувань, розподіл ресурсів, бронювання залів та відстеження участі клієнтів. Це не лише підвищує ефективність роботи персоналу, але й створює зручні умови для клієнтів, дозволяючи їм швидко знаходити вільний час для тренувань або змінювати свої графіки.

Основною метою розробки веб-орієнтованої системи для спортивного залу є створення єдиного інформаційного простору, що полегшить управління всіма аспектами його роботи. Система повинна забезпечувати:

- Автоматизацію планування графіків тренувань, бронювання ресурсів (зали, тренажери, персонал).
- Простий і зрозумілий інтерфейс для введення та обробки даних.
- Швидкий доступ до інформації, як для клієнтів, так і для персоналу.

- Оперативний контроль над усіма процесами залу та можливість швидких коригувань.

- Аналіз ефективності використання ресурсів та залучення клієнтів.

Для ефективного управління великою мережею спортивних клубів чи залів необхідно вести облік обладнання, реєстрацію клієнтів та персоналу. Якщо кожен зал веде паперовий облік, стає складно аналізувати динамічні дані всієї мережі загалом, адже цей процес забирає багато часу, навіть коли існують окремі сервіси для кожного клубу. Це також ускладнює роботу для клієнтів, які бажають відвідати одне з представництв мережі, але не мають доступу до актуальної інформації щодо наявного обладнання, персоналу чи власного прогресу в тренуваннях.

Запропонована система призначена для автоматизації обліку відвідувань мережі спортивних залів, зручного планування клієнтами тренувань, а також для отримання динамічної статистики по клієнтах, тренерах, залах та проведенням заняттям. Додатково, система дозволяє вести облік обладнання у спортивних залах, надаючи інформацію про його стан.

Відстеження потоку клієнтів у різні часові проміжки допоможе оптимізувати робочі години залів. Наприклад, якщо зросте попит на тренування перед роботою, система може запропонувати відкривати зал раніше. Зручність для клієнтів полягає в можливості записатися на тренування в обраному залі з обраним тренером (чи без нього) у зручний для них час, отримати інформацію про стан обладнання та власні досягнення, а також рекомендації щодо покращення ефективності тренувань.

Автоматизована система реєстрації також враховує потреби клієнтів у безпечному відвідуванні залів під час карантинних обмежень. Вона дозволяє залучати більше клієнтів, надсилаючи повідомлення про акції, знижки, та інші новини, а також регулювати потік відвідувачів. У випадку перевищення допустимої кількості людей, система може автоматично закрити запис на конкретний час, відповідно до вимог Міністерства охорони здоров'я.

Дані щодо відвідуваності залів допоможуть скласти бізнес-план для

розширення мережі та проведення маркетингових досліджень. Якщо в певному районі є підвищений попит, це може бути сигналом для відкриття нового залу або розширення існуючого.

Основні завдання системи:

- Облік відвідувань клієнтами мережі спортивних залів;
- Можливість створення, редагування та скасування записів на тренування;
- Перегляд зайнятості тренерів;
- Формування гнучкого розкладу для тренерів у різних залах;
- Аналіз попиту на окремі зали у мережі;
- Отримання статистики по клієнтам, тренерам, залам та проведеним заняттям;
- Облік спорядження та його стану у спортивних залах.

Предметною областю даної роботи є автоматизація процесу управління спортивним залом, зокрема планування відвідувань, бронювання обладнання та занять, а також моніторинг активності клієнтів. Це дозволить значно спростити та пришвидшити процес внесення нових подій, створення розкладу та перегляду графіків як для адміністрації, так і для відвідувачів.

Для того щоб повністю реалізувати функціонал веб-орієнтованої системи, було виділено основні бізнес-процеси, які система повинна автоматизувати:

1. Управління розкладом тренувань. Адміністратори зможуть зручно формувати графіки занять для тренерів і клієнтів, вносячи зміни в розклад у реальному часі.
2. Бронювання залів та обладнання. Клієнти зможуть легко бронювати зали для індивідуальних або групових занять, а також орендувати обладнання через веб-додаток.
3. Моніторинг активності клієнтів. Система дозволяє відстежувати відвідуваність залу, популярність окремих занять і ефективність роботи тренерів, що допоможе адміністрації оптимізувати процеси та ресурси.

4. Управління абонементом та оплатою. Система підтримуватиме облік абонементів, що дозволить клієнтам самостійно продовжувати свої контракти, оплачувати послуги онлайн і відстежувати залишки часу або тренувань.

5. Аналітичні звіти та планування. Адміністрація отримуватиме звіти про завантаженість залу, відвідуваність занять та фінансові показники, що дозволить приймати обґрунтовані рішення щодо покращення сервісу та оптимізації ресурсів.

Впровадження веб-орієнтованої системи управління спортивним залом матиме низку важливих переваг:

- Покращення якості обслуговування клієнтів. Швидкий доступ до інформації, можливість планування тренувань і бронювання в онлайн-режимі значно підвищить зручність для відвідувачів.
- Оптимізація роботи персоналу. Автоматизація процесів зменшить кількість ручної роботи для адміністрації та тренерів, дозволяючи їм зосередитися на наданні якісних послуг.
- Підвищення ефективності управління. Завдяки швидкому аналізу даних система допоможе керівництву приймати стратегічні рішення, покращувати роботу залу та оптимізувати використання ресурсів.

Для того, щоб повністю представити функціонал веб-орієнтованої системи для спортивного залу, виділено основні бізнес-процеси, які представлені на рис. 1.2.

Для того, щоб повністю представити функціонал веб-орієнтованої системи для спортивного залу, виділено основні бізнес-процеси, які представлені на рис. 1.2. Ці процеси поділяються на кілька ключових категорій, що охоплюють управління основними аспектами діяльності спортивного залу.

1. **Реєстрація та управління клієнтами:**

- Облік нових клієнтів, включаючи створення персонального профілю.

- Відстеження історії відвідувань та статистики використання послуг.

- Управління абонементами (придбання, продовження, замороження).

2. Управління персоналом:

- Планування графіків роботи тренерів та обслуговуючого персоналу.

- Контроль навантаження та продуктивності співробітників.

- Зворотний зв'язок від клієнтів щодо роботи тренерів.

3. Розклад тренувань та бронювання:

- Автоматизоване створення розкладу групових та індивідуальних занять.

- Надання можливості клієнтам бронювати місця через веб-інтерфейс.

- Управління списками очікування для популярних тренувань.

4. Фінансовий облік:

- Облік оплати за абонементи та додаткові послуги.

- Формування фінансових звітів для адміністрації.

- Нагадування про необхідність оплати через електронну пошту або SMS.

5. Модуль для тренерів:

- Інструменти для створення персональних програм тренувань для клієнтів.

- Відстеження прогресу клієнтів та фіксація результатів.

6. Інтеграція з додатковими сервісами:

- Підключення платіжних систем для зручності оплати послуг.

- Інтеграція з системами електронних замків для доступу до залу.

- Використання CRM-системи для маркетингових кампаній.

7. Аналітика та звітність:

- Збір та аналіз даних щодо популярності послуг, тренерів, часу

відвідувань.

- Побудова графіків та звітів для оптимізації процесів роботи.

Завдяки комплексному підходу, веб-орієнтована система сприяє автоматизації діяльності спортивного залу, підвищенню зручності для клієнтів і оптимізації внутрішніх процесів управління.



Рисунок 1.2 – Діаграма бізнес-процесів розроблюваного програмного продукту

Тепер детальніше розглянемо кожен з вище представлених бізнес-процесів. На рис. 1.3. зображено процес формування категорій графіку.



Рисунок 1.3 – Діаграма функцій процесу формування категорій графіку

Першим кроком для адміністратора є створення категорії, яку надалі буде додано до бази даних. Процес створення категорій має кілька ключових аспектів: Вибір типу – визначення виду активності, наприклад, заняття з

фізкультури для студентських груп, секції для спортивної кафедри або відвідування тренажерного залу. Опис характеристик – це можуть бути, наприклад, відповідальна особа за подію або кількість доступних місць у залі.

Характеристику бізнес-процесу створення категорій адміністратором детально подано у табл. 1.1.

Таблиця 1.1 – Характеристика бізнес-процесу формування категорій

Етап	Дія	Опис	Відповідальний
Створення категорії	Введення назви категорії	Адміністратор створює нову категорію для подальшого використання в системі	Адміністратор
Вибір типу категорії	Вибір виду активності	Визначається тип події: фізкультура, секції, тренажерний зал тощо	Адміністратор
Опис характеристик	Додавання інформації про подію	Вказуються характеристики: відповідальний за подію, кількість місць, час тощо	Адміністратор
Перевірка та збереження	Перевірка введеної інформації	Перевіряються всі дані перед збереженням у базу даних	Адміністратор
Додавання до бази даних	Збереження категорії у системі	Категорія додається до бази даних та стає доступною для використання	Система (після підтвердження)
Перегляд і редагування	Перегляд створеної категорії та можливість її редагування	Категорію можна переглянути або внести до неї зміни у разі необхідності	Адміністратор

На рис. 1.4. представлено діаграму функцій процесу формування графіку відвідувань, а саме виконання завдання.

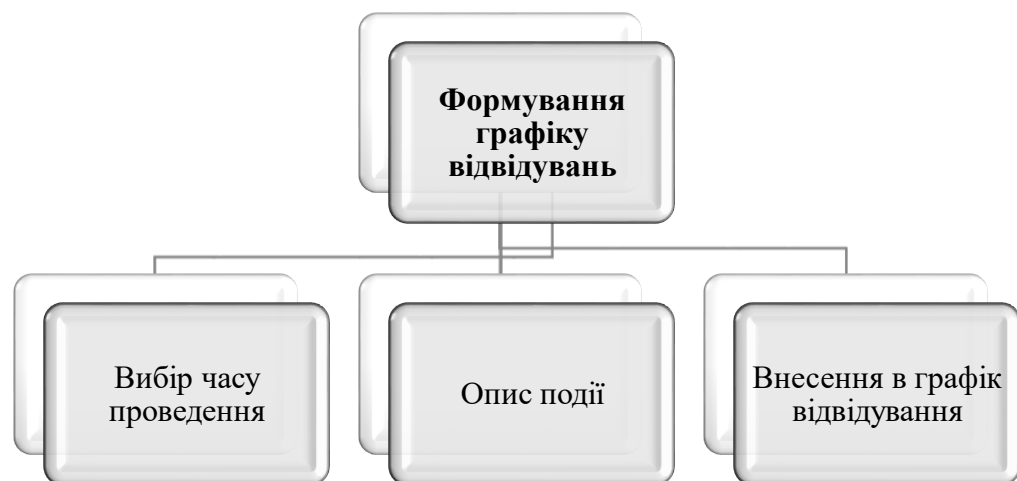


Рисунок 1.4 – Діаграма функцій процесу формування графіку відвідувань

Наступним кроком адміністратора є додавання записів до категорій для відвідувань та їхнє занесення до бази даних. Це ключовий процес, оскільки саме тут формується графік відвідувань. Формування графіку включає такі важливі етапи:

- Вибір часу проведення – визначення часу та періодичності проведення подій.
- Опис події – вказання місця проведення та відповідальних осіб, наприклад, тренерів для спортивних секцій.
- Додавання в графік відвідувань – процес занесення інформації про подію в базу даних.

Характеристику бізнес-процесу формування графіку відвідувань адміністратором наведено у табл.1.2.

Таблиця 1.2 – Характеристика бізнес-процесу формування графіку відвідувань

Етап	Дія	Опис	Відповідальний
Вибір категорії	Вибір категорії події	Адміністратор обирає категорію події для якої створюється графік	Адміністратор
Вибір часу проведення	Визначення дати, часу та періодичності	Встановлюється точний час та дата проведення події, а також періодичність (щодня, щотижня)	Адміністратор
Опис події	Внесення інформації про подію	Включає місце проведення події, відповідальних осіб (тренерів, організаторів)	Адміністратор
Внесення в базу даних	Занесення інформації до бази даних	Інформація про подію фіксується у базі даних для подальшого використання	Система (після підтвердження)
Перегляд та редагування	Перегляд або внесення змін до графіку	Адміністратор може переглядати та редагувати графік подій у разі потреби	Адміністратор

На рис. 1.5 наведемо діаграму функцій процесу перегляду графіку відвідувань спортивного залу.



Рисунок 1.5 – Діаграма функцій процесу перегляду графіку відвідувань

Автоматизація роботи спортивного залу за допомогою веб-орієнтованої системи забезпечує актуальність бази даних і надає можливість отримання достовірної статистичної інформації. Це сприяє значному прискоренню процесів створення графіків відвідувань і забезпечує постійний оперативний контроль над управлінням роботою залу.

Система пропонує зручний і інтуїтивний інтерфейс, який дозволяє співробітникам легко працювати з графіками та іншими даними. Основними користувачами системи є працівники, відповідальні за планування тренувань та управління ресурсами залу. Автоматизація усуває необхідність у ручному заповненні документів, мінімізує ризик плутанини в паперовій документації та значно знижує кількість помилок.

Система пропонує зручний і інтуїтивно зрозумілий інтерфейс, що дозволяє співробітникам ефективно управляти графіками тренувань та іншими даними. Основними користувачами цієї системи є працівники, які відповідають за планування тренувань та управління ресурсами спортивного залу, зокрема адміністративний персонал та тренери.

Автоматизація процесів значно спрощує роботу, усуваючи необхідність вручну заповнювати документи та вести паперову документацію. Це не тільки мінімізує ризик виникнення плутанини або помилок через людський фактор, але й дозволяє зекономити час, який раніше витрачався на обробку даних. Завдяки системі, усі зміни та оновлення вносяться оперативно, і всі

користувачі мають доступ до актуальної інформації в режимі реального часу.

Це підвищує точність і ефективність роботи залу, дозволяє краще організувати використання ресурсів, полегшує контроль над доступністю тренерів та обладнання, а також сприяє покращенню обслуговування клієнтів. Зокрема, зменшується кількість помилок при бронюванні тренувань або розподілі часу для занять, що є критично важливим для підтримки високого рівня сервісу.

Таким чином, створення веб-орієнтованої системи для спортивного залу дозволить суттєво оптимізувати адміністративні процеси, автоматизувати рутинні операції та підвищити ефективність використання ресурсів. Це сприятиме не лише покращенню якості обслуговування клієнтів, але й зміцненню довіри та лояльності відвідувачів.

Окрім цього, така система стане важливим кроком у популяризації активного способу життя, адже завдяки зручності й доступності послуг більше людей матиме змогу організовано займатися спортом. Це, у свою чергу, сприятиме розвитку спортивної інфраструктури, підвищенню обізнаності про важливість фізичної активності та загальному зміцненню здоров'я населення.

Розробка подібної системи не лише вирішить поточні проблеми управління, а й забезпечить перспективи для розширення бізнесу, адаптації до змін у ринку та впровадження інноваційних підходів у спортивній індустрії.

1.3. Огляд і аналіз існуючих аналогів

На сьогодні існує чимало програмних рішень, які дозволяють здійснювати планування різних подій. Однак, програм, що спеціалізуються на плануванні відвідувань спортивного залу, значно менше.

Одним із таких продуктів є програмне забезпечення, яке інтегрується з інфраструктурою спортивного залу та дозволяє автоматизувати ключові процеси його функціонування. Це включає такі можливості, як онлайн-запис на тренування, управління графіком тренувань, облік відвідуваності, надання

користувачам інформації про доступні послуги, тренерів і вільний час для занять. Використання веб-орієнтованих систем у спортивних закладах надає клієнтам можливість швидко отримати доступ до необхідної інформації та забронювати зручний для себе час.

Крім того, такі системи можуть надавати додаткові можливості для управління фінансовою складовою залу, зокрема автоматизований облік платежів за абонементи, додаткові послуги та оренду обладнання. За допомогою сучасних технологій, адміністратори можуть оперативно переглядати звіти про фінансові операції, рівень заповнюваності залу та статистику по тренуваннях.

Застосування веб-технологій у розробці таких систем значно підвищує гнучкість і зручність використання як для клієнтів, так і для працівників залу. Користувачі можуть здійснювати бронювання в будь-який час і з будь-якого пристрою, підключеного до Інтернету. Це забезпечує не лише кращий користувацький досвід, але й сприяє збільшенню кількості постійних клієнтів та оптимізації ресурсів закладу. Нижче наведено огляд найбільш поширених аналогів:

Система **Mindbody** є однією з провідних платформ для управління фітнес-залами, студіями й іншими спортивними та оздоровчими установами. Вона пропонує великий набір функцій, який включає в себе все, від управління клієнтською базою до аналітики бізнесу та фінансового обліку. Водночас, як і будь-яке програмне забезпечення, Mindbody має свої переваги та недоліки, які важливо враховувати при виборі цієї платформи (табл.1.3).

Таблиця 1.3 – Переваги та недоліки системи Mindbody

Переваги	Недоліки
<i>Мобільний додаток для користувачів і адміністраторів. Mindbody пропонує зручні мобільні додатки для клієнтів і адміністраторів, що дозволяє легко керувати записами та фінансами прямо з мобільного пристрою.</i>	<i>Висока вартість для малих спортивних залів. Платформа може бути досить дорогою для невеликих фітнес-залів або студій через високі щомісячні тарифи, що включають багато функцій, які можуть бути надмірними для невеликих підприємств.</i>

<i>Потужні інструменти для управління клієнтами та аналітики.</i> Система пропонує детальну аналітику відвідувань, продажів і фінансових показників, що дозволяє глибоко аналізувати ефективність роботи закладу.	<i>Велика кількість функцій може ускладнити інтеграцію.</i> Через великий набір функцій інтеграція з іншими системами може бути складною і вимагати значних часових та фінансових ресурсів.
<i>Інтеграція з різними платіжними системами.</i> Mindbody підтримує інтеграцію з популярними платіжними системами, що полегшує процеси оплати клієнтами за послуги.	<i>Складний інтерфейс для новачків.</i> Через розгалужену функціональність система може здатися складною для нових користувачів, що потребує додаткового часу на навчання.
<i>Підтримка різних бізнес-моделей.</i> Платформа підходить для широкого кола підприємств, таких як фітнес-зали, студії йоги, масажні салони та спа-центри, що робить її універсальним інструментом.	<i>Обмежена кастомізація для малих закладів.</i> Невеликі фітнес-центри можуть відчувати брак можливостей налаштування під конкретні потреби, що може обмежити їх у використанні певних функцій.
<i>Платформа надає розширені звіти та аналітичні дані.</i> Система пропонує детальні звіти, які допомагають власникам бізнесу приймати обґрунтовані рішення на основі статистики та аналітики.	<i>Високі витрати на навчання персоналу.</i> Через складність інтерфейсу і багатий функціонал, персонал потребує додаткового навчання, що може вимагати часу та додаткових і

На рис. 1.6 представлено головну сторінку даної веб-орієнтованої системи.

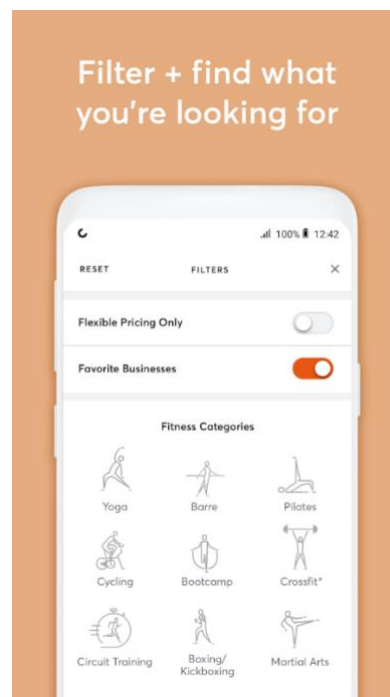


Рисунок 1.6 – Головна сторінка програмного забезпечення для спортивного залу Mindbody

Джерело: [25]

Mindbody є потужним та універсальним інструментом для автоматизації роботи фітнес-залів і студій, але її широкі можливості можуть виявитися надмірними для малих закладів. Платформа особливо підходить для середніх та великих фітнес-центрів, які прагнуть ефективно керувати клієнтською базою, фінансовими операціями та графіком занять. Водночас, вартість і складність можуть стати перешкодою для малого бізнесу.

GloFox - програмне забезпечення створено спеціально для фітнес-клубів і студій, і орієнтоване на автоматизацію записів на заняття, продаж абонементів та ведення клієнтської бази. GloFox пропонує зручний інтерфейс для як тренерів, так і клієнтів, а також мобільний додаток для запису на заняття. Переваги та недоліки представлені у табл.1.4, головну сторінку програмного забезпечення на рис. 1.7.

Таблиця 1.4 – Переваги та недоліки GloFox

Переваги	Недоліки
<i>Простий у використанні інтерфейс.</i> GloFox пропонує інтуїтивно зрозумілий інтерфейс, що спрощує навчання нових користувачів і прискорює процес впровадження системи.	<i>Обмежена функціональність для великих спортивних залів.</i> Платформа може не мати всіх необхідних функцій для великих фітнес-центрів, що потребують складніших рішень.
<i>Мобільний додаток для клієнтів.</i> Наявність мобільного додатку дозволяє клієнтам легко записуватися на заняття, переглядати свій графік та управляти абонементом.	<i>Вартість підписки може бути високою для нових спортивних центрів.</i> Для нових або малих залів вартість підписки може бути суттєвим фінансовим навантаженням.
<i>Інтеграція з CRM для управління клієнтами.</i> GloFox дозволяє інтегруватися з системами управління клієнтами, що полегшує облік та аналітику.	<i>Можливі обмеження щодо кастомізації.</i> Наявні шаблони можуть обмежувати можливості налаштування під специфічні потреби закладу.
<i>Гнучкі можливості бронювання занять.</i> Користувачі можуть легко бронювати заняття в режимі реального часу, що підвищує зручність і ефективність процесу.	<i>Може бути недостатньо маркетингових інструментів.</i> Платформа не надає достатньо вбудованих маркетингових інструментів для просування послуг.
<i>Зручні звіти та аналітика.</i> GloFox надає зручні звіти, що дозволяють власникам бізнесу відстежувати фінансові результати та продуктивність залу.	<i>Обмежена підтримка для користувачів.</i> Клієнти можуть зіштовхнутися з проблемами при отриманні технічної підтримки, що може затримувати вирішення питань.

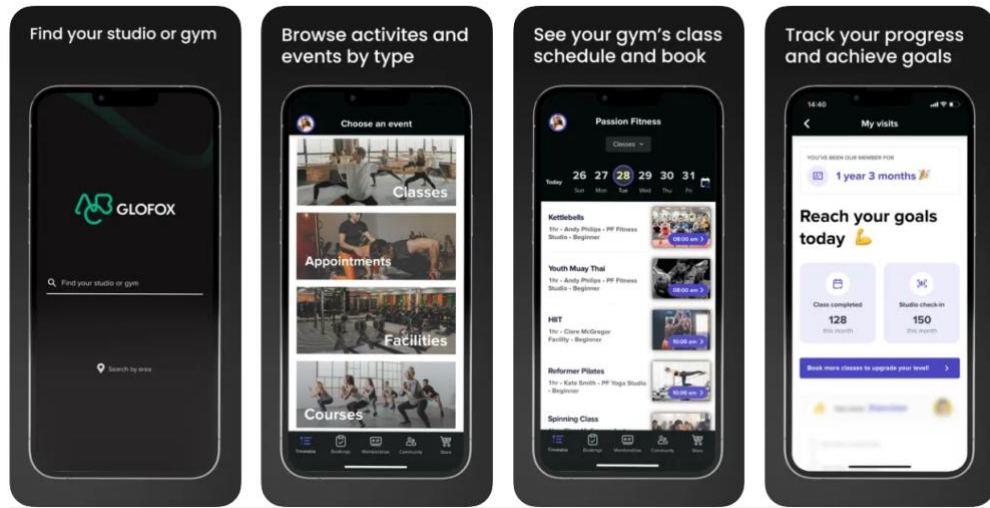


Рисунок 1.7 – Головна сторінка GloFox

Джерело: [8]

GloFox є зручним і простим у використанні рішенням для малих і середніх фітнес-залів, пропонуючи достатню кількість функцій для управління клієнтами та заняттями. Однак, для великих спортивних залів або тих, хто потребує більш складних функцій і кастомізації, GloFox може виявитися обмеженим. Власники залів повинні оцінити свої потреби та ресурси перед вибором цієї платформи.

GymMaster - система автоматизує управління фітнес-центрами, включаючи бронювання занять, контроль за доступом до залу, управління абонементом та інтеграцію з системами оплати. GymMaster орієнтований на малі та середні фітнес-зали, пропонуючи оптимальний набір функцій для повсякденної роботи. Переваги та недоліки представлені у табл. 1.5 а вигляд головного меню на рис. 1.8 відповідно.

Таблиця 1.5 – Переваги та недоліки системи GymMaster

Переваги	Недоліки
<i>Інтеграція з системами доступу.</i> GymMaster дозволяє інтегруватися з системами контролю доступу, що підвищує безпеку та управлінські можливості.	<i>Менший набір функцій порівняно з великими системами.</i> Платформа може не пропонувати всі можливості, необхідні великим фітнес-центрам.
<i>Зручний інтерфейс для адміністраторів.</i> Інтуїтивно зрозумілий інтерфейс робить його легким у використанні для адміністраторів, що сприяє швидкому впровадженню.	<i>Обмежені можливості кастомізації.</i> Можливості налаштування можуть бути обмеженими для специфічних потреб деяких залів.

<i>Підтримка різних бізнес-моделей.</i> GymMaster підходить для різних типів спортивних залів, включаючи фітнес-центри, студії йога та інші спортивні установи.	<i>Деякі функції можуть бути зайвими для малих залів.</i> Для невеликих залів наявні функції можуть виявитися зайвими або невикористаними.
<i>Управління абонементом та платіжними системами.</i> Система дозволяє легко управляти абонементом, що спрощує фінансові процеси для адміністраторів.	<i>Вартість для малих і середніх залів.</i> Вартість підписки може бути високою для малих і середніх залів, які тільки починають свою діяльність.
<i>Гнучкі можливості звітності.</i> GymMaster надає можливість створення звітів, що дозволяє власникам залів контролювати фінанси та ефективність роботи.	<i>Можливі затримки в технічній підтримці</i> Клієнти можуть стикатися з затримками в отриманні технічної підтримки при вирішенні питань.

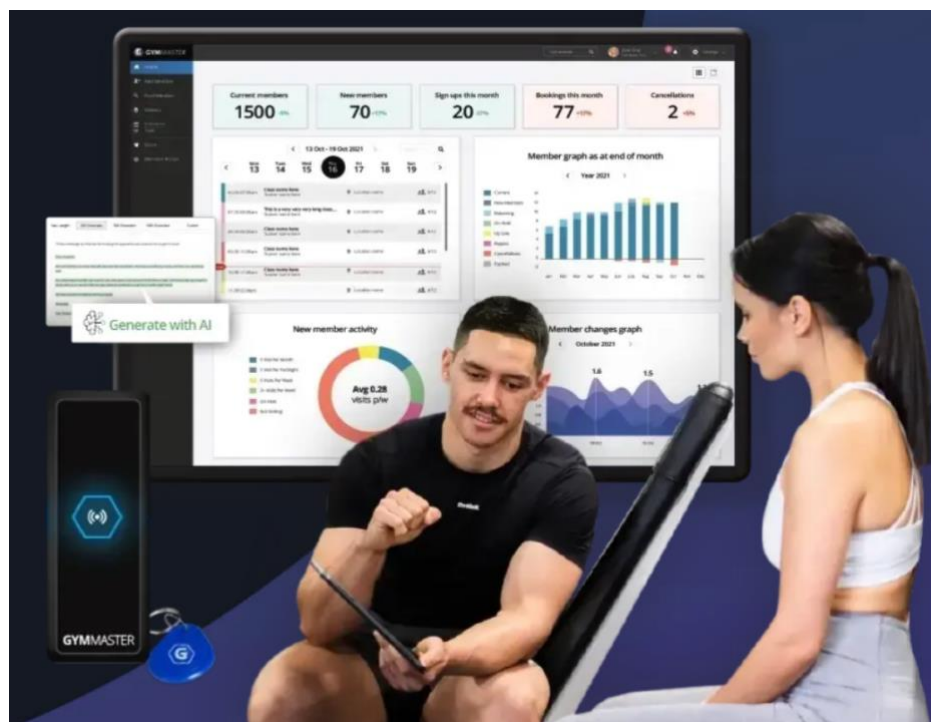


Рисунок 1.8 – Головна сторінка GymMaster

Джерело: [8]

GymMaster є надійним рішенням для управління спортивними залами, пропонуючи зручний інтерфейс та інтеграцію з системами контролю доступу. Однак, платформа може бути менш придатною для малих залів через вартість і складність деяких функцій. Власники спортивних закладів повинні ретельно оцінити свої потреби та ресурси перед вибором GymMaster як системи управління.

Wodify - платформа спочатку була створена для CrossFit-студій, однак згодом розширила свої функціональні можливості для інших видів фітнесу.

Wodify дозволяє керувати клієнтами, розкладом занять, а також забезпечує можливість проведення онлайн-змагань та відстеження результатів тренувань.

У табл. 1.6. показано переваги та недоліки даного програмного забезпечення а на рис. 1.9. головну сторінку застосунку.

Таблиця 1.6 – Переваги та недоліки системи Wodify

Переваги	Недоліки
<i>Спеціалізація на CrossFit.</i> Wodify спочатку була розроблена для студій CrossFit, що робить її ідеальною для цього типу залів, з потужними функціями для відстеження результатів тренувань.	<i>Обмежена функціональність для універсальних залів.</i> Платформа може бути менш ефективною для загальних фітнес-залів, які не спеціалізуються на CrossFit.
<i>Можливості для проведення онлайн-змагань.</i> Wodify дозволяє організовувати онлайн-змагання та події, що стимулює залученість клієнтів та сприяє розвитку спільноти.	<i>Складний інтерфейс для нових користувачів.</i> Нові користувачі можуть зіткнутися з труднощами при освоєнні інтерфейсу, що потребує додаткового часу на навчання.
<i>Відстеження результатів та прогресу.</i> Система дозволяє клієнтам відстежувати свій прогрес у тренуваннях, що підвищує мотивацію та зацікавленість у заняттях.	<i>Висока вартість підписки.</i> Для малих студій вартість підписки може бути значним фінансовим навантаженням, особливо якщо вони тільки починають.
<i>Зручні функції управління клієнтами.</i> Wodify надає можливості для ефективного управління клієнтською базою та комунікації з ними, що підвищує рівень обслуговування.	<i>Обмежені можливості кастомізації.</i> Можливості налаштування можуть бути обмежені, що може не задовольнити специфічні потреби деяких залів.
<i>Аналітика та звітність.</i> Платформа надає детальні аналітичні дані про відвідуваність, продажі та ефективність тренувань, що допомагає приймати обґрунтовані бізнес-рішення.	<i>Можливі проблеми з підтримкою користувачів.</i> Користувачі можуть зітхнути з затримками в отриманні технічної підтримки при виникненні проблем.

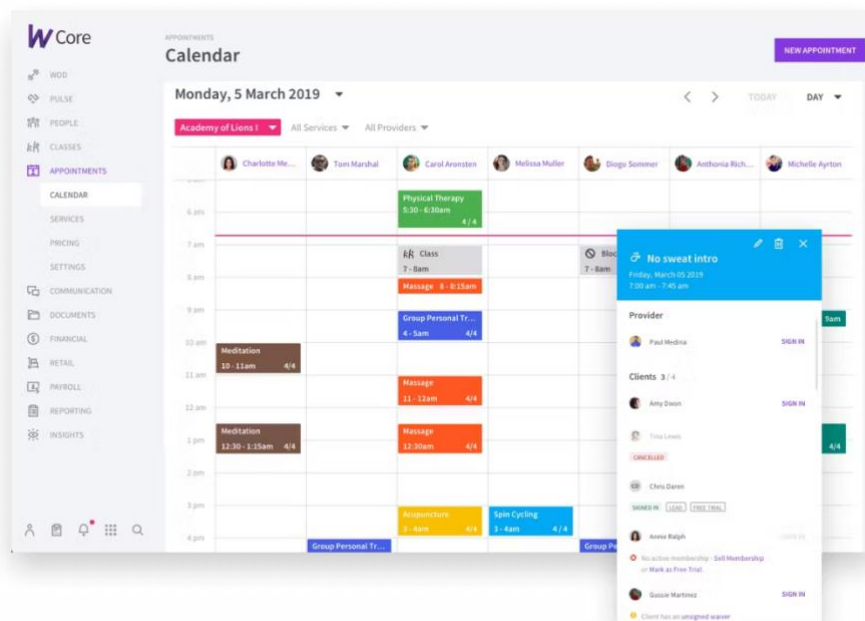


Рисунок 1.9 – Веб-застосунок Wodify

Джерело: [8]

Wodify є потужним інструментом для управління студіями CrossFit, пропонуючи специфічні функції, які задовольняють потреби цієї ніші. Однак, для універсальних фітнес-залів або малих студій, які не спеціалізуються на CrossFit, платформа може виявитися менш ефективною через обмежені можливості кастомізації та високу вартість.

Perfect Gym - комплексна система для управління великими спортивними залами та мережами фітнес-клубів. Вона забезпечує широкий спектр функцій, включаючи управління клієнтською базою, відстеження активності клієнтів, інтеграцію з CRM-системами та маркетинговими інструментами. Переваги та недоліки використання комплексної системи представлені у табл. 1.7.

Таблиця 1.7 – Переваги та недоліки системи Perfect Gym

Переваги	Недоліки
<i>Універсальність платформи.</i> Perfect Gym підходить для різних типів фітнес-центрів, включаючи тренажерні зали, студії йоги та комерційні спортзали.	<i>Складний процес налаштування.</i> Інтеграція та налаштування системи можуть займати багато часу та ресурсів, особливо для нових користувачів.
<i>Розширені функції аналітики.</i> Система пропонує потужні інструменти аналітики та звітності, що дозволяє власникам контролювати ефективність бізнесу.	<i>Вартість підписки.</i> Вартість користування платформою може бути високою для малих та середніх залів, що тільки починають свою діяльність.
<i>Інтеграція з платіжними системами.</i> Perfect Gym підтримує різноманітні платіжні системи, що полегшує фінансові транзакції і управління абонементом.	<i>Можливі проблеми з технічною підтримкою.</i> Користувачі можуть стикатися з затримками у відповіді служби підтримки при виникненні технічних питань.
<i>Мобільний додаток для клієнтів.</i> Наявність мобільного додатку дозволяє клієнтам легко записуватися на заняття, перевіряти графік та управляти абонементом.	<i>Складний інтерфейс для нових користувачів.</i> Інтерфейс може бути заплутаним для нових користувачів, що може вимагати додаткового навчання.
<i>Гнучкі можливості кастомізації.</i> Система дозволяє налаштовувати функції під специфічні потреби залу, що підвищує зручність використання.	<i>Обмеження в інтеграції з деякими сторонніми системами.</i> Можуть виникати проблеми з інтеграцією з деякими сторонніми платформами, що обмежує гнучкість у використанні.

На рис. 1.10. можна наочно спостерігати головну сторінку веб застосунку Perfect Gym.

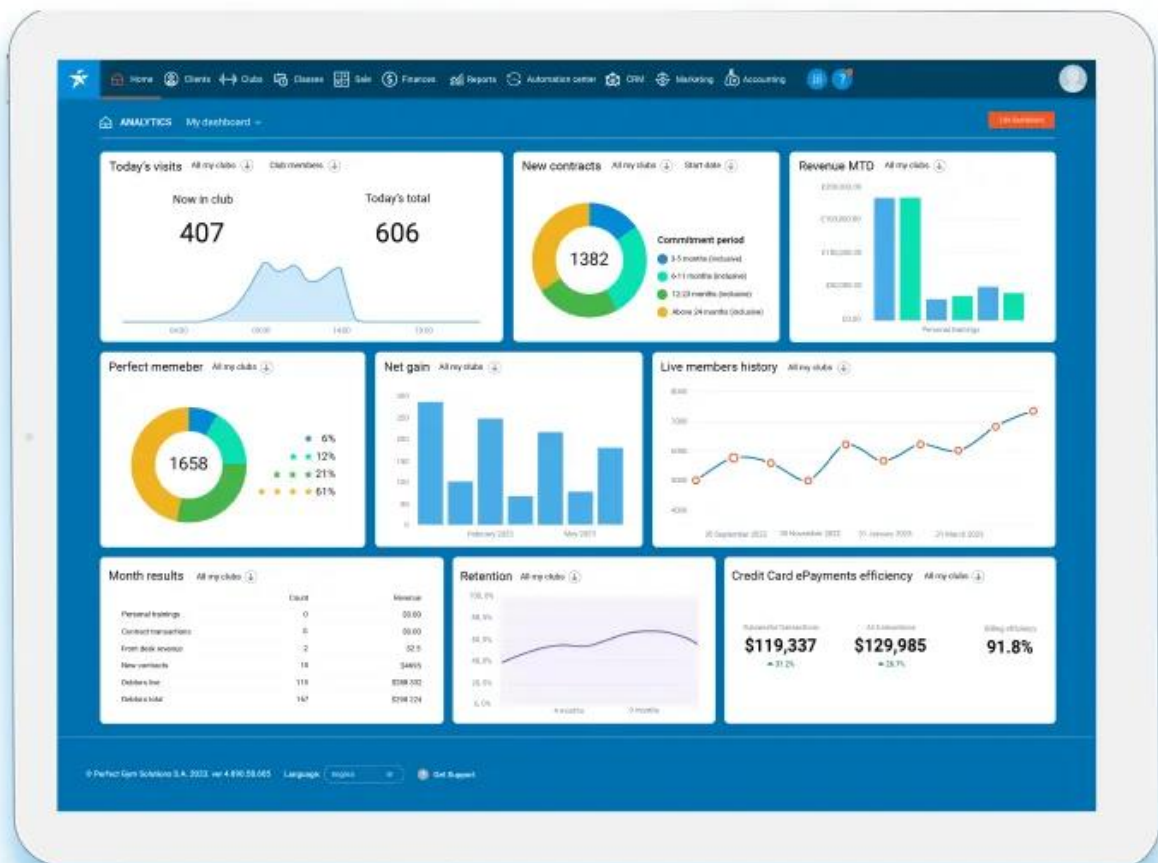


Рисунок 1.10 – Веб-застосунок Perfect Gym

Джерело: [23]

Perfect Gym є потужним інструментом для управління фітнес-залами, пропонуючи широкий набір функцій для аналізу, управління клієнтами та фінансовими транзакціями. Однак, процес налаштування може бути складним, а вартість підписки може стати перешкодою для малих закладів. Власники спортивних установ повинні ретельно оцінити свої потреби та ресурси перед вибором цієї платформи.

Вивчивши та проаналізувавши наявні аналоги, можна дійти висновку, що перелічені вище програмні системи є ефективними для організації відвідувань приватних спортивних залів. Кожен з продуктів працює досить стабільно.

1.4. Специфікація вимог до системи

Специфікація вимог до веб-орієнтованої системи для організації спортивного залу - це докладний опис функціонування системи, що розробляється. Вона містить різноманітні прецеденти, які ілюструють усі взаємодії користувачів з програмним забезпеченням. Прецеденти також відомі як функціональні вимоги. Крім них, у специфікації присутні й нефункціональні вимоги, які накладають обмеження на проект чи його реалізацію, не пов'язані з основними функціями системи. Специфікація вимог також містить глосарій проекту та опис варіантів використання. У таблиці 1.8 наведено глосарій основних термінів, що використовуються під час розробки системи планування відвідувань спортзалу.

Таблиця 1.8 – Глосарій основних термінів

Термін	Опис
Клієнт	Особа, яка користується послугами спортивного залу, має абонемент або відвідує заняття.
Адміністратор	Користувач, відповідальний за управління системою, налаштування параметрів та облік клієнтів.
Тренер	Спеціаліст, який проводить заняття та тренування, а також контролює прогрес клієнтів.
Функціональні вимоги	Вимоги, що описують конкретні функції, які система повинна виконувати.
Нефункціональні вимоги	Вимоги, які накладають обмеження на систему, що не стосуються її основних функцій (продуктивність, безпека тощо).
Прецедент	Опис ситуації, в якій користувач взаємодіє з системою для досягнення певної мети.
Варіанти використання	Описує, як користувачі можуть взаємодіяти із системою для виконання певних завдань.
Інтерфейс	Частина системи, з якою користувач взаємодіє, включаючи графічні елементи та елементи управління.
Аутентифікація	Процес перевірки особи користувача при вході в систему для підтвердження її прав.
Бронювання	Процес запису на заняття або тренування через систему.
Аналітика	Процес збору та аналізу даних для оцінки ефективності роботи системи та прийняття бізнес-рішень.

Далі, на рис. 1.11 представлено діаграму варіантів використання для акторів системи (замовник, адміністратор, користувач).

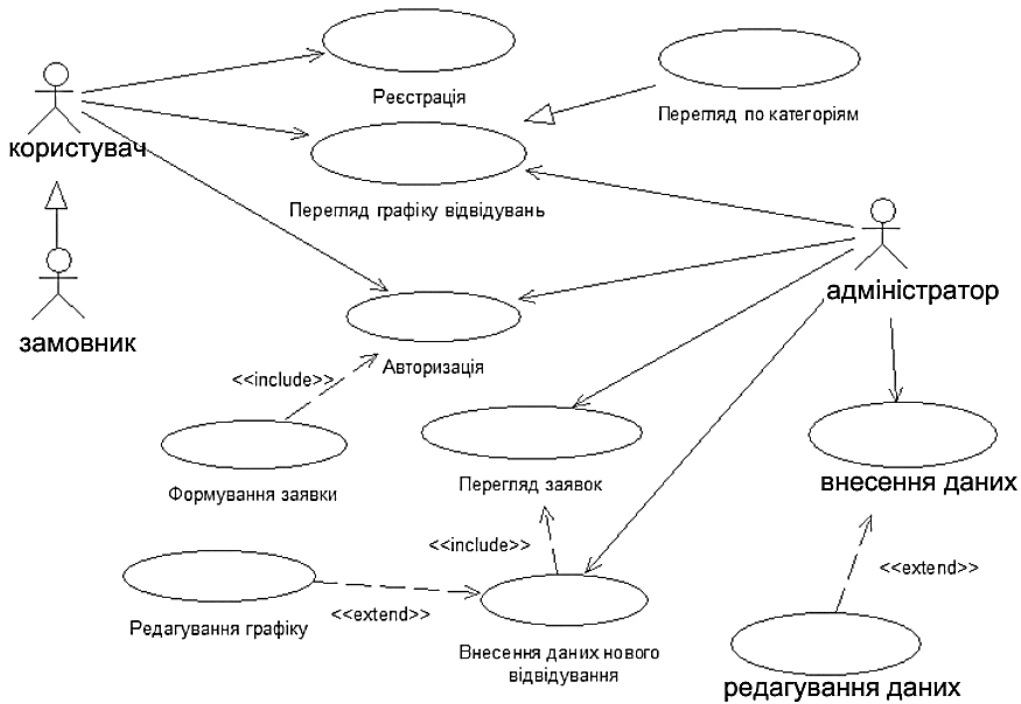


Рисунок 1.11 – Діаграма варіантів використання

Наступним етапом у визначенні специфікацій вимог до веб-орієнтованої системи є опис варіантів використання. Варіанти використання представлені у наступних таблицях. Опис варіанту використання «Реєстрація» наведено у таблиці 1.9.

Таблиця 1.9 – Опис варіанту використання «Реєстрація»

Контекст використання	Реєстрація
Дійові особи	Не зареєстрований користувач
Передумова	Користувач знаходиться на сторінці реєстрації
Тригер	Відкрита сторінка для реєстрації
Сценарій	1.Заповнюємо поле «Введіть логін»; 2.Заповнюємо поле «Введіть пароль»; 3.Заповнюємо поле «Введіть пароль ще раз»; 4.Натискаємо кнопку «Зареєструвати»
Пост-умова	Користувач зареєстрований

Варіант використання «Авторизація» подано у табл. 1.10.

Таблиця 1.10 – Опис варіанту використання «Авторизація»

Контекст використання	Авторизація
Дійові особи	Не авторизований користувач, адміністратор
Передумова	Користувач знаходиться на сторінці авторизації
Тригер	Відкрита сторінка для авторизації
Сценарій	1.Заповнюємо поле «Введіть логін»; 2.Заповнюємо поле «Введіть пароль»; 3.Натискаємо кнопку «Увійти»
Пост-умова	Користувач, адміністратор авторизований

Варіант використання «Перегляд графіку відвідувань» подано у табл. 1.11.

Таблиця 1.11 – Опис варіанту використання «Перегляд графіку відвідувань»

Контекст використання	Перегляд графіку
Дійові особи	Користувач, адміністратор, гість
Передумова	Користувач знаходиться на головній сторінці
Тригер	Відкрита головна сторінка
Сценарій	I. Клік на вкладку календаря
Пост-умова	Відкрита головна сторінка

Варіант використання «Внесення даних відвідування» подано у табл. 1.12.

Таблиця 1.12 – Опис варіанту використання «Внесення даних нового відвідування»

Контекст використання	Додавання відвідування
Дійові особи	Адміністратор
Передумова	Авторизація адміністратора
Тригер	Відкрита особиста сторінка
Сценарій	1. Обираємо дату; 2. Обираємо тип відвідування; 3. Обираємо час; 4. Якщо не суперечить актуальному розкладу, - натискаємо оформити.
Пост-умова	Збереження даних в БД

Варіант використання «Формування заявки» подано у табл. 1.13.

Таблиця 1.13 – Опис варіанту використання «Формування заявки»

Контекст використання	Формування заявки
Дійові особи	Користувач
Передумова	Авторизація користувача
Тригер	Відкрита особиста сторінка
Сценарій	Обираємо тип відвідування; Обираємо час; Обираємо тривалість; Натискаємо кнопку «Подати заявку»
Пост-умова	Збереження даних в БД

Варіант використання «Внесення студентських даних» подано у таблиці 1.14.

Таблиця 1.14 – Опис варіанту використання «Внесення студентських даних»

Контекст використання	Внесення студентських даних
Дійові особи	Адміністратор
Передумова	Авторизація Адміністратора
Тригер	Відкрита сторінка студентських даних
Сценарій	Додаємо необхідний факультет; Додаємо необхідний групу; Додаємо необхідного студента; Вносимо дані про студента.
Пост-умова	Збереження даних в БД

Варіант використання «Перегляд по категоріям» представлено у табл. 1.15.

Таблиця 1.16 – Опис варіанту використання «Перегляд по категоріям»

Контекст використання	Перегляд по категоріям
Дійові особи	Користувач, адміністратор
Передумова	Користувач знаходиться на головній сторінці
Тригер	Відкрита особиста сторінка
Сценарій	1. Обраємо категорію відвідування
Пост-умова	Відкрита головна сторінка

Варіант використання «Перегляд заявок» представлено у таблиці 1.16.

Таблиця 1.16 – Опис варіанту використання «Перегляд заявок»

Контекст використання	Перегляд заявок
Дійові особи	Адміністратор
Передумова	Авторизація адміністратора
Тригер	Відкрита особиста сторінка

Сценарій	Обираємо заявку Натискаємо на неї Переглядаємо зміст заявки
Пост-умова	Відкриття варіанту використання «Внесення даних нового відвідування»

Для демонстрації варіантів використання було створено розкадровку – прототипи системи, які відображають основну функціональність та вимоги до неї (рис. 1.12- 1.8).

Введіть логін

Введіть пароль

Введіть пароль ще раз

Введіть логін

Введіть пароль

 Якщо ви не зареєстровані зареєструйтесь

Рисунок 1.12 – Вікно реєстарції та авторизації

Графік занять в спортзалі

графік занять в спортзалі у вибраний день

жовтень 2024 р.

Пн	Вт	Ср	Чт	Пт	Сб	Нд
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3
4	5	6	7	8	9	10

☐ Сьогодні: 0810.2024

Рисунок 1.13 – Перегляд розкладу відвідувань

Додавання відвідувань

Тип відвідування

День тижня

Час

Графік відвідувань

графік відвідування конкретного

Рисунок 1.14 – Адміністраторський обліковий запис: введення даних для нового візиту

The screenshot shows a web form titled 'Формування заявки' (Formulation of application). On the left, there are three dropdown menus: 'Тип відвідування' (Type of visit), 'День тижня' (Day of the week), and 'Час' (Time). In the center, there is a button labeled 'Подати заявку' (Submit application). On the right, there is a large text area labeled 'Текст сформованої заявки' (Text of the formulated application).

Рисунок 1.15 – Користувацький обліковий запис: формування заявки

The screenshot shows a web form for entering student data. It is divided into several sections:

- Студент (Student):** Includes fields for 'Прізвище' (Surname), 'Ім'я' (Name), 'По-батькові' (Patronymic), 'Дата народження' (Date of birth) with a calendar icon, 'Телефон' (Phone), and 'Стать' (Gender) dropdown.
- Паспорт (Passport):** Includes fields for 'Серія' (Series) and 'Номер' (Number).
- Адреса (Address):** Includes fields for 'Країна' (Country), 'Область' (Region), 'Район' (District), 'Місто' (City), 'Вулиця' (Street), 'Будинок' (House), and 'Квартира' (Apartment).
- Навчання (Education):** Includes fields for 'Факультет' (Faculty) and 'Група' (Group).

 At the bottom right, there is a button labeled 'Зберегти' (Save).

Рисунок 1.16 – Внесення даних відвідувача

Результати пошуку

Відображення результатів пошуку

◀ жовтень 2024 р. ▶

Пн	Вт	Ср	Чт	Пт	Сб	Нд
30	31	1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	1	2	3
4	5	6	7	8	9	10

☐ Сьогодні: 08.10.2024

Пошук по фільтру

фільтр

Тип відвідування

Дата та час

День тижня

Час

Пошук

Рисунок 1.17 – Адміністраторський обліковий запис: перегляд за критеріями

Отримані заявки

Список Заявок

Текст заявки

Текст вибраної заявки

Внесення даних нового відвідування

Рисунок 1.18 – Адміністраторський обліковий запис: перегляд заявок

Після проведення розкадровки можна виокремити основні функціональні вимоги системи, наведені у табл. 1.17, а нефункціональні вимоги – у табл.1.18.

Таблиця 1.17 – Специфікація функціональних вимог

№	Функціональна вимога	Опис вимоги	Пріоритет	Коментарі
1	Автентифікація користувача	Система повинна дозволяти користувачам входити за допомогою логіна та пароля	Високий	Можлива інтеграція з SSO
2	Управління обліковими записами	Адміністратор має можливість додавати, редагувати та видаляти облікові записи	Високий	Застосовується лише до адміністратора
3	Перегляд розкладу відвідувань	Користувачі можуть переглядати свої заплановані візити	Середній	
4	Формування та відправка заявок	Користувачі можуть створювати заявки на відвідування	Високий	
5	Пошук за критеріями	Система повинна дозволяти фільтрувати заявки та візити за різними критеріями	Середній	
6	Генерація звітів	Адміністратор може створювати звіти про активність користувачів та заявки	Низький	

Така структура дозволяє систематизувати функціональні вимоги та вказати їх важливість для реалізації.

Таблиця 1.18 – Специфікація нефункціональних вимог

№	Нефункціональна вимога	Опис вимоги	Пріоритет	Коментарі
1	Продуктивність	Система повинна обробляти не менше 100 запитів на хвилину	Високий	
2	Масштабованість	Система повинна підтримувати додавання нових користувачів та функцій без зниження продуктивності	Високий	
3	Надійність	Час безперебійної роботи системи має становити не менше 99.9%	Високий	
4	Безпека	Дані користувачів повинні бути зашифровані, а доступ – контрольований	Високий	Вимога до відповідності стандартам безпеки
5	Юзабіліті	Інтерфейс повинен бути інтуїтивно зрозумілим для користувачів	Середній	
6	Сумісність	Система повинна працювати на різних операційних системах та браузерах	Середній	
7	Відновлюваність	Система повинна мати механізми для швидкого відновлення після збоїв	Середній	

Така таблиця допомагає структурувати вимоги, що не стосуються безпосередньо функцій веб-орієнтованої системи, але є важливими для її ефективної роботи. Далі представлено додаткові значення для

нефункціональних вимог:

- ✓ Час, необхідний для навчання звичайних користувачів, складає 30 хвилин.
- ✓ Час для навчання досвідчених користувачів становить 15 хвилин.
- ✓ Основні вимоги до застосовності нової системи у порівнянні з іншими системами, відомими користувачам: всі функції системи є легкими у використанні, а її структура схожа на існуючі аналоги.
- ✓ Вимоги до відповідності загальним стандартам застосовності та стандартам графічного інтерфейсу користувача: додаток повинен підтримувати роботу у всіх провідних браузерах.
- ✓ Доступність: час, витрачений на обслуговування системи, не повинен перевищувати 3% від загального часу її функціонування.
- ✓ Середній час безперервної роботи системи - 3 години.
- ✓ Використання ресурсів: мінімальні системні вимоги: (об'єм HDD: > 20 Мб; об'єм RAM: > 512 Мб; відеопам'ять: > 128 Мб; процесор: > 1 ГГц; клавіатура, миша; доступ до інтернету та веб-браузер).

Висновки до розділу 1

На підставі проведених досліджень можна зробити наступні висновки:

1. Проведений аналіз автоматизації роботи спортивного залу виявив низку проблем, пов'язаних із ручними процесами управління, які знижують ефективність роботи закладу. Визначено, що автоматизація є необхідною умовою для підвищення продуктивності та покращення обслуговування клієнтів.
2. Визначено ключові аспекти функціонування спортивного залу, включаючи управління відвідувачами, ведення обліку занять та моніторинг використання обладнання. Ця інформація стала основою для розробки вимог до нової інформаційної системи.
3. Аналіз конкурентних систем показав, що існуючі рішення не

повністю відповідають потребам користувачів, часто мають складний інтерфейс або обмежені функціональні можливості. Це вказує на потенціал для створення більш зручного та ефективного веб-орієнтованого рішення.

4. На основі проведеного дослідження сформульовано функціональні та нефункціональні вимоги до нової інформаційної системи. Ці вимоги забезпечать належну інтеграцію, безпеку, доступність та ефективність роботи системи.

Таким чином, дослідження проведені у першому розділі, надали всебічний аналіз поточного стану автоматизації роботи спортивного залу, визначили основні проблеми, а також було обґрунтовано необхідність розробки нової веб-орієнтованої інформаційної системи, що відповідатиме сучасним вимогам користувачів.

2. РОЗРОБЛЕННЯ АРХІТЕКТУРИ ПРОГРАМНОЇ СИСТЕМИ

2.1. Архітектура веб-орієнтованої інформаційної системи та технології розв'язання поставлених задач

Для розробки додатку було обрано трирівневу архітектуру системи, відому як модель прикладного сервера (AS - Application Server). У цій архітектурі реалізовано трирівневу систему розподілу функцій, що забезпечує ефективну взаємодію між клієнтською частиною, серверами бізнес-логіки та базами даних (рис. 2.1). Такий підхід дозволяє підвищити продуктивність системи, полегшити масштабування та забезпечити кращу організацію процесів, що є важливими факторами для стабільної роботи сучасних веб-додатків. Розподіл функцій на різних рівнях також сприяє спрощенню обслуговування та оновлення системи.

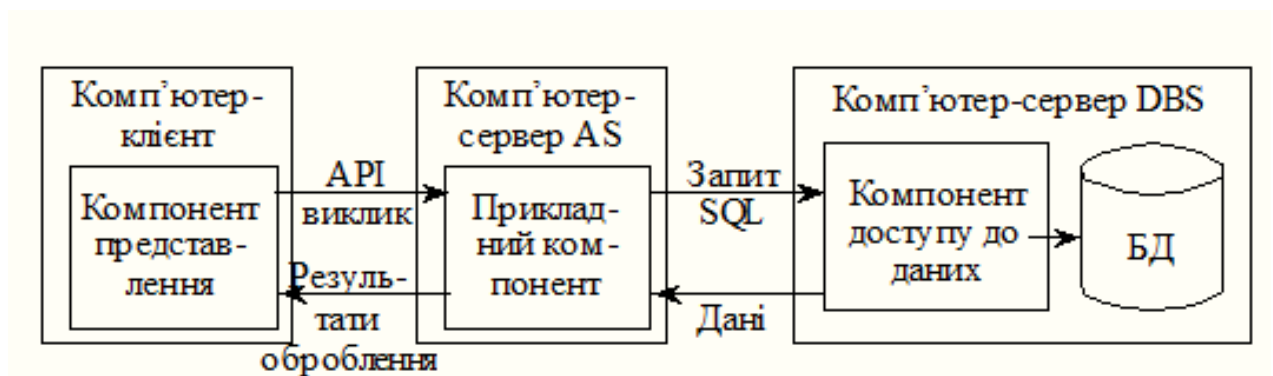


Рисунок 2.1 – Модель трирівневої архітектури веб-орієнтованої системи

Перший рівень - це комп'ютер клієнта, на якому розташований користувацький інтерфейс, а також функції для локального редагування та перевірки даних. Система повинна взаємодіяти з мережею, тому на клієнті виконуються функції, що відповідають за комунікацію з користувачем. У цьому випадку комп'ютер виступає в ролі клієнта додатку, звертаючись до компонентів, розміщених на сервері.

Другий рівень - це прикладний сервер, який є ключовим елементом

трирівневої архітектури клієнт-сервер. Його основна функція полягає у зберіганні та виконанні бізнес-логіки. Прикладний сервер реалізований як набір процедур, що виконують специфічні прикладні функції, і часто називається Application Server. Відокремлення прикладної логіки в окремий архітектурний рівень дає можливість використовувати різноманітні мови програмування (C, C++, C#, Cobol, PHP), які володіють великою спеціалізацією для роботи з базами даних. Це підвищує незалежність функціональних компонентів одного рівня від змін або вдосконалень компонентів на інших рівнях.

Третій рівень - це сервер бази даних, відповідальний за зберігання та обробку даних. Він виконує важливі функції, такі як забезпечення цілісності даних, запобігання несанкціонованим або некоректним змінам, а також створення резервних копій. Цей рівень забезпечує правильне структурування та управління інформацією, що зберігається в базі даних.

Для опису бізнес-логіки та наочного представлення її структури було використано UML-діаграму класів (рис.2.2).

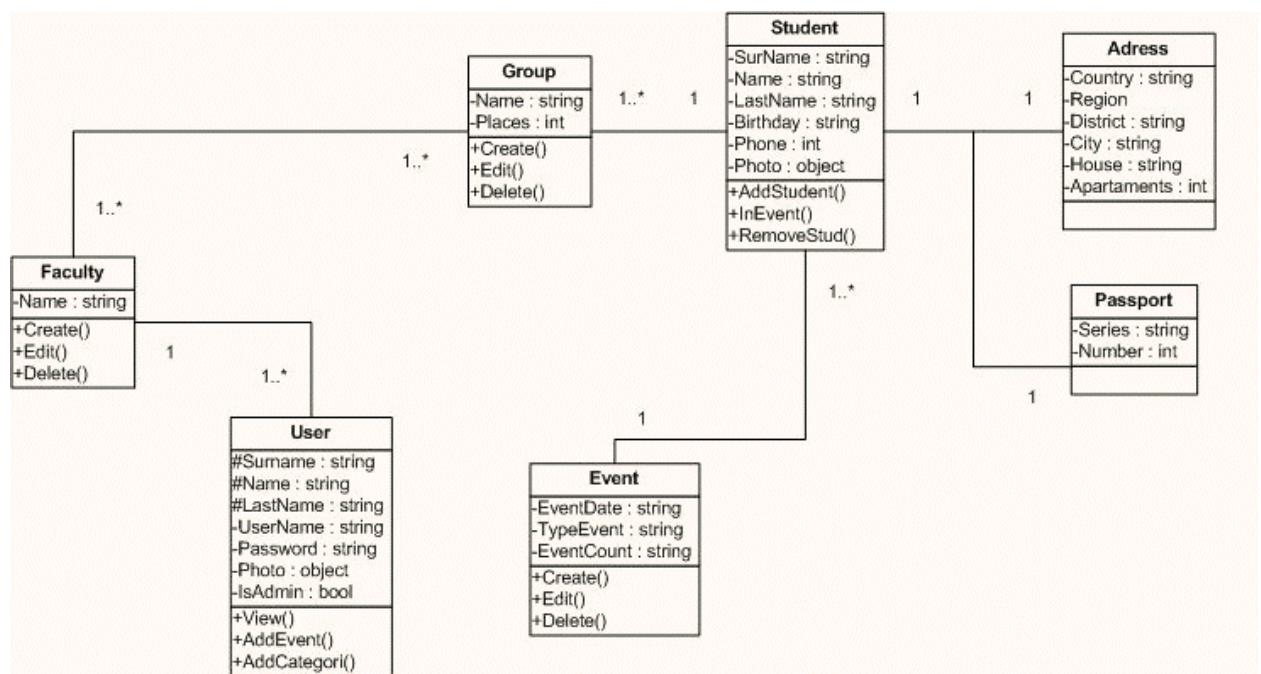


Рисунок 2.2 – UML-діаграма класів

Дана діаграма дозволяє візуалізувати основні класи системи, їх

атрибути, методи та взаємозв'язки між ними, що сприяє кращому розумінню архітектури системи та її функціональності. UML-діаграма класів слугує важливим інструментом у процесі проектування, оскільки допомагає командам розробників узгодити бачення системи та визначити ключові елементи бізнес-логіки.

На даній діаграмі представлені прототипи основних класів системи та зв'язки між ними. Описані функції включають додавання, редагування, видалення та перегляд графіку відвідувань, а також внесення нових категорій до графіку. Функціональну структуру системи умовно можна розділити на такі модулі:

1. Модуль обробки інформації про користувачів – реєстрація та авторизація
2. Модуль обробки інформації про графіки відвідувань – додавання відвідування; додавання даних відвідувачів так перегляд графіку відвідування

Для кращого розуміння процесу функціонування системи були побудовані діаграми активності для кожного модуля (рис. 2.3 -).



Рисунок 2.3 – Діаграма активності для авторизації користувача



Рисунок 2.4 – Діаграма активності процесу реєстрації користувача



Рисунок 2.5 – Діаграма активності процесу додавання відвідування

Таким чином, архітектура веб-орієнтованої інформаційної системи базується на трирівневій моделі, що забезпечує ефективний розподіл функцій між клієнтською частиною, прикладним сервером і сервером бази даних. Впровадження сучасних технологій розв'язання задач, таких як використання UML-діаграм для моделювання бізнес-логіки та активності, дозволяє створити інтуїтивно зрозумілий інтерфейс та забезпечити стабільну роботу системи. Такий підхід забезпечує високу продуктивність, масштабованість та безпеку, що є ключовими аспектами для успішної реалізації системи.

2.2. Проектування структури бази даних

Етап проектування бази даних (БД) вважається одним з найскладніших етапів у її створенні, оскільки не має чітко визначених початкових і кінцевих точок. На відміну від аналізу вимог до БД або розробки програм, проектування БД часто розглядається як погано структуроване завдання, за словами багатьох провідних експертів. У той час як етапи створення БД взаємопов'язані, етап проектування переплітається з усіма іншими етапами.

Проектування починається з ухвалення стратегічних рішень і продовжується протягом етапів реалізації та тестування. Процес проектування БД охоплює кілька ключових аспектів:

- Проектування об'єктів БД (таблиці, подання, індекси, тригери, збережені процедури, функції, пакети) для представлення даних у системі.
- Проектування інтерфейсу взаємодії з БД (форми, звіти тощо), що передбачає розробку додатків, які супроводжуватимуть дані в БД і реалізовуватимуть запитально-відповідні відносини.
- Проектування БД для конкретного обчислювального середовища або інформаційної технології (архітектура «клієнт-сервер», паралельні архітектури, розподілені обчислювальні середовища).
- Проектування БД з урахуванням призначення системи (інтелектуальний аналіз даних, OLAP, OLTP тощо).

Розробку бази даних слід починати зі створення DFD-моделі предметної області, яка зображена на рис. 2.6.

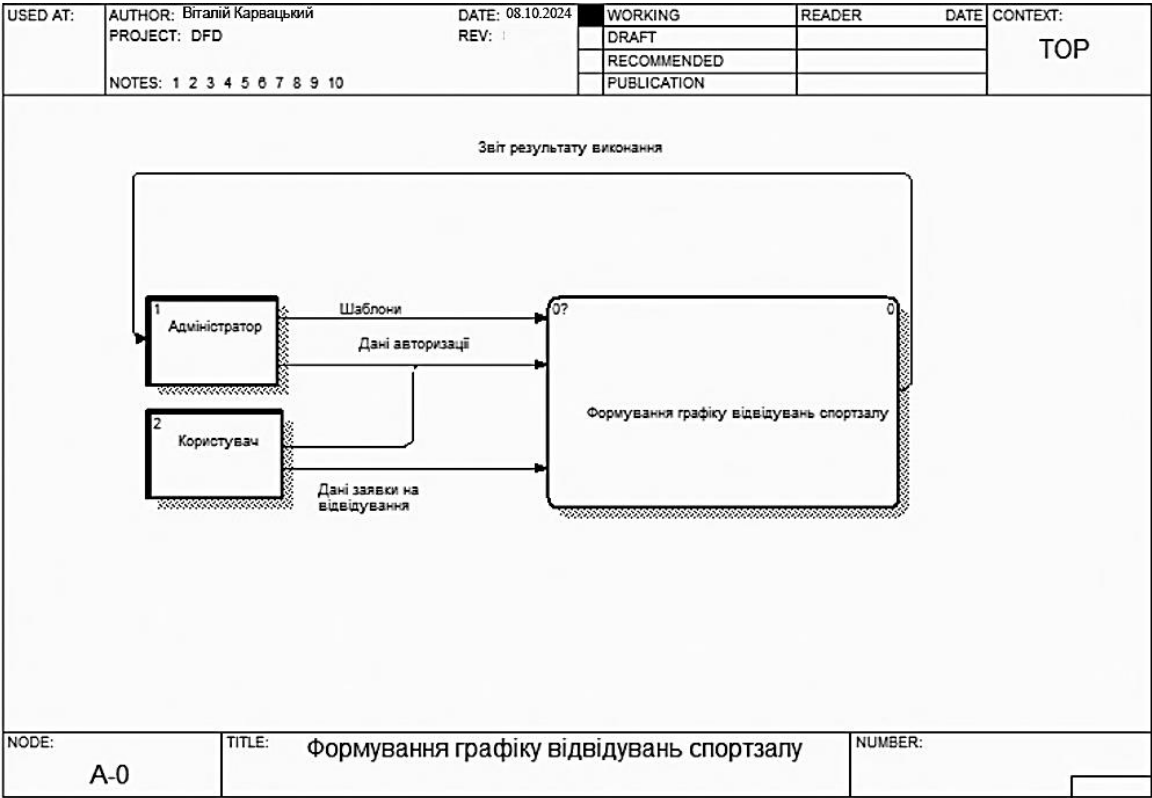


Рисунок 2.6 – Діаграма потоків даних

Діаграма декомпозиції представлена на рис. 2.7.

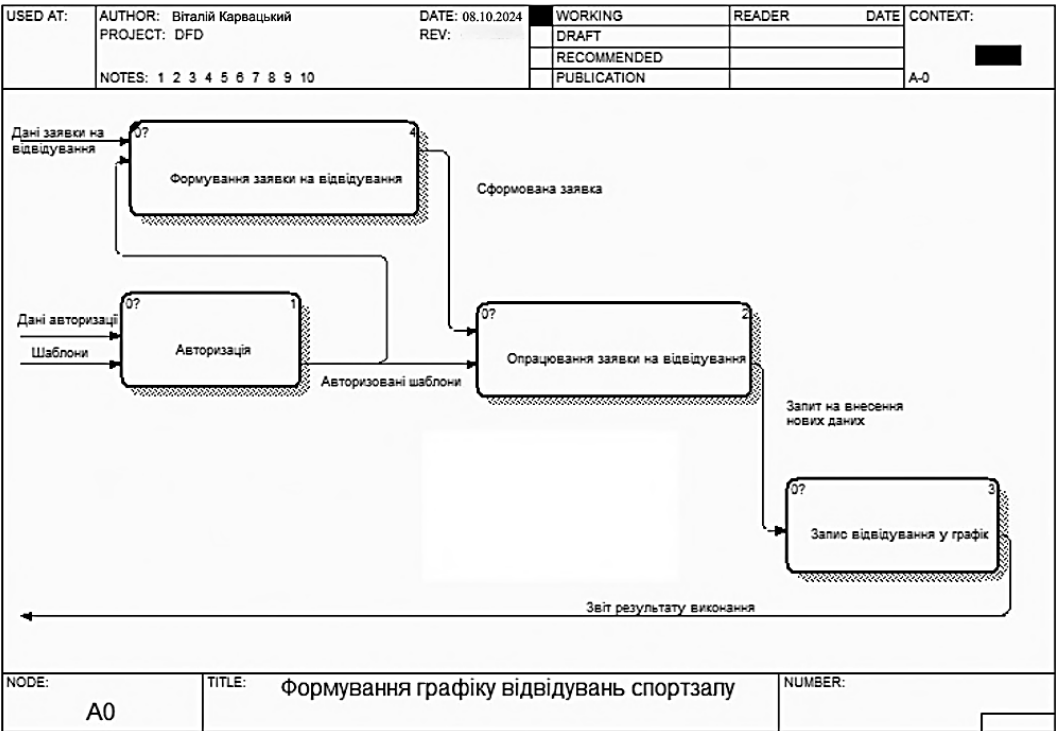


Рисунок 2.7 – Діаграма декомпозиції

Як видно на рисунку 2.7, функціонування програмної системи відбувається наступним чином:

- **Входи (input)** - це потоки даних, що надходять до системи для обробки і перетворення у вихідні результати. Ці елементи зображені зліва на діаграмі і включають: дані авторизації, шаблони, а також інформацію зі заявки на відвідування.
- **Виходи (output)** - результати роботи системи, які є продуктами перетворення вхідних даних. Вони відображені справа на діаграмі і включають: звіт про результати виконання.

Процес складання графіку здійснює адміністратор, який взаємодіє із програмною системою. Після перегляду заявки на відвідування адміністратор обирає дати початку та завершення події, щоб визначити рамки виконання. Далі він вводить інформацію про подію, включаючи її опис та категорію.

Наступним кроком є внесення інформації про користувачів системи. Оскільки користувачами є студенти та викладачі, необхідно розробити ієрархію сутностей, починаючи від факультету і закінчуючи даними студентів.

Для виявлення основних сутностей і зв'язків між ними була створена діаграма елементів та зв'язків (рис. 2.8).

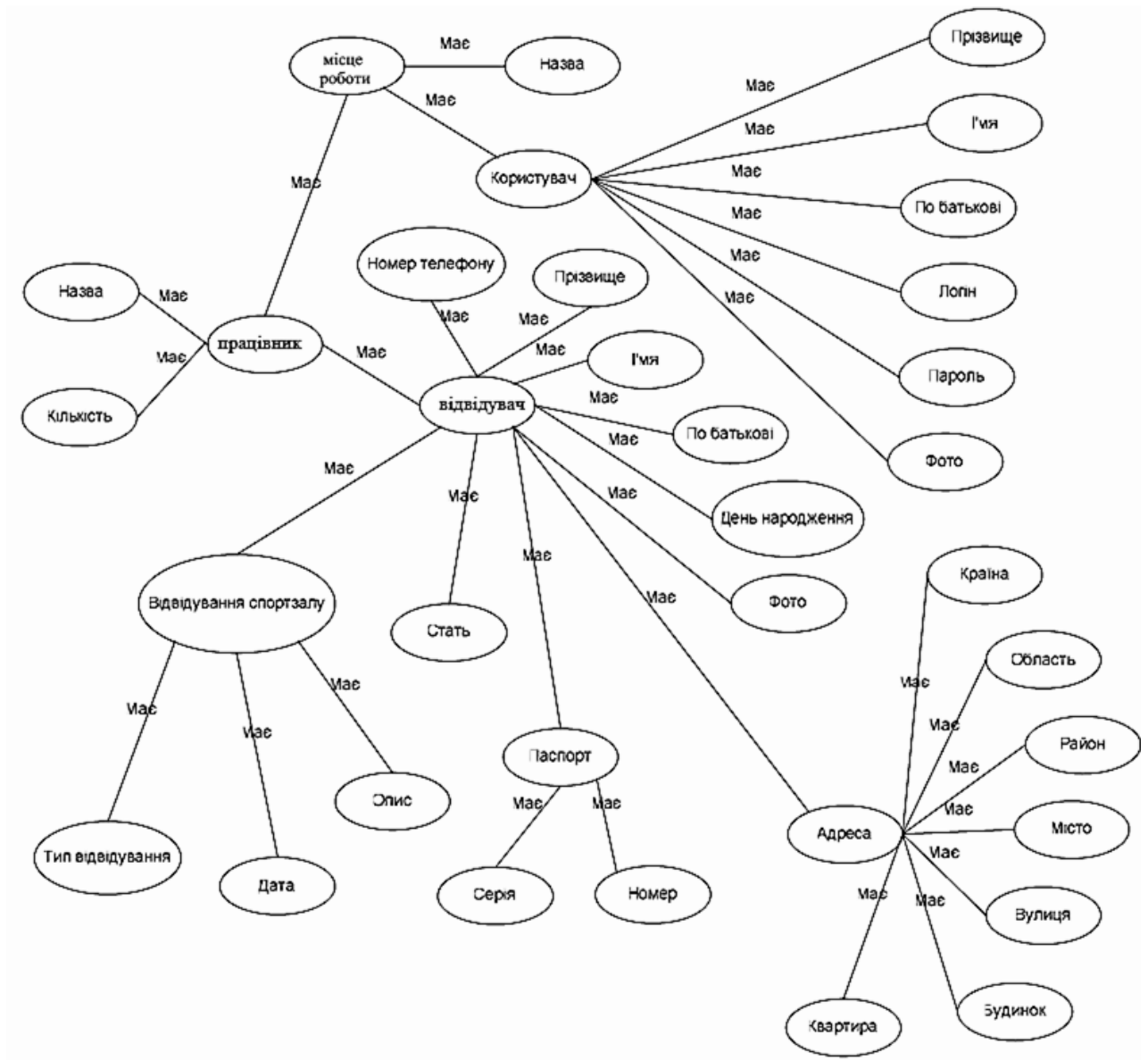


Рисунок 2.8 – Діаграма елементів і зв'язків

Після цього була створена таблиця ідентифікаторів, яка представлена у табл. 2.1.

Таблиця 2.1 – Таблиця ідентифікаторів

Об'єкт	Властивість	Тип	Розмірність	Ідентифікатор
Користувач	Ім'я	Текст	50	user_name
Користувач	Пароль	Текст	50	user_password
Користувач	Роль	Текст	20	user_role
Відвідування	Дата	Дата/Час	-	visit_date
Відвідування	Опис	Текст	255	visit_description
Відвідування	Категорія	Текст	30	visit_category
Шаблон	Назва	Текст	100	template_name
Шаблон	Вміст	Текст	1000	template_content

Завершальним етапом проєктування бази даних є розробка ER-діаграми, яка представлена на рис. 2.9

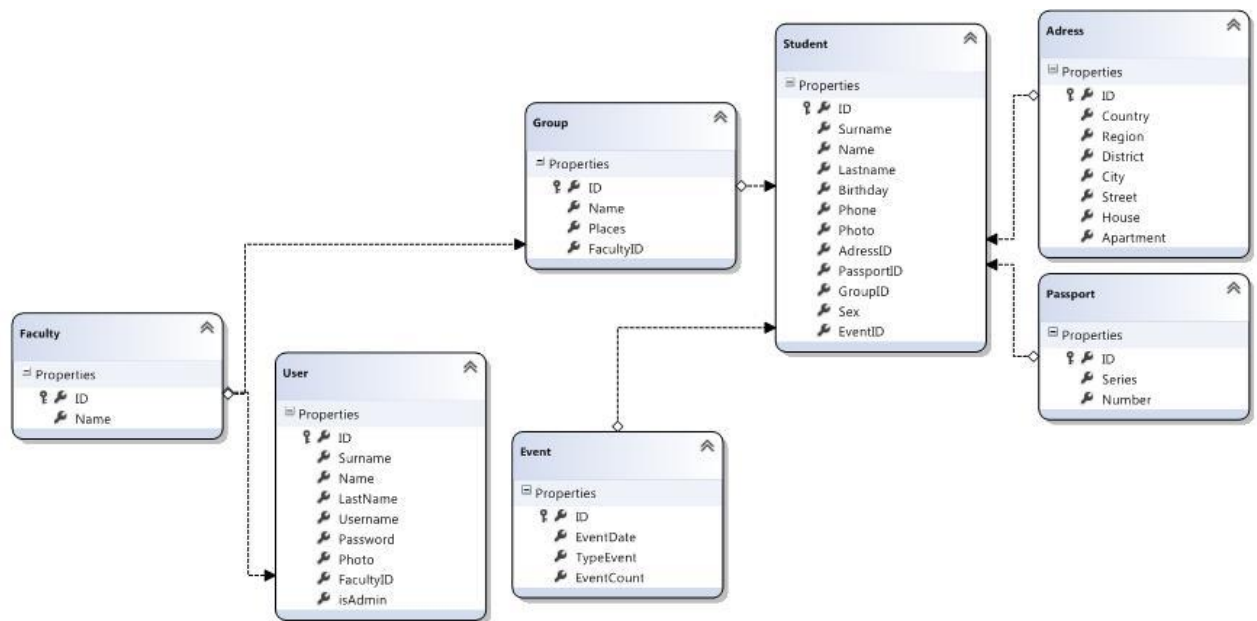


Рисунок 2.9 – ER-діаграма розробленої веб-орієнтованої системи для користування спортивним залом

Отже, проєктування структури бази даних дає змогу чітко визначити та детально описати всі зв'язки та поля, слугуючи останнім кроком перед переходом до програмної реалізації бази даних.

2.3. Функціональна структура задачі та підсистеми забезпечення функціональної частини

Автоматизація процесів управління системами (ПС) може бути реалізована на різних рівнях – від автоматизації окремих інформаційних процесів і процедур обробки даних до комплексних функціонально завершених автоматизованих інформаційних систем (АІС). Функціонально завершені АІС охоплюють автоматизацію всіх підсистем, функцій і комплексів завдань. Як і будь-яка складна система, АІС складається з елементів – підсистем, які можуть бути розглянуті як підсистеми нижчого рівня, що містять свої елементи. У зв'язку з наявністю великої кількості

елементів у АІС виникає потреба в структурному визначенні цих елементів.

Структура АІС визначає внутрішню організацію системи, що виникає в процесі розподілу П на частини та виявлення зв'язків між цими частинами. Структура складається з безлічі елементів та взаємозв'язків між ними. Найзагальнішим поділом АІС є розмежування на функціональну та забезпечувальну частини. Функціональна частина системи завжди пов'язана з ПС і фактично є моделлю управління конкретним об'єктом. Вона ділиться на функціональні підсистеми, які можуть різнитися в залежності від специфіки конкретного економічного об'єкта. Важливим критерієм для виділення функціональних підсистем є виконувані функції управління об'єктом.

Функціональна підсистема представляє собою частину АІС, виділену за спільністю функціональних ознак управління. У функціональній структурі АІС часто відтворюється склад функціональних підрозділів організаційної структури об'єкта. Назви функціональних підсистем відображають їх цільові функції та діяльність управлінських органів.

Функціональна характеристика декомпозиції АІС визначає призначення підсистеми, включаючи сферу діяльності, основні цілі, завдання та функції, які вона виконує.

Кількість функціональних підсистем може варіюватися від 10 до 20 і більше, залежно від складності об'єкта. Незважаючи на те що деякі підсистеми можуть мати однакові назви на різних об'єктах (наприклад, підсистема бухгалтерського обліку на підприємстві), їх внутрішній зміст може значно відрізнятися.

Специфічні особливості кожної підсистеми відображаються в функціональних завданнях, для автоматизації яких вона призначена. При виділенні функціональних підсистем в АІС необхідно дотримуватися таких вимог:

- Межі завдань, що формують підсистему, не повинні перекриватися.
- Завдання, що виконуються в підсистемах, мають бути тісно

пов'язані в інформаційному плані, тобто для їх виконання повинна використовуватися єдина вхідна інформація, а результати одних завдань повинні використовуватися для розв'язання інших.

- Результати розв'язання мають мати єдиного споживача (управлінський персонал одного або кількох функціональних підрозділів об'єкта).

При виділенні функціональних підсистем також слід визначити їх параметри: мету функціонування підсистеми; вид керованих ресурсів; особливості показників, що розраховуються в підсистемі та підрозділи, які здійснюють управління.

Розробка функціональної структури задачі та підсистем для веб-орієнтованої системи, яка організовує роботу спортивного залу, може включати кілька основних етапів:

Перший етап: створення інтегрованої веб-орієнтованої системи, що забезпечує ефективне управління діяльністю спортивного залу, включаючи: бронювання залів; управління членством; планування тренувань та комунікацію з клієнтами.

Другий етап включає: управління членством (реєстрація нових членів, відстеження статусу членства, обробка платежів та підписок), бронювання залів (вибір дати та часу для бронювання, перегляд доступності залів, підтвердження, редагування та скасування бронювань), планування тренувань (створення та редагування графіка тренувань, запис членів на тренування, відстеження відвідуваності та результатів), комунікація з клієнтами (надсилання сповіщень, організація опитувань, надання чату або підтримки), аналітика та звітність (генерація звітів про використання залу, аналіз фінансових показників, оцінка потреб клієнтів).

Третій етап включає саме *підсистему забезпечення функціональної частини*, яка складається із *інформаційної підсистеми* (бази даних, забезпечення безпеки даних, обробка запитів), *інтерфейсу користувача* (веб-додаток, адаптивний дизайн, мобільний додаток), *система управління*

платежами (інтеграція з платіжними системами, забезпечення захисту фінансових даних), *технічна підсистема* (серверна частина, забезпечення масштабованості, інфраструктура) та *контроль якості* (моніторинг, логи для аудиту, оцінка задоволеності).

Даний поетапний план дозволяє чітко структурувати розробку веб-орієнтованої системи для організації роботи спортивного залу, з акцентом на основні функції та підсистеми.

Висновки до розділу 2

У другому розділі дослідження, були зроблені такі висновки:

1. Розроблена архітектура веб-орієнтованої інформаційної системи забезпечує інтеграцію всіх основних компонентів, що дозволяє ефективно вирішувати задачі управління діяльністю спортивного залу. Використання сучасних технологій, таких як RESTful API та хмарні рішення, підвищує гнучкість і масштабованість системи.

2. Структура бази даних, яка була спроектована, дозволяє зберігати всю необхідну інформацію про членів, бронювання, тренування та фінансові дані у впорядкованому вигляді. Забезпечено безпеку та цілісність даних через реалізацію механізмів контролю доступу та регулярних резервних копій.

3. Визначено основні функції системи, включаючи управління членством, бронювання залів, планування тренувань та комунікацію з клієнтами, що створює комплексний підхід до автоматизації процесів. Окремі підсистеми, такі як інформаційна підсистема, інтерфейс користувача та система управління платежами, сприяють ефективному виконанню завдань, підвищуючи загальну продуктивність та зручність для користувачів.

Таким чином, розробка архітектури програмної системи забезпечує необхідні умови для реалізації поставлених цілей і завдань, створюючи надійну основу для подальшого розвитку та вдосконалення веб-орієнтованої системи управління спортивним залом.

3. ПРОЄКТУВАННЯ ТА ТЕСТУВАННЯ ВЕБ-ОРІЄНТОВАНОЇ ІНФОРМАЦІЙНОЇ СИСТЕМИ РОБОТИ СПОРТИВНОГО ЗАЛУ

3.1. Програмна реалізація проєкту

3.1.1. Вибір програмних засобів для реалізації проєкту

Платформа ASP.NET представляє собою технологію від компанії Microsoft, яка забезпечує потужні можливості для розробки веб-додатків. Вибір ASP.NET для створення веб-орієнтованої інформаційної системи для управління спортивним залом зумовлений кількома ключовими факторами:

- *Продуктивність і ефективність.* ASP.NET використовує компіляцію, що забезпечує високу продуктивність веб-додатків. Це дозволяє обробляти запити швидше, що є важливим аспектом для системи, що обробляє численні запити від користувачів.
- *Безпека.* Платформа ASP.NET містить вбудовані функції безпеки, такі як аутентифікація та авторизація користувачів, що дозволяє захистити дані членів і фінансову інформацію. Це особливо важливо для системи, яка обробляє чутливу інформацію.
- *Зручність у використанні.* ASP.NET надає розробникам потужні інструменти для створення інтуїтивно зрозумілих інтерфейсів користувача, завдяки чому адміністратори і клієнти можуть легко взаємодіяти з системою. Це підвищує зручність використання та загальну задоволеність користувачів.
- *Підтримка та документація.* Microsoft забезпечує широке співтовариство і документацію для ASP.NET, що спрощує процес навчання і розв'язання проблем під час розробки. Це дає змогу швидше знайти рішення для різних технічних викликів, що можуть виникнути під час створення системи.
- *Гнучкість і масштабованість.* ASP.NET підтримує розробку як

невеликих, так і великих проєктів, що робить його ідеальним вибором для системи, яка може зрости в майбутньому. Це дозволяє легко масштабувати систему відповідно до зростаючих потреб спортивного залу.

- *Інтеграція з базами даних.* Платформа надає зручні механізми для роботи з базами даних, включаючи Entity Framework, що полегшує створення і управління базами даних. Це важливо для зберігання та обробки даних про членів, бронювання та фінансову інформацію. На даний момент останньою версією цієї технології є ASP.NET 5.

З одного боку, ASP.NET 5 є продовженням еволюції платформи ASP.NET. Проте, з іншого боку, це не просто ще один реліз. Випуск ASP.NET 5 фактично знаменує собою революцію в усій платформі, що супроводжується якісними змінами.

Тепер ASP.NET 5 є повністю відкритим фреймворком з підтримкою крос-платформеності. Всі вихідні файли фреймворку доступні на GitHub, що означає, що веб-додатки можна запускати не лише на операційній системі Windows, а й на Linux і MacOS.

Додаток ASP.NET 5 може функціонувати в двох середовищах виконання: .NET Core та повній версії фреймворку .NET. .NET Core є модульним крос-платформеним середовищем, яке спрощує процес розгортання програм.

Завдяки модульності, усі необхідні компоненти веб-додатку можуть бути завантажені як окремі модулі через пакетний менеджер NuGet. Більше того, на відміну від попередніх версій платформи, відпала необхідність у використанні бібліотеки System.Web.dll.

ASP.NET 5 оптимізовано для роботи в хмарних середовищах і має вбудовану підтримку впровадження залежностей. Для розгортання веб-додатка можна використовувати традиційний IIS, а також запускати його за допомогою крос-платформного веб-сервера Kestrel.

ASP.NET 5 включає в себе фреймворк MVC 6, який поєднує функціональність MVC, Web API та Web Pages. У попередніх версіях ці

технології реалізувалися окремо, що призводило до дублювання функцій. Тепер же вони інтегровані в єдину програмну модель MVC 6, в той час як Web Forms остаточно пішли в минуле.

Крім об'єднання цих технологій в одну модель, MVC отримав нові функції. Однією з таких є тег-хелпери, які дозволяють більш природно поєднувати HTML-синтаксис з кодом C#.

C# належить до сімейства мов з C-подібним синтаксисом, і з усіх цих мов його синтаксис найближчий до C++ і Java. Мова має статичну типізацію, підтримує поліморфізм, перевантаження операторів (включаючи явне та неявне приведення типів), делегати, атрибути, події, властивості, узагальнені типи та методи, ітератори, анонімні функції з підтримкою замикань, LINQ, обробку виключень і XML-коментарі.

Спадкуючи багато з мов, таких як C++, Pascal, Smalltalk і, особливо, Java, C# унікає деяких моделей, які зарекомендували себе як проблемні при розробці програмних систем. Наприклад, на відміну від C++, C# не підтримує множинне успадкування класів, але допускає множинне спадкування інтерфейсів.

Для розробки даної системи було обрано Microsoft Visual Studio 2013. Структура системи передбачає наявність двох ролей: користувача та адміністратора.

Користувач - це особа, яка, перейшовши на головну сторінку сайту, може ознайомитися з інформацією та, пройшовши процедуру авторизації, подати заявку на абонемент у спортзал. Адміністратор, також проходячи авторизацію, отримує значно розширені права. Він має можливість додавати, видаляти та змінювати всю інформацію на сайті, а також доступ до персональних даних клієнтів.

Програмування веб-сайтів на платформі ASP.NET складається з двох основних етапів: розробка інтерфейсу системи та написання алгоритмів роботи продукту. Інтерфейс формується з файлів, які містять розмітку сторінок і інші елементи, тоді як алгоритми системи реалізуються в коді, який

відповідає за взаємодію з користувачами.

Для реалізації авторизації адміністратора на сайті з фіксованими логіном та паролем використовується наступний код (рис.3.1):

```
public bool AuthorizeAdmin(string username, string password)
{
    // Перевірка логіна та пароля
    if (username == "admin" && password == "password123")
    {
        return true; // Авторизація пройшла успішно
    }
    return false; // Авторизація не вдалася
}
```

Рисунок 3.1 – Фрагмент коду «авторизація адміністратора на основі логіна та пароля»

Після успішної авторизації на веб-сайті адміністратор матиме можливість виконувати такі дії: додавати нові відвідування; редагувати графік відвідувань; вносити нові дані; переглядати заявки та вносити зміни до існуючих даних.

У великих компаніях може бути кілька адміністраторів і багато користувачів, тому система враховує цю ситуацію. Перед внесенням змін до даних на сайті виконується пошук і витягнення інформації з бази даних про конкретного адміністратора або користувача. Ця операція забезпечує моніторинг внесених змін на веб-сайті та дозволяє визначити персональну відповідальність за них (рис.3.2).

```
using System;
using System.Data.Entity;
using System.Linq;

public class UserService
{
```

```

private readonly MyDbContext _context;

public UserService(MyDbContext context)
{
    _context = context;
}

// Метод для отримання адміністратора за його ID
public Administrator GetAdministratorById(int adminId)
{
    return _context.Administrators
        .FirstOrDefault(admin => admin.Id ==
adminId);
}

// Метод для отримання користувача за його ID
public User GetUserById(int userId)
{
    return _context.Users
        .FirstOrDefault(user => user.Id ==
userId);
}

// Метод для внесення змін до даних
public void UpdateUserData(User updatedUser)
{
    var existingUser =
_context.Users.Find(updatedUser.Id);

```

**Рисунок 3.2 – Фрагмент коду сервісу користувачів для взаємодії з
базою даних**

Спочатку здійснюється підключення до бази даних, після чого виконується пошук інформації про конкретного адміністратора та зчитується його ідентифікаційний номер. Таким чином, вибір ASP.NET як мови програмування для розробки веб-орієнтованої інформаційної системи обґрунтований її перевагами в продуктивності, безпеці, зручності використання, підтримці та гнучкості, що сприятиме створенню ефективної системи управління діяльністю спортивного залу.

3.1.2. Організація інтерфейсу з користувачем

Створення інтерфейсу користувача є важливим і невід’ємним етапом у процесі розробки програмного забезпечення. У системі обліку та контролю занять спортом, яка розробляється, буде реалізовано графічний інтерфейс у традиційному стилі додатків для ОС Windows. Основні елементи інтерфейсу включатимуть:

- Меню «Файл», що містить пункт «Вихід», який забезпечує завершення роботи програми.
- Меню «Допомога», з пунктом «Про програму...», вибір якого відкриває відповідне інформаційне вікно

Під час розробки веб-орієнтованої системи для управління діяльністю спортивного залу особливу увагу було приділено створенню інтуїтивно зрозумілого, зручного та доступного інтерфейсу для різних категорій користувачів, таких як адміністратори та звичайні користувачі.

Завантаження головної сторінки сайту (рис. 3.3) є однаковим для всіх: користувачі отримують доступ до загальної інформації та можливості перегляду розкладу занять (рис. 3.4). Після авторизації вони переходять до своїх персональних облікових записів. У разі відсутності реєстрації користувач може скористатися кнопкою «Реєстрація» для переходу до відповідної форми (рис. 3.5).

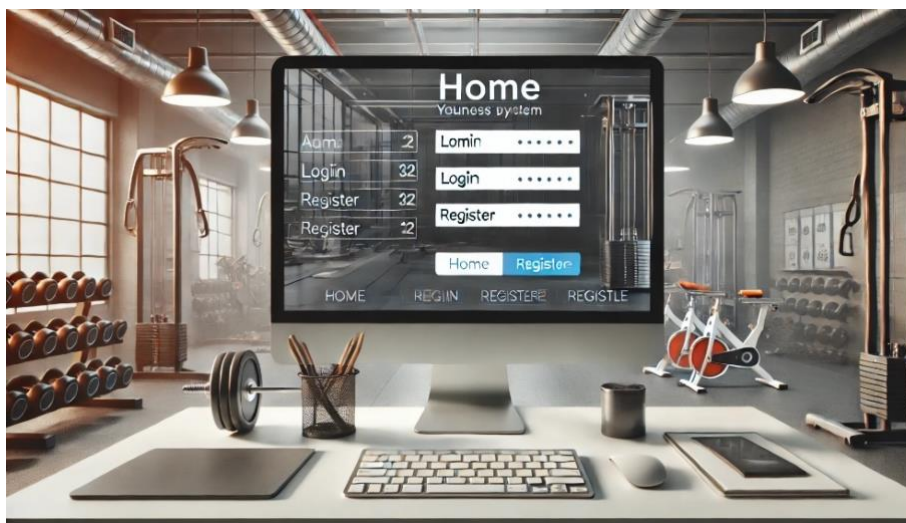


Рисунок 3.3 – Головна сторінка сайту спортивного залу

Реалізація головної сторінки забезпечує доступ до ключових функцій системи, таких як перегляд розкладу занять, авторизація та реєстрація. Такий підхід сприяє зручності використання, оптимізації роботи адміністратора та підвищенню залученості користувачів.

На рис. 3.4 представлено «Графік відвідування» спортивного залу.

Головна				Графік відвідування			
Час	Пн	Вт	Ср	Чт	Пт	Сб	Нд
09:00	Розумне тіло	Stretching	Розумне тіло	Stretching	Pilates		
10:00	Body sculpt	Стрункі ніжки	Табата	Upper body + прес	Стрункі ніжки	Pilates	Stretching
11:00	Stretching	Pilates	Плоский живіт+АВТ	Pilates	Body sculpt	Body sculpt	Functional training
12:00						Бокс	Body sculpt
17:10	Гнучка сила	(17:00) Прес+Стрейч	Stretching	(17:00) Upper body	Каланетика		
18:10	Body sculpt	(18:00) СТОП Целюліт	Кругове тренування	(18:00) Body sculpt	Прес+Стрейч		
19:10	Кругове тренування	(19:00) Бокс	Стрункі ніжки	(19:00) Бокс	Кругове тренування		
20:10	Step Dance	(20:30) Бокс	Functional+Step	(20:30) Бокс	Step Dance		
21:10	Stretching RELAX		Stretching RELAX		Stretching RELAX		

Рисунок 3.4 – Сторінка перегляду розкладу занять

Рисунок зображує сторінку для перегляду розкладу занять у спортивному залі, яка включає: зображення залу для візуалізації атмосфери спортивного простору; блок авторизації та реєстрації для зручного доступу користувачів та таблицю розкладу занять, поділену за днями тижня та часом, із зазначенням типів тренувань (наприклад, Stretching, Body Sculpt, Кругове тренування).



Реєстрація

Введіть логін

Введіть пароль

Введіть пароль ще раз

Рисунок 3.5 – Сторінка аутентифікації

Даний рисунок демонструє сторінку аутентифікації для доступу до сервісу спортивного залу. Основні елементи: *верхня частина* - зображення тренажерного залу для створення приємного візуального контексту; *форма реєстрації*: включає три поля для введення (Логін, Пароль та Підтвердження пароля) та *кнопка «Зареєструвати»* - для завершення процесу створення облікового запису.

При авторизації адміністратор переходить в свій обліковий запис, сторінка якого має дві вкладки «Внесення даних» та «Заявки». На вкладці «Внесення даних» (рис. 3.6.) адміністратор вводить, редагує чи видаляє дані відвідувачів.

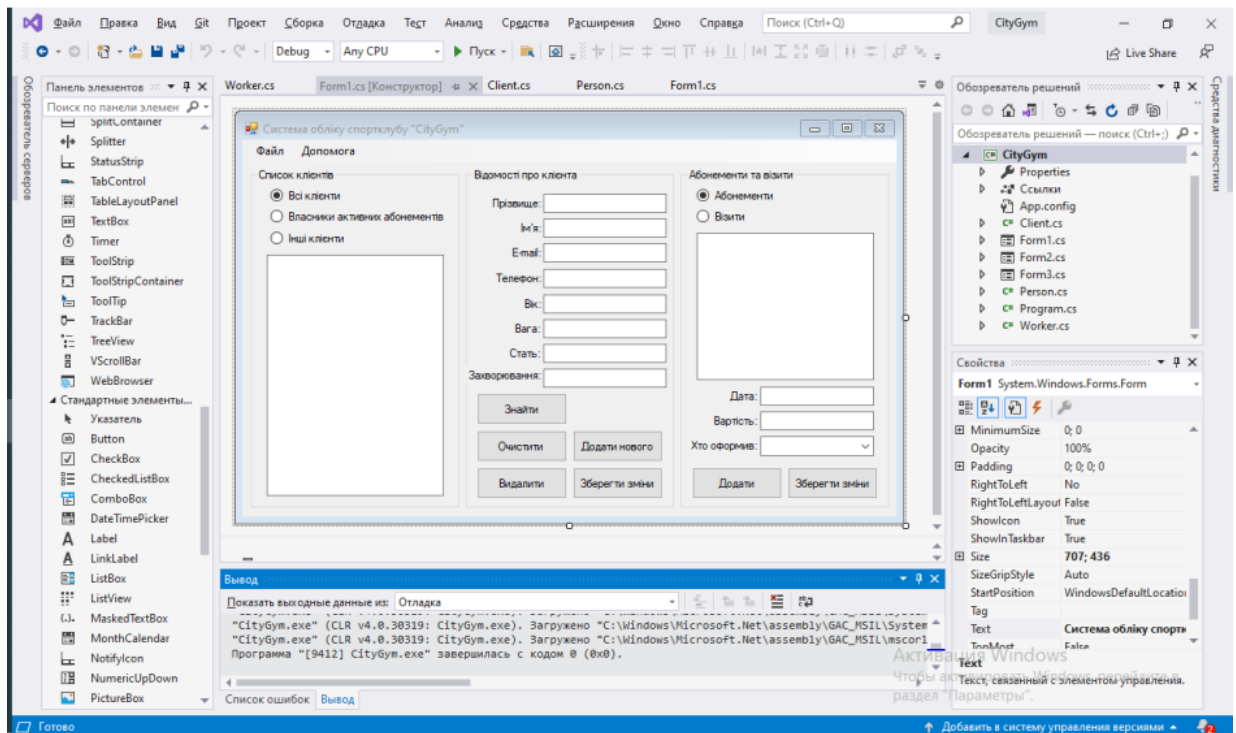


Рисунок 3.6 – Сторінка облікового запису адміністратора, вкладка «Внесення даних»

Рисунок 3.7 демонструє сторінку облікового запису адміністратора спортивного комплексу на вкладці «Заявки». Основні функції представленої сторінки:

- Список отриманих заявок. Ліворуч відображено перелік студентів, які подали заявки на відвідування тренажерного залу або занять спортом.

Виділення студента дозволяє переглянути деталі заявки у вікні праворуч.

- Деталі заявки. У середній частині сторінки відображається вміст обраної заявки. Наприклад, студент Петренко Петро подає запит на відвідування тренажерного залу щосереди з 10:00 до 12:00.
- Редагування заявки. Можливість змінити тип відвідування, день тижня і час за допомогою відповідних випадаючих списків. Після внесення змін доступна кнопка для підтвердження даних нового відвідування.
- Пошук та розклад. Праворуч відображено пошукові фільтри за типом відвідування, днем тижня та часом. Нижче розташований розклад занять у спортивній залі, що відображає доступний час та кількість вільних місць для кожного виду занять. Окремо представлений розклад для тренажерного залу.

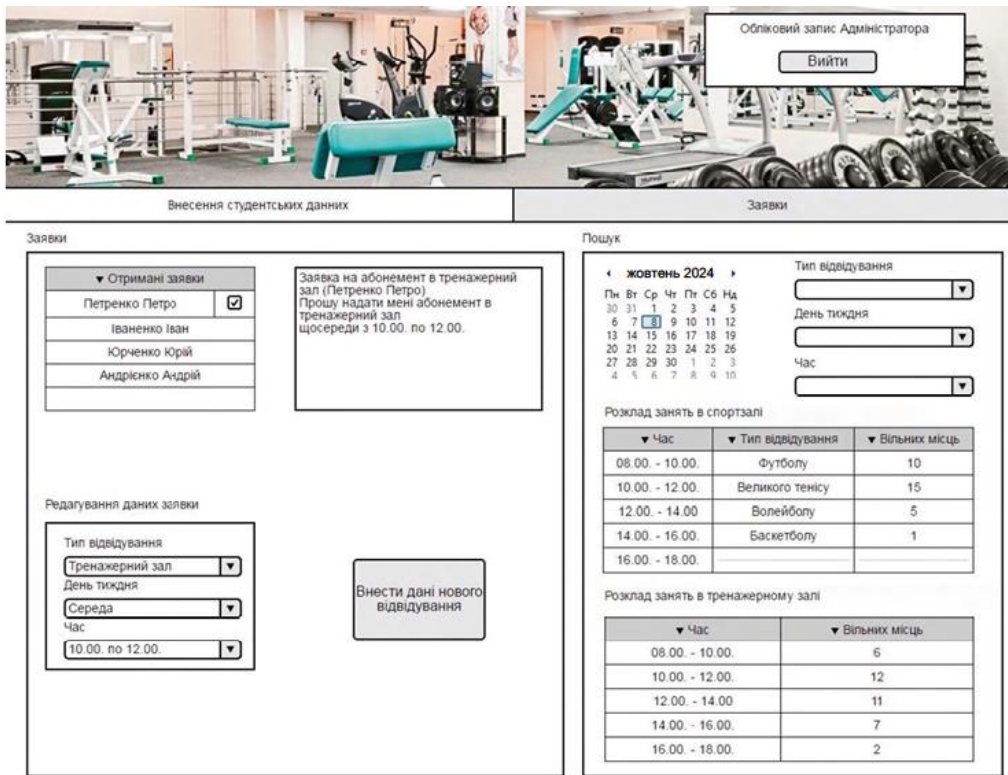


Рисунок 3.7 – Сторінка облікового запису адміністратора, вкладка «Заявки»

Загалом, цей інтерфейс дозволяє адміністратору ефективно керувати заявками, планувати заняття та відслідковувати завантаженість спортивних зон.

3.2. Створення бази даних системи

Для створення бази даних було обрано Microsoft SQL Server Management Studio 2016, інструмент, який використовується для роботи з СУБД Microsoft SQL Server 2016. Цей програмний продукт забезпечує управління, налаштування та адміністрування компонентів бази даних, а також інтегрується з Microsoft Visual Studio, що гарантує високу сумісність і підтримку розробки баз даних із програмним забезпеченням.

SQL Server Management Studio (SSMS) слугує менеджером для управління об'єктами SQL-сервера. Він має зручний інтерфейс і дозволяє створювати та виконувати скрипти для адміністрування баз даних.

У рамках розробки веб-орієнтованої системи планування графіку спортивного залу передбачено створення єдиної бази даних. Ця база зберігатиме інформацію про графік відвідувань спортзалу, дані користувачів та студентів, які є основними відвідувачами. Для організації структури бази даних необхідно створити таблиці, зокрема: *Faculty*, *User*, *Specialization*, *Profession*, *Student*, *Address*, *Passport*, *Event*, *Gym*.

Щоб розпочати створення таблиць, необхідно спочатку встановити та налаштувати Microsoft SQL Server і SQL Server Management Studio. Після цього, запустивши SSMS, потрібно підключитися до сервера, як показано на рис. 3.8.

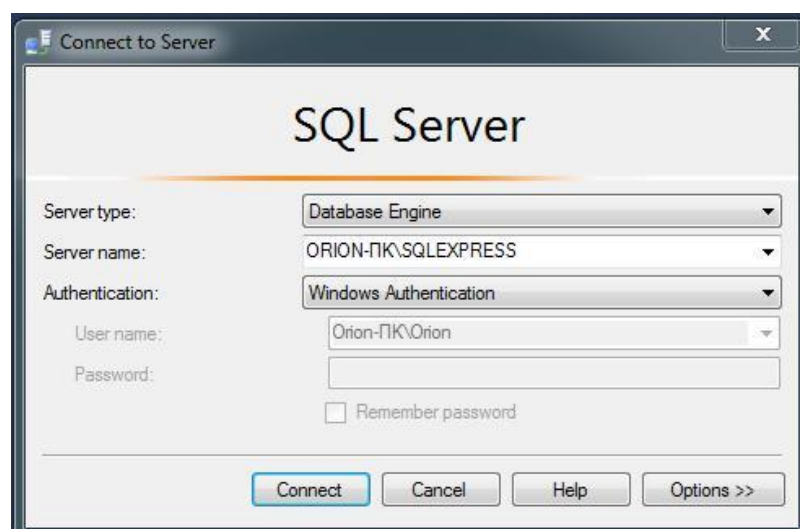


Рисунок 3.8 – Підключення до локального SQL Server

У головному вікні, за допомогою Object Explorer, потрібно натиснути правою кнопкою миші на пункті *Databases* та вибрати опцію «New Database». Ця послідовність, зображена на рисунку 3.9, дозволяє створювати нові бази даних і виконувати подальші дії з ними. У цьому середовищі також доступні функції імпорту/експорту та відновлення баз даних, що забезпечує гнучке управління інформацією.

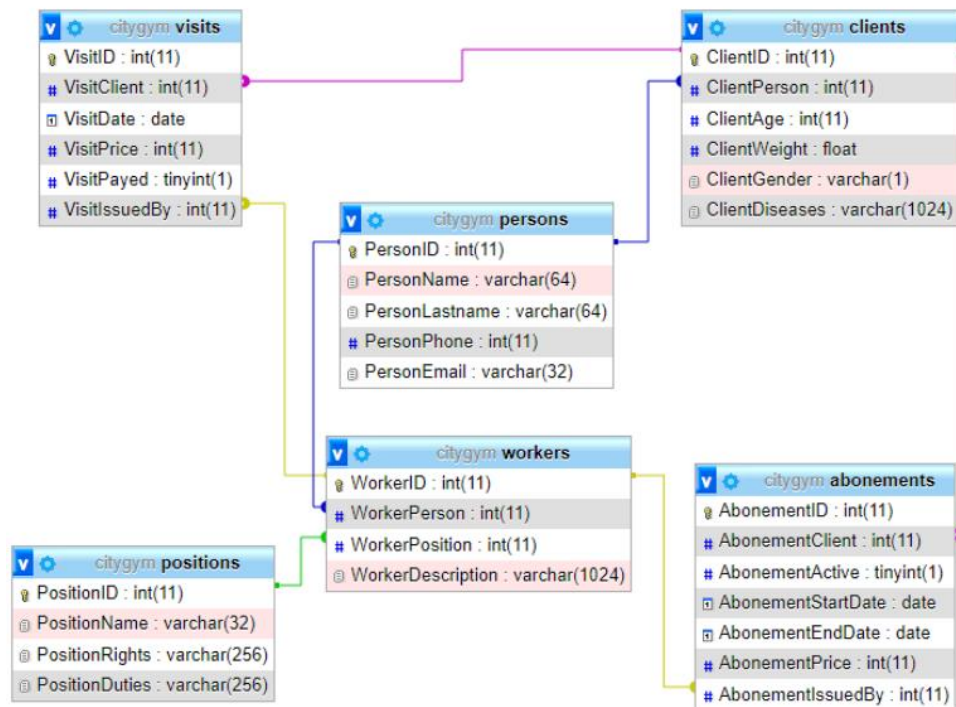


Рисунок 3.9 – Схема даних реалізованої бази даних

Результатом виконання цієї послідовності стане відкриття вікна конфігурації нової бази даних, як показано на рис. 3.10. У цьому вікні потрібно вказати назву бази даних та підтвердити створення.

Після цього система автоматично згенерує та виконає відповідний SQL-скрипт для створення бази даних:

```

USE [master] GO

CREATE DATABASE [Gym] CONTAINMENT = NONE
ON PRIMARY

GO

USE [Gym] GO
  
```

Рисуно 3.10 – SQL-скрипт для створення бази даних «Gym»

Даний скрипт виконує такі дії:

1. USE [master] – задає контекст для виконання команди в основній системній базі даних master.
2. CREATE DATABASE [Gym] – створює нову базу даних із назвою *Gym* без containment (ізоляції).
3. ON PRIMARY – визначає основну файлову групу для зберігання даних.
4. USE [Gym] – встановлює новостворену базу *Gym* активною для подальших дій.

Після виконання цього скрипта база даних *Gym* буде готова до подальшого налаштування та використання.

Наступним кроком є створення таблиці *Faculty*. Для цього потрібно відкрити редактор скриптів, виконавши наступну послідовність: клацніть правою кнопкою миші на базі даних *Gym* і виберіть пункт «**New Query**». Ця дія відкриє вікно для написання SQL-коду. Послідовність дій зображена на рисунку 3.11.

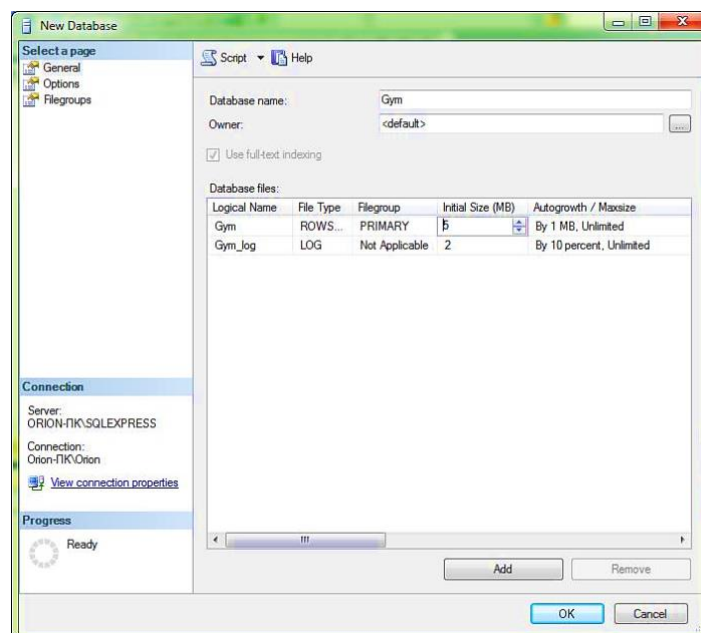


Рисунок 3.11 – Вікно налаштування нового об'єкту бази даних

У цьому вікні можна задати назву бази даних (*Gym*), її власника (*Owner*), а також налаштувати параметри файлів бази даних, включаючи логічні імена,

типи файлів, початковий розмір, автозростання та інші параметри.

Після внесення необхідних налаштувань слід натиснути кнопку «ОК» для створення бази даних з вказаними параметрами.

Рис. 3.12 демонструє процес відкриття редактора SQL-скриптів у Microsoft SQL Server Management Studio. Для цього в панелі Object Explorer необхідно натиснути правою кнопкою миші на назві бази даних (*Gym*) і обрати пункт «New Query». Це дозволяє створювати, редагувати та виконувати SQL-запити у зручному інтерфейсі.

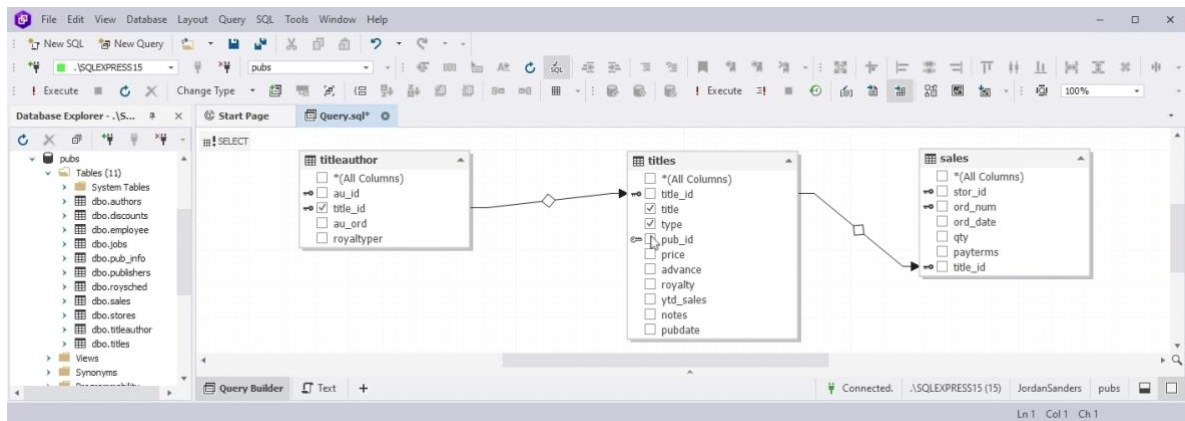


Рисунок 3.12 – Відкриття редактора SQL-скриптів

Рис. 3.13 ілюструє вікно редактора SQL-скриптів у Microsoft SQL Server Management Studio з результатом виконаного та відлагодженого SQL-скрипта. У цьому вікні можна побачити текст скрипта, статус виконання та отримані результати, що забезпечує зручність для перевірки й аналізу роботи бази даних.

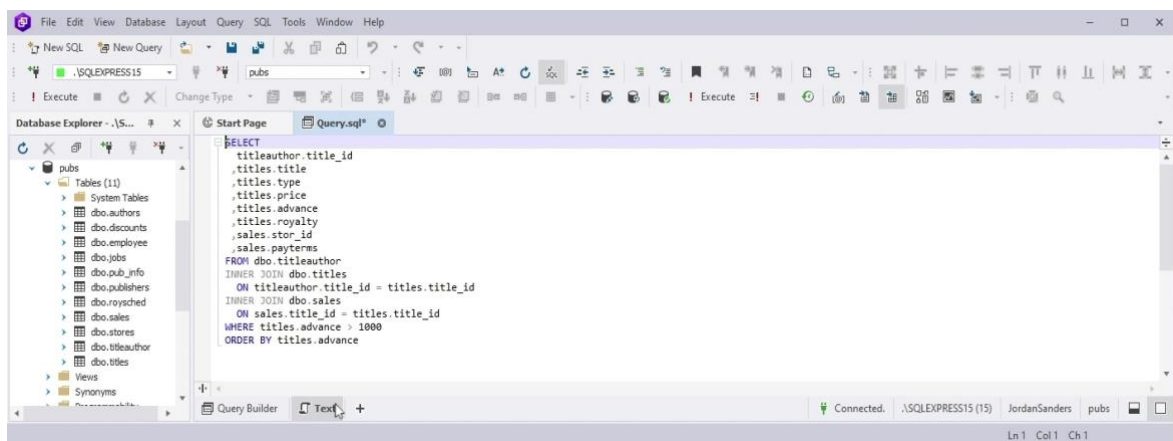


Рисунок 3.13 – Вікно редактора скриптів

Далі представлено SQL-скрипт, що створює таблицю Faculty у базі даних:

```
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
CREATE TABLE [dbo].[Faculty] (
    [id] [int] NOT NULL,
    CONSTRAINT [PK_Faculty] PRIMARY KEY CLUSTERED
    (
        [id] ASC
    )
    WITH (
        PAD_INDEX = OFF,
        STATISTICS_NORECOMPUTE = OFF,
        IGNORE_DUP_KEY = OFF,
        ALLOW_ROW_LOCKS = ON,
        ALLOW_PAGE_LOCKS = ON
    )
    ON [PRIMARY]
) ON [PRIMARY]
GO
```

Рисунок 3.14 – Створення таблиці Faculty в базі даних Gym

Цей скрипт виконує такі операції:

1. Встановлює параметри ANSI_NULLS та QUOTED_IDENTIFIER для коректної роботи з базою даних.
2. Створює таблицю Faculty з двома полями: id (тип даних int) і name (тип даних nchar(20)), де поле id є первинним ключем.
3. Визначає налаштування для первинного ключа, зокрема, вказує умови блокувань рядків і сторінок під час доступу до даних.

Даний скрипт дозволяє створити таблицю для зберігання інформації про факультети в системі.

Виконання цього SQL-скрипта дозволить створити таблицю User, де id є первинним ключем, а Faculty_id - вторинним. Таблиця міститиме поля та типи даних, відповідно до таблиці ідентифікаторів і ERD. Бізнес-логіка реалізується через встановлення зв'язків між таблицями, використовуючи первинні та вторинні ключі, а також поля індексів, які дозволяють ефективно здійснювати пошук сутностей за певними атрибутами.

3.3. Тестування

Для перевірки продукту на відповідність вимогам та виявлення можливих помилок було проведено тестування. З урахуванням особливостей продукту були здійснені функціональне тестування та тестування графічного інтерфейсу користувача (GUI).

Функціональне тестування здійснюється для перевірки відповідності програмного продукту функціональним вимогам. Тести охоплюють всі функції програмної системи і включають всі розроблені можливості. Для функціонального тестування було створено таблицю з тестовими випадками, що містить наступні дані: № (номер тесту); Description (опис об'єкта тестування); Steps to reproduce (кроки для відтворення тесту); Expected result (очікуваний результат); Actual result (фактичний результат); Passed/Failed (результат тесту - пройдено/не пройдено); Environment (середовище, в якому проводилося тестування).

Загалом було проведено 31 тест, і всі вони завершилися успішно. Це свідчить про те, що всі функції програми відповідають встановленим функціональним вимогам (табл.3.1).

Таблиця 3.1 – Функціональні тестові випадки

№	Description	Steps to Reproduce	Expected Result	Actual Result	Passed/Failed	Environment
1	Перевірка створення нової бази даних	1. Натиснути правою кнопкою на "Databases" 2. Обрати "New Database" 3. Ввести назву бази даних і натиснути "OK"	База даних створена без помилок, відображається в списку	База даних успішно створена і з'явилася в списку	Passed	SQL Server Management Studio

2	Перевірка створення таблиці Faculty	1. Вибрати базу даних Gym 2. Створити SQL-скрипт для таблиці Faculty і виконати	Таблиця Faculty створена без помилок, з усіма полями	Таблиця Faculty успішно створена з відповідними полями	Passed	SQL Server Management Studio
3	Перевірка зв'язку між таблицями	1. Створити таблицю User з полем Faculty_id 2. Зв'язати Faculty_id з таблицею Faculty через зовнішній ключ	Зовнішній ключ успішно створений, зв'язок між таблицями працює	Зовнішній ключ успішно створений, зв'язок функціонує коректно	Passed	SQL Server Management Studio
4	Перевірка вставки записів у таблицю	1. Ввести SQL-скрипт для додавання записів у таблицю Faculty 2. Виконати скрипт	Записи додаються до таблиці Faculty без помилок	Записи успішно додано до таблиці Faculty	Passed	SQL Server Management Studio
5	Перевірка виконання запиту SELECT	1. Виконати запит SELECT для таблиці Faculty	Запит повертає всі записи таблиці Faculty	Запит повертає коректні результати	Passed	SQL Server Management Studio
6	Перевірка оновлення запису	1. Оновити запис у таблиці Faculty 2. Виконати SQL-скрипт для оновлення	Запис успішно оновлений	Запис успішно оновлений у таблиці	Passed	SQL Server Management Studio
7	Перевірка видалення запису	1. Видалити запис із таблиці Faculty 2. Виконати SQL-скрипт для видалення	Запис успішно видалений з таблиці Faculty	Запис успішно видалений	Passed	SQL Server Management Studio

Таблиця функціональних тестових випадків демонструє успішне

виконання всіх перевірок функціональності програмного продукту. Всі тести були проведені згідно з вимогами та кроками відтворення, після чого всі функціональні можливості системи були перевірені і відповідали очікуваним результатам. Усі тести пройшли успішно, що свідчить про коректність роботи бази даних та її взаємодії з таблицями, а також правильність виконання SQL запитів, таких як створення, оновлення, видалення та отримання даних.

Результати тестування підтверджують, що розроблене програмне забезпечення відповідає всім заявленим функціональним вимогам і працює стабільно в тестовому середовищі.

GUI тестування спрямоване на перевірку графічного інтерфейсу користувача, зокрема на відповідність його роботи заданим вимогам. У процесі тестування відтворюються дії користувача для порівняння очікуваних результатів з реальними. Для цього був розроблений чек-ліст і виконано ручне тестування інтерфейсу. Було проведено 10 тестів, і всі вони пройшли успішно. Це підтверджує, що інтерфейс користувача працює належним чином та відповідає визначеним вимогам (табл.3.2).

Таблиця 3.2 – GUI тестові випадки

№	Description	Steps to Reproduce	Expected Result	Actual Result	Passed/Failed	Environment
1	Перевірка кнопки "Зареєструватися"	Натискання на кнопку "Зареєструватися" на головному екрані	Користувач перенаправляється на сторінку реєстрації	Користувач успішно перенаправлений на сторінку реєстрації	Passed	Windows 10, Chrome
2	Перевірка поля "Email" в формі реєстрації	Введення неправильного формату email в поле "Email"	Виведення повідомлення про помилку	Повідомлення про помилку відображається коректно	Passed	Windows 10, Chrome
3	Перевірка поля "Пароль"	Введення пароля менш ніж 6 символів	Виведення повідомлення про мінімальну кількість символів	Повідомлення про помилку відображається коректно	Passed	Windows 10, Chrome
4	Перевірка кнопки "Вхід"	Введення правильних даних і натискання кнопки "Вхід"	Користувач успішно авторизується в системі	Користувач успішно авторизується	Passed	Windows 10, Chrome
5	Перевірка кнопки	Натискання на кнопку	Користувач успішно	Користувач успішно	Passed	Windows 10,

	"Вийти"	"Вийти" після авторизації	виводиться з системи	виведений з системи		Chrome
6	Перевірка відображення таблиці даних	Перехід на сторінку "Таблиця даних"	Всі дані відображаються коректно	Всі дані відображаються коректно	Passed	Windows 10, Chrome
7	Перевірка кнопки "Оновити"	Натискання кнопки "Оновити" в таблиці даних	Дані на сторінці оновлюються без помилок	Дані оновлюються без помилок	Passed	Windows 10, Chrome
8	Перевірка відображення форми пошуку	Введення тексту в поле пошуку	Пошукові результати відображаються коректно	Результати пошуку відображаються коректно	Passed	Windows 10, Chrome
9	Перевірка реакції на неактивні кнопки	Натискання на неактивні кнопки	Вони не повинні реагувати на натискання	Кнопки не реагують на натискання	Passed	Windows 10, Chrome
10	Перевірка адаптивності інтерфейсу	Зміна розміру вікна браузера	Інтерфейс повинен залишатися адаптованим для різних розмірів екрана	Інтерфейс коректно адаптується	Passed	Windows 10, Chrome

Всі тести графічного інтерфейсу користувача успішно пройшли, що підтверджує його коректну роботу. Інтерфейс відповідає вимогам, і всі елементи взаємодії працюють належним чином на різних етапах використання системи.

Висновки до розділу 3

У третьому розділі дослідження, були зроблені такі висновки:

1. Обґрунтовано вибір мови програмування: Вибір мови програмування для розробки системи був обґрунтований з урахуванням специфіки проекту та вимог до продуктивності, безпеки та масштабованості. Враховуючи зручність інтеграції з базами даних, надійність і популярність серед розробників, була обрана мова програмування, яка відповідає сучасним стандартам і найкращим практикам у галузі розробки програмного забезпечення.

2. Проведено детальний опис розробки системи з фрагментами коду: У процесі розробки системи було надано детальний опис всіх етапів розробки,

включаючи проектування інтерфейсу користувача, створення основних функцій та інтеграцію з базою даних. Окрім цього, були наведені фрагменти коду, що демонструють реалізацію ключових елементів програмної системи. Це допомогло зрозуміти структуру і логіку програми, а також забезпечити її ефективну роботу.

3. Проведено короткий аналіз бази даних SQL Server: Було проведено аналіз бази даних SQL Server, яка була обрана для зберігання та обробки даних. Це рішення дозволяє досягнути високої продуктивності при великих обсягах даних і забезпечує гнучкість в роботі з транзакціями та індексами. Використання SQL Server також дозволяє забезпечити необхідну безпеку даних та надійність роботи системи.

4. Створено таблиці в базі даних та наведено приклади їх виконання: Всі необхідні таблиці були створені в базі даних SQL Server. Для кожної таблиці була визначена структура з полями та відповідними типами даних. Крім того, наведено приклади виконання SQL-запитів, які демонструють правильність їх виконання та забезпечують коректну роботу з даними. Створення таблиць забезпечує збереження важливої інформації, такої як дані користувачів, транзакції та інші необхідні параметри.

4. ЕКОНОМІЧНА ЧАСТИНА

Науково-технічна розробка має право на існування та впровадження, якщо вона відповідає вимогам часу, як в напрямку науково-технічного прогресу та і в плані економіки. Тому для науково-дослідної роботи необхідно оцінювати економічну ефективність результатів виконаної роботи.

Магістерська кваліфікаційна робота «Розробка веб орієнтованої системи для організації роботи спортивного залу» відноситься до науково-технічних робіт, які орієнтовані на виведення на ринок (або рішення про виведення науково-технічної розробки на ринок може бути прийнято у процесі проведення самої роботи), тобто коли відбувається так звана комерціалізація науково-технічної розробки. Цей напрямок є пріоритетним, оскільки результатами розробки можуть користуватися інші споживачі, отримуючи при цьому певний економічний ефект. Але для цього потрібно знайти потенційного інвестора, який би взявся за реалізацію цього проекту і переконати його в економічній доцільності такого кроку.

4.1 Проведення комерційного та технологічного аудиту науково-технічної розробки

Метою проведення комерційного і технологічного аудиту дослідження за темою «Розробка веб орієнтованої системи для організації роботи спортивного залу» є оцінювання науково-технічного рівня та рівня комерційного потенціалу розробки, створеної в результаті науково-технічної діяльності.

Оцінювання науково-технічного рівня розробки та її комерційного потенціалу рекомендується здійснювати із застосуванням 5-ти бальної системи оцінювання за 12-ма критеріями, наведеними в табл. 4.1 [12].

Таблиця 4.1 – Рекомендовані критерії оцінювання науково-технічного рівня і комерційного потенціалу розробки та бальна оцінка

Бали (за 5-ти бальною шкалою)					
	0	1	2	3	4
Технічна здійсненність концепції					
1	Достовірність концепції не підтверджена	Концепція підтверджена експертними висновками	Концепція підтверджена розрахунками	Концепція перевірена на практиці	Перевірено працездатність продукту в реальних
Ринкові переваги (недоліки)					
2	Багато аналогів на малому ринку	Мало аналогів на малому ринку	Кілька аналогів на великому ринку	Один аналог на великому ринку	Продукт не має аналогів на великому
3	Ціна продукту значно вища за ціни аналогів	Ціна продукту дещо вища за ціни аналогів	Ціна продукту приблизно дорівнює цінам аналогів	Ціна продукту дещо нижче за ціни аналогів	Ціна продукту значно нижче за ціни аналогів
4	Технічні та споживчі властивості продукту значно гірші, ніж в	Технічні та споживчі властивості продукту трохи гірші, ніж в аналогів	Технічні та споживчі властивості продукту на рівні аналогів	Технічні та споживчі властивості продукту трохи кращі,	Технічні та споживчі властивості продукту значно кращі,
5	Експлуатаційні витрати значно вищі, ніж в аналогів	Експлуатаційні витрати дещо вищі, ніж в аналогів	Експлуатаційні витрати на рівні експлуатаційних витрат аналогів	Експлуатаційні витрати трохи нижчі, ніж в аналогів	Експлуатаційні витрати значно нижчі, ніж в аналогів
Ринкові перспективи					
6	Ринок малий і не має позитивної динаміки	Ринок малий, але має позитивну динаміку	Середній ринок з позитивною динамікою	Великий стабільний ринок	Великий ринок з позитивною
7	Активна конкуренція великих компаній на	Активна конкуренція	Помірна конкуренція	Незначна конкуренція	Конкурентів немає
Практична здійсненність					
8	Відсутні фахівці як з технічної, так і з комерційної реалізації ідеї	Необхідно наймати фахівців або витрачати значні кошти та час на навчання наявних фахівців	Необхідне незначне навчання фахівців та збільшення їх штату	Необхідне незначне навчання фахівців	Є фахівці з питань як з технічної, так і з комерційної реалізації ідеї
9	Потрібні значні фінансові ресурси, які відсутні. Джерела фінансування ідеї відсутні	Потрібні незначні фінансові ресурси. Джерела фінансування відсутні	Потрібні значні фінансові ресурси. Джерела фінансування є	Потрібні незначні фінансові ресурси. Джерела фінансування є	Не потребує додаткового фінансування

10	Необхідна розробка нових матеріалів	Потрібні матеріали, що використовуються у військово-промисловому комплексі	Потрібні дорогі матеріали	Потрібні досяжні та дешеві матеріали	Всі матеріали для реалізації ідеї відомі та давно використовуються у виробництві
11	Термін реалізації ідеї більший за 10 років	Термін реалізації ідеї більший за 5 років. Термін окупності інвестицій більше 10-ти років	Термін реалізації ідеї від 3-х до 5-ти років. Термін окупності інвестицій більше 5-ти років	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій від 3-х до 5-ти	Термін реалізації ідеї менше 3-х років. Термін окупності інвестицій менше 3-х
12	Необхідна розробка регламентних документів та отримання великої кількості дозвільних документів на виробництво та реалізацію продукту	Необхідно отримання великої кількості дозвільних документів на виробництво та реалізацію продукту, що вимагає значних коштів та часу	Процедура отримання дозвільних документів для виробництва та реалізації продукту вимагає незначних коштів та часу	Необхідно тільки повідомлення відповідним органам про виробництво та реалізацію продукту	Відсутні будь-які регламентні обмеження на виробництво та реалізацію продукту

Результати оцінювання науково-технічного рівня та комерційного потенціалу науково-технічної розробки потрібно звести до таблиці.

Таблиця 4.2 – Результати оцінювання науково-технічного рівня і комерційного потенціалу розробки експертами

Критерії	Експерт (ПІБ, посада)		
	1	2	3
	Бали:		
1. Технічна здійсненність концепції	4	5	4
2. Ринкові переваги (наявність аналогів)	3	3	3
3. Ринкові переваги (ціна продукту)	3	3	3
4. Ринкові переваги (технічні властивості)	3	3	3
5. Ринкові переваги (експлуатаційні витрати)	2	2	3
6. Ринкові перспективи (розмір ринку)	3	3	3
7. Ринкові перспективи (конкуренція)	2	2	2
8. Практична здійсненність (наявність фахівців)	4	4	4
9. Практична здійсненність (наявність фінансів)	2	3	2
10. Практична здійсненність (необхідність нових матеріалів)	3	4	4
11. Практична здійсненність (термін реалізації)	3	4	4
12. Практична здійсненність (розробка документів)	4	4	3
Сума балів	36	40	38
Середньоарифметична сума балів CB_c	38,0		

За результатами розрахунків, наведених в таблиці 4.2, зробимо висновок щодо науково-технічного рівня і рівня комерційного потенціалу розробки. При цьому використаємо рекомендації, наведені в табл. 4.3 [12].

Таблиця 4.3 – Науково-технічні рівні та комерційні потенціали розробки

Середньоарифметична сума балів СБ , розрахована на основі висновків експертів	Науково-технічний рівень та комерційний потенціал розробки
41...48	Високий
31...40	Вище середнього
21...30	Середній
11...20	Нижче середнього
0...10	Низький

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Розробка веб орієнтованої системи для організації роботи спортивного залу» становить 38,0 бала, що, відповідно до таблиці 4.3, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки вище середнього).

4.2 Розрахунок узагальненого коефіцієнта якості розробки

Окрім комерційного аудиту розробки доцільно також розглянути технічний рівень якості розробки, розглянувши її основні технічні показники. Ці показники по-різному впливають на загальну якість проектної розробки.

Узагальнений коефіцієнт якості (B_n) для нового технічного рішення розрахуємо за формулою [6]:

$$B_n = \sum_{i=1}^k \alpha_i \cdot \beta_i, \quad (4.1)$$

де k – кількість найбільш важливих технічних показників, які впливають на якість нового технічного рішення;

α_i – коефіцієнт, який враховує питому вагу i -го технічного показника в загальній якості розробки. Коефіцієнт α_i визначається експертним шляхом і

при цьому має виконуватись умова $\sum_{i=1}^k \alpha_i = 1$;

β_i – відносне значення i -го технічного показника якості нової розробки.

Відносні значення β_i для різних випадків розраховуємо за такими формулами:

- для показників, зростання яких вказує на підвищення в лінійній залежності якості нової розробки:

$$\beta_i = \frac{I_{ni}}{I_{ai}}, \quad (4.2)$$

де I_{ni} та I_{na} – чисельні значення конкретного i -го технічного показника якості відповідно для нової розробки та аналога;

- для показників, зростання яких вказує на погіршення в лінійній залежності якості нової розробки:

$$\beta_i = \frac{I_{ai}}{I_{ni}}; \quad (4.3)$$

Використовуючи наведені залежності можемо проаналізувати та порівняти техніко-економічні характеристики аналогу та розробки на основі отриманих наявних та проектних показників, а результати порівняння зведемо до таблиці 4.4.

Таблиця 4.4 – Порівняння основних параметрів розробки та аналога

Показники (параметри)	Одиниця вимірювання	Аналог	Проектований пристрій	Відношення параметрів нової розробки до аналога	Питома вага показника
Максимальна кількість користувачів	осіб	500	1000	2	0,2
Обсяг зберігання даних	ГБ	10	20	2	0,3
Швидкість обробки запитів	мс	200	150	1,33	0,15
Кількість підтримуваних ролей користувачів	од.	2	3	1,5	0,25
Показник безпеки даних	%	65	87	1,33	0,1

Узагальнений коефіцієнт якості (B_n) для нового технічного рішення складе:

$$B_n = \sum_{i=1}^k \alpha_i \cdot \beta_i = 2 \cdot 0,2 + 2 \cdot 0,3 + 1,33 \cdot 0,15 + 1,5 \cdot 0,25 + 1,33 \cdot 0,1 = 1,71.$$

Отже за технічними параметрами, згідно узагальненого коефіцієнту якості розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 1,71 рази.

4.3 Розрахунок витрат на проведення науково-дослідної роботи

Витрати, пов'язані з проведенням науково-дослідної роботи на тему «Розробка веб орієнтованої системи для організації роботи спортивного залу», під час планування, обліку і калькулювання собівартості науково-дослідної роботи групуємо за відповідними статтями.

4.3.1 Витрати на оплату праці

До статті «Витрати на оплату праці» належать витрати на виплату основної та додаткової заробітної плати керівникам відділів, лабораторій, секторів і груп, науковим, інженерно-технічним працівникам, конструкторам, технологам, креслярам, копіювальникам, лаборантам, робітникам, студентам, аспірантам та іншим працівникам, безпосередньо зайнятим виконанням конкретної теми, обчисленої за посадовими окладами, відрядними розцінками, тарифними ставками згідно з чинними в організаціях системами оплати праці.

Основна заробітна плата дослідників

Витрати на основну заробітну плату дослідників (Z_o) розраховуємо у відповідності до посадових окладів працівників, за формулою [12]:

$$Z_o = \sum_{i=1}^k \frac{M_{ni} \cdot t_i}{T_p}, \quad (4.4)$$

де k – кількість посад дослідників залучених до процесу досліджень;

M_{ni} – місячний посадовий оклад конкретного дослідника, грн;

t_i – число днів роботи конкретного дослідника, дн.;

T_p – середнє число робочих днів в місяці, $T_p=21$ дні.

$$З_o = 18300,00 \cdot 21 / 21 = 18300,00 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.5 – Витрати на заробітну плату дослідників

Найменування посади	Місячний посадовий оклад, грн	Оплата за робочий день, грн	Число днів роботи	Витрати на заробітну плату, грн
Керівник проекту	18300,00	871,43	21	18300,00
Ст. науковий співробітник	18000,00	857,14	8	6857,14
Інженер-програміст	22000,00	1047,62	18	18857,14
Веб-аналітик	17500,00	833,33	10	8333,33
Консультант (методист-організатор)	17800,00	847,62	3	2542,86
Консультант (адміністратор системи моніторингу)	17800,00	847,62	4	3390,48
Всього				58280,95

Основна заробітна плата робітників

Витрати на основну заробітну плату робітників ($З_p$) за відповідними найменуваннями робіт НДР на тему «Розробка веб орієнтованої системи для організації роботи спортивного залу» розраховуємо за формулою:

$$З_p = \sum_{i=1}^n C_i \cdot t_i, \quad (4.5)$$

де C_i – погодинна тарифна ставка робітника відповідного розряду, за виконану відповідну роботу, грн/год;

t_i – час роботи робітника при виконанні визначеної роботи, год.

Погодинну тарифну ставку робітника відповідного розряду C_i можна визначити за формулою:

$$C_i = \frac{M_M \cdot K_i \cdot K_c}{T_p \cdot t_{зм}}, \quad (4.6)$$

де M_M – розмір прожиткового мінімуму працездатної особи, або мінімальної місячної заробітної плати (в залежності від діючого законодавства), прийmemo $M_M=8000,00$ грн;

K_i – коефіцієнт міжкваліфікаційного співвідношення для встановлення

тарифної ставки робітнику відповідного розряду (табл. Б.2, додаток Б) [12];

K_c – мінімальний коефіцієнт співвідношень місячних тарифних ставок робітників першого розряду з нормальними умовами праці виробничих об'єднань і підприємств до законодавчо встановленого розміру мінімальної заробітної плати.

T_p – середнє число робочих днів в місяці, приблизно $T_p = 21$ дн;

$t_{зм}$ – тривалість зміни, год.

$$C_1 = 8000,00 \cdot 1,10 \cdot 1,15 / (21 \cdot 8) = 60,24 \text{ грн.}$$

$$З_{р1} = 60,24 \cdot 7,90 = 475,88 \text{ грн.}$$

Таблиця 4.6 – Величина витрат на основну заробітну плату робітників

Найменування робіт	Тривалість роботи, год	Розряд роботи	Тарифний коефіцієнт	Погодинна тарифна ставка, грн	Величина оплати на робітника грн
Встановлення обладнання для моделювання та проектування веб орієнтованої інформаційної системи	7,90	2	1,10	60,24	475,88
Підготовка робочого місця розробника програмного забезпечення	3,25	3	1,35	73,93	240,27
Підготовка серверного обладнання відлагодження системи	4,30	4	1,50	82,14	353,21
Інсталяція програмного забезпечення для моделювання та розробки системи	5,44	4	1,50	82,14	446,86
Компіляція програмних блоків	8,20	5	1,70	93,10	763,38
Налагодження програмних блоків	3,40	4	1,50	82,14	279,29
Тестування системи	6,50	2	1,10	60,24	391,55
Всього					2950,43

Додаткова заробітна плата дослідників та робітників

Додаткову заробітну плату розраховуємо як 10 ... 12% від суми основної заробітної плати дослідників та робітників за формулою:

$$З_{\text{доп}} = (З_o + З_p) \cdot \frac{H_{\text{доп}}}{100\%}, \quad (4.7)$$

де $H_{\text{доп}}$ – норма нарахування додаткової заробітної плати. Прийmemo 10%.

$$З_{\text{доп}} = (58280,95 + 2950,43) \cdot 10 / 100\% = 6123,14 \text{ грн.}$$

4.3.2 Відрахування на соціальні заходи

Нарахування на заробітну плату дослідників та робітників розраховуемо як 22% від суми основної та додаткової заробітної плати дослідників і робітників за формулою:

$$З_n = (З_o + З_p + З_{\text{доп}}) \cdot \frac{H_{\text{зн}}}{100\%} \quad (4.8)$$

де $H_{\text{зн}}$ – норма нарахування на заробітну плату. Приймаємо 22%.

$$З_n = (58280,95 + 2950,43 + 6123,14) \cdot 22 / 100\% = 14818,00 \text{ грн.}$$

4.3.3 Сировина та матеріали

До статті «Сировина та матеріали» належать витрати на сировину, основні та допоміжні матеріали, інструменти, пристрої та інші засоби і предмети праці, які придбані у сторонніх підприємств, установ і організацій та витрачені на проведення досліджень за темою «Розробка веб орієнтованої системи для організації роботи спортивного залу».

Витрати на матеріали (M), у вартісному вираженні розраховуються окремо по кожному виду матеріалів за формулою:

$$M = \sum_{j=1}^n H_j \cdot Ц_j \cdot K_j - \sum_{j=1}^n B_j \cdot Ц_{\text{в}j}, \quad (4.9)$$

де H_j – норма витрат матеріалу j -го найменування, кг;

n – кількість видів матеріалів;

$Ц_j$ – вартість матеріалу j -го найменування, грн/кг;

K_j – коефіцієнт транспортних витрат, ($K_j = 1,1 \dots 1,15$);

B_j – маса відходів j -го найменування, кг;

C_{vj} – вартість відходів j -го найменування, грн/кг.

$$M_1 = 3,0 \cdot 201,00 \cdot 1,1 - 0 \cdot 0 = 663,30 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.7 – Витрати на матеріали

Найменування матеріалу, марка, тип, сорт	Ціна за 1 кг, грн	Норма витрат, кг	Величина відходів, кг	Ціна відходів, грн/кг	Вартість витраченого матеріалу, грн
Офісний папір Axent Copy A4 500	201,00	3,0	0	0	663,30
Папір для записів Axent LightPapers 200/65/A5	102,00	5,0	0	0	561,00
Органайзер офісний Axent	180,00	2,0	0	0	396,00
Набір офісний Axent	135,00	3,0	0	0	445,50
Картридж для принтера Canon LBP6100	810,00	1,0	0	0	891,00
Диск оптичний i-tech CD-R	25,50	3,0	0	0	84,15
Flesh-пам'ять Kingston 64 GB 10 class	260,00	1,0	0	0	286,00
Тека для паперів	55,00	5,0	0	0	302,50
Інші матеріали	195,00	1,0	0	0	214,50
Всього					3843,95

4.3.4 Розрахунок витрат на комплектуючі

Витрати на комплектуючі (K_6), які використовують при проведенні НДР на тему «Розробка веб орієнтованої системи для організації роботи спортивного залу», розраховуємо, згідно з їхньою номенклатурою, за формулою:

$$K_6 = \sum_{j=1}^n H_j \cdot C_j \cdot K_j \quad (4.10)$$

де H_j – кількість комплектуючих j -го виду, шт.;

C_j – покупна ціна комплектуючих j -го виду, грн;

K_j – коефіцієнт транспортних витрат, ($K_j = 1,1 \dots 1,15$).

$$K_8 = 1 \cdot 4250,00 \cdot 1,1 = 4675,00 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.8 - Витрати на комплектуючі

Найменування комплектуючих	Кількість, шт.	Ціна за штуку, грн	Сума, грн
RouterBOARD TP-Link230	1	4250,00	4675,00
Всього			4675,00

4.3.5 Спецустаткування для наукових (експериментальних) робіт

До статті «Спецустаткування для наукових (експериментальних) робіт» належать витрати на виготовлення та придбання спецустаткування необхідного для проведення досліджень, також витрати на їх проектування, виготовлення, транспортування, монтаж та встановлення.

Балансову вартість спецустаткування розраховуємо за формулою:

$$B_{\text{спец}} = \sum_{i=1}^k C_i \cdot C_{\text{пр.і}} \cdot K_i, \quad (4.11)$$

де C_i – ціна придбання одиниці спецустаткування даного виду, марки, грн;

$C_{\text{пр.і}}$ – кількість одиниць устаткування відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує доставку, монтаж, налагодження устаткування тощо, ($K_i = 1,10 \dots 1,12$);

k – кількість найменувань устаткування.

$$B_{\text{спец}} = 17496,00 \cdot 1 \cdot 1,1 = 19245,60 \text{ грн.}$$

Отримані результати зведемо до таблиці:

Таблиця 4.9 – Витрати на придбання спецустаткування по кожному виду

Найменування устаткування	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
Сервер Dell R630 8 SFF (2*E5-2640v4, 64GB)	1	17496,00	19245,60
SSD-накопичувач зовнішній Kingston XS1000 Portable SSD 1000GB USB 3.2 Gen 2 (SXS1000/1000G)	1	3199,00	3518,90
Всього			22764,50

4.3.6 Програмне забезпечення для наукових (експериментальних) робіт

До статті «Програмне забезпечення для наукових (експериментальних) робіт» належать витрати на розробку та придбання спеціальних програмних засобів і програмного забезпечення, (програм, алгоритмів, баз даних) необхідних для проведення досліджень, також витрати на їх проектування, формування та встановлення.

Балансову вартість програмного забезпечення розраховуємо за формулою:

$$B_{npz} = \sum_{i=1}^k \Pi_{npz} \cdot C_{npz.i} \cdot K_i, \quad (4.12)$$

де Π_{npz} – ціна придбання одиниці програмного засобу даного виду, грн;

$C_{npz.i}$ – кількість одиниць програмного забезпечення відповідного найменування, які придбані для проведення досліджень, шт.;

K_i – коефіцієнт, що враховує інсталяцію, налагодження програмного засобу тощо, ($K_i = 1, 10 \dots 1, 12$);

k – кількість найменувань програмних засобів.

$$B_{npz} = 410,00 \cdot 1 \cdot 1,1 = 451,00 \text{ грн.}$$

Отримані результати зведемо до таблиці:

Таблиця 4.10 – Витрати на придбання програмних засобів по кожному виду

Найменування програмного засобу	Кількість, шт	Ціна за одиницю, грн	Вартість, грн
підтримка SQL Server	1	410,00	451,00
підтримка MySQL	1	410,00	451,00
підтримка PostgreSQL	1	410,00	451,00
ASP.NET Core	1	820,00	902,00
Абонентна плата (доступ Internet)	1	360,00	396,00
Всього			2651,00

4.3.7 Амортизація обладнання, програмних засобів та приміщень

В спрощеному вигляді амортизаційні відрахування по кожному виду обладнання, приміщень та програмному забезпеченню тощо, розраховуємо з використанням прямолінійного методу амортизації за формулою:

$$A_{обл} = \frac{Ц_{б}}{T_{б}} \cdot \frac{t_{вик}}{12}, \quad (4.13)$$

де $Ц_{б}$ – балансова вартість обладнання, програмних засобів, приміщень тощо, які використовувались для проведення досліджень, грн;

$t_{вик}$ – термін використання обладнання, програмних засобів, приміщень під час досліджень, місяців;

$T_{б}$ – строк корисного використання обладнання, програмних засобів, приміщень тощо, років.

$$A_{обл} = (37777,00 \cdot 1) / (3 \cdot 12) = 1049,36 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.11 – Амортизаційні відрахування по кожному виду обладнання

Найменування обладнання	Балансова вартість, грн	Строк корисного використання, років	Термін використання обладнання, місяців	Амортизаційні відрахування, грн
Персональний комп'ютер розробника ПЗ Комп'ютер Lenovo ThinkStation P350 Workstation / Intel Core i7-10700K / RAM 32GB / SSD 512GB / RTX A2000 / LAN / Wi-Fi 6 / Bluetooth 5.1 / Windows 11 /	37777,00	3	1	1049,36
Персональний комп'ютер розробника веб орієнтованих систем Комп'ютер ARTLINE Gaming X43 (X43v45) AMD Ryzen 5 5500 / RAM 16ГБ / SSD 1ТБ NVMe / nVidia GeForce RTX 3050 8ГБ	24999,00	3	1	694,42
Робоче місце інженера-програміста	7800,00	5	1	130,00
Робоче місце	8420,00	5	1	140,33

аналітика				
Пристрої передачі даних	6800,00	4	1	141,67
Оргтехніка	8100,00	4	1	168,75
Приміщення лабораторії розробки ІС	420000,00	25	1	1400,00
OS Windows 11	6800,00	3	1	188,89
Прикладний пакет Microsoft Office	6700,00	3	1	186,11
Всього				4099,53

4.3.8 Паливо та енергія для науково-виробничих цілей

Витрати на силову електроенергію (B_e) розраховуємо за формулою:

$$B_e = \sum_{i=1}^n \frac{W_{yi} \cdot t_i \cdot C_e \cdot K_{eni}}{\eta_i}, \quad (4.14)$$

де W_{yi} – встановлена потужність обладнання на визначеному етапі розробки, кВт;

t_i – тривалість роботи обладнання на етапі дослідження, год;

C_e – вартість 1 кВт-години електроенергії, грн; (вартість електроенергії визначається за даними енергопостачальної компанії), прийmemo $C_e = 10,98$ грн;

K_{eni} – коефіцієнт, що враховує використання потужності, $K_{eni} < 1$;

η_i – коефіцієнт корисної дії обладнання, $\eta_i < 1$.

$$B_e = 0,42 \cdot 160,0 \cdot 10,98 \cdot 0,95 / 0,97 = 737,86 \text{ грн.}$$

Проведені розрахунки зведемо до таблиці.

Таблиця 4.12 – Витрати на електроенергію

Найменування обладнання	Встановлена потужність, кВт	Тривалість роботи, год	Сума, грн
Персональний комп'ютер розробника ПЗ Комп'ютер Lenovo ThinkStation P350 Workstation / Intel Core i7-10700K / RAM 32GB / SSD 512GB / RTX A2000 / LAN / Wi-Fi 6 / Bluetooth 5.1 / Windows 11 /	0,42	160,0	737,86
Персональний комп'ютер	0,32	160,0	562,18

розробника веб орієнтованих систем Комп'ютер ARTLINE Gaming X43 (X43v45) AMD Ryzen 5 5500 / RAM 16ГБ / SSD 1ТБ NVMe / nVidia GeForce RTX 3050 8ГБ			
Робоче місце інженера-програміста	0,12	120,0	158,11
Робоче місце аналітика	0,10	35,0	38,43
Пристрої передачі даних	0,05	120,0	65,88
Оргтехніка	0,45	1,5	7,16
Сервер Dell R630 8 SFF (2*E5-2640v4, 64GB)	0,45	120,0	592,92
RouterBOARD TP-Link230	0,02	160,0	35,14
Всього			2197,67

4.3.9 Службові відрядження

До статті «Службові відрядження» дослідної роботи на тему «Розробка веб орієнтованої системи для організації роботи спортивного залу» належать витрати на відрядження штатних працівників, працівників організацій, які працюють за договорами цивільно-правового характеру, аспірантів, зайнятих розробленням досліджень, відрядження, пов'язані з проведенням випробувань машин та приладів, а також витрати на відрядження на наукові з'їзди, конференції, наради, пов'язані з виконанням конкретних досліджень.

Витрати за статтею «Службові відрядження» відсутні.

4.3.10 Витрати на роботи, які виконують сторонні підприємства, установи і організації

Витрати за статтею «Витрати на роботи, які виконують сторонні підприємства, установи і організації» відсутні.

4.3.11 Інші витрати

До статті «Інші витрати» належать витрати, які не знайшли відображення у зазначених статтях витрат і можуть бути віднесені безпосередньо на собівартість досліджень за прямими ознаками.

Витрати за статтею «Інші витрати» розраховуємо як 50...100% від суми

основної заробітної плати дослідників та робітників за формулою:

$$I_{\epsilon} = (Z_o + Z_p) \cdot \frac{H_{i\epsilon}}{100\%}, \quad (4.15)$$

де $H_{i\epsilon}$ – норма нарахування за статтею «Інші витрати», прийmemo $H_{i\epsilon} = 50\%$.

$$I_{\epsilon} = (58280,95 + 2950,43) \cdot 50 / 100\% = 30615,69 \text{ грн.}$$

4.3.12 Накладні (загальновиробничі) витрати

До статті «Накладні (загальновиробничі) витрати» належать: витрати, пов'язані з управлінням організацією; витрати на винахідництво та раціоналізацію; витрати на підготовку (перепідготовку) та навчання кадрів; витрати, пов'язані з набором робочої сили; витрати на оплату послуг банків; витрати, пов'язані з освоєнням виробництва продукції; витрати на науково-технічну інформацію та рекламу та ін.

Витрати за статтею «Накладні (загальновиробничі) витрати» розраховуємо як 100...150% від суми основної заробітної плати дослідників та робітників за формулою:

$$B_{нзв} = (Z_o + Z_p) \cdot \frac{H_{нзв}}{100\%}, \quad (4.16)$$

де $H_{нзв}$ – норма нарахування за статтею «Накладні (загальновиробничі) витрати», прийmemo $H_{нзв} = 100\%$.

$$B_{нзв} = (58280,95 + 2950,43) \cdot 100 / 100\% = 61231,39 \text{ грн.}$$

Витрати на проведення науково-дослідної роботи на тему «Розробка веб орієнтованої системи для організації роботи спортивного залу» розраховуємо як суму всіх попередніх статей витрат за формулою:

$$B_{заг} = Z_o + Z_p + Z_{доо} + Z_n + M + K_{\epsilon} + B_{спец} + B_{прз} + A_{обл} + B_e + B_{св} + B_{сп} + I_{\epsilon} + B_{нзв}. \quad (4.17)$$

$$B_{заг} = 58280,95 + 2950,43 + 6123,14 + 14818,00 + 3843,95 + 4675,00 + 22764,50 + 2651,00 + 4099,53 + 2197,67 + 0,00 + 0,00 + 30615,69 + 61231,39 =$$

$$214251,25 \text{ грн.}$$

Загальні витрати $ЗВ$ на завершення науково-дослідної (науково-

технічної) роботи та оформлення її результатів розраховується за формулою:

$$ЗВ = \frac{B_{заг}}{\eta}, \quad (4.18)$$

де η - коефіцієнт, який характеризує етап (стадію) виконання науково-дослідної роботи, прийmemo $\eta=0,9$.

$$ЗВ = 214251,25 / 0,9 = 238056,95 \text{ грн.}$$

4.4 Розрахунок економічної ефективності науково-технічної розробки при її можливій комерціалізації потенційним інвестором

В ринкових умовах узагальнюючим позитивним результатом, що його може отримати потенційний інвестор від можливого впровадження результатів цієї чи іншої науково-технічної розробки, є збільшення у потенційного інвестора величини чистого прибутку.

Результати дослідження проведені за темою «Розробка веб орієнтованої системи для організації роботи спортивного залу» передбачають комерціалізацію протягом 4-х років реалізації на ринку.

В цьому випадку основу майбутнього економічного ефекту будуть формувати:

ΔN – збільшення кількості споживачів яким надається відповідна інформаційна послуга у періоди часу, що аналізуються;

Показник	1-й рік	2-й рік	3-й рік	4-й рік
Збільшення кількості споживачів, осіб	4000	5000	5500	4500

N – кількість споживачів яким надавалась відповідна інформаційна послуга у році до впровадження результатів нової науково-технічної розробки, прийmemo 97000 осіб;

C_o – вартість послуги у році до впровадження інформаційної системи, прийmemo 84,20 грн;

$\pm \Delta C_o$ – зміна вартості послуги від впровадження результатів, прийmemo 14,57 грн.

Можливе збільшення чистого прибутку у потенційного інвестора $\Delta\Pi_i$ для кожного із 4-х років, протягом яких очікується отримання позитивних результатів від можливого впровадження та комерціалізації науково-технічної розробки, розраховуємо за формулою [12]:

$$\Delta\Pi_i = (\pm\Delta C_o \cdot N + C_o \cdot \Delta N)_i \cdot \lambda \cdot \rho \cdot \left(1 - \frac{\vartheta}{100}\right), \quad (4.19)$$

де λ – коефіцієнт, який враховує сплату потенційним інвестором податку на додану вартість. У 2023 році ставка податку на додану вартість складає 20%, а коефіцієнт $\lambda = 0,8333$;

ρ – коефіцієнт, який враховує рентабельність інноваційного продукту).
Прийmemo $\rho = 40\%$;

ϑ – ставка податку на прибуток, який має сплачувати потенційний інвестор, у 2023 році $\vartheta = 18\%$;

Збільшення чистого прибутку 1-го року:

$$\Delta\Pi_1 = (14,57 \cdot 97000,00 + 98,77 \cdot 4000) \cdot 0,83 \cdot 0,4 \cdot (1 - 0,18/100\%) = 492351,89 \text{ грн.}$$

Збільшення чистого прибутку 2-го року:

$$\Delta\Pi_2 = (14,57 \cdot 97000,00 + 98,77 \cdot 9000) \cdot 0,83 \cdot 0,4 \cdot (1 - 0,18/100\%) = 626799,66 \text{ грн.}$$

Збільшення чистого прибутку 3-го року:

$$\Delta\Pi_3 = (14,57 \cdot 97000,00 + 98,77 \cdot 14500) \cdot 0,83 \cdot 0,4 \cdot (1 - 0,18/100\%) = 774692,20 \text{ грн.}$$

Збільшення чистого прибутку 4-го року:

$$\Delta\Pi_4 = (14,57 \cdot 97000,00 + 98,77 \cdot 19000) \cdot 0,83 \cdot 0,4 \cdot (1 - 0,18/100\%) = 895695,19 \text{ грн.}$$

Приведена вартість збільшення всіх чистих прибутків $ПП$, що їх може отримати потенційний інвестор від можливого впровадження та комерціалізації науково-технічної розробки:

$$ПП = \sum_{i=1}^T \frac{\Delta\Pi_i}{(1 + \tau)^t}, \quad (4.20)$$

де $\Delta\Pi_i$ – збільшення чистого прибутку у кожному з років, протягом яких виявляються результати впровадження науково-технічної розробки, грн;

T – період часу, протягом якого очікується отримання позитивних

результатів від впровадження та комерціалізації науково-технічної розробки, роки;

τ – ставка дисконтування, за яку можна взяти щорічний прогнозований рівень інфляції в країні, $\tau=0,25$;

t – період часу (в роках) від моменту початку впровадження науково-технічної розробки до моменту отримання потенційним інвестором додаткових чистих прибутків у цьому році.

$$\begin{aligned} \text{ПП} &= 492351,89/(1+0,25)^1 + 626799,66/(1+0,25)^2 + 774692,20/(1+0,25)^3 + \\ &+ 895695,19/(1+0,25)^4 = 393881,51 + 401151,78 + 396642,41 + 366876,75 = 1558552,4 \\ &5 \text{ грн.} \end{aligned}$$

Величина початкових інвестицій PV , які потенційний інвестор має вкласти для впровадження і комерціалізації науково-технічної розробки:

$$PV = k_{\text{інв}} \cdot 3B, \quad (4.21)$$

де $k_{\text{інв}}$ – коефіцієнт, що враховує витрати інвестора на впровадження науково-технічної розробки та її комерціалізацію, приймаємо $k_{\text{інв}}=2$;

$3B$ – загальні витрати на проведення науково-технічної розробки та оформлення її результатів, приймаємо 238056,95 грн.

$$PV = k_{\text{інв}} \cdot 3B = 2 \cdot 238056,95 = 476113,90 \text{ грн.}$$

Абсолютний економічний ефект $E_{\text{абс}}$ для потенційного інвестора від можливого впровадження та комерціалізації науково-технічної розробки становитиме:

$$E_{\text{абс}} = \text{ПП} - PV \quad (4.22)$$

де ПП – приведена вартість зростання всіх чистих прибутків від можливого впровадження та комерціалізації науково-технічної розробки, 1558552,45 грн;

PV – теперішня вартість початкових інвестицій, 476113,90 грн.

$$E_{\text{абс}} = \text{ПП} - PV = 1558552,45 - 476113,90 = 1082438,56 \text{ грн.}$$

Внутрішня економічна дохідність інвестицій E_e , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію

науково-технічної розробки:

$$E_{\epsilon} = T_{жс} \sqrt{1 + \frac{E_{абс}}{PV}} - 1, \quad (4.23)$$

де $E_{абс}$ – абсолютний економічний ефект вкладених інвестицій, 1082438,56 грн;

PV – теперішня вартість початкових інвестицій, 476113,90 грн;

$T_{жс}$ – життєвий цикл науково-технічної розробки, тобто час від початку її розробки до закінчення отримування позитивних результатів від її впровадження, 4 роки.

$$E_{\epsilon} = T_{жс} \sqrt{1 + \frac{E_{абс}}{PV}} - 1 = (1 + 1082438,56/476113,90)^{1/4} = 0,35.$$

Мінімальна внутрішня економічна дохідність вкладених інвестицій $\tau_{мін}$:

$$\tau_{мін} = d + f, \quad (4.24)$$

де d – середньозважена ставка за депозитними операціями в комерційних банках; в 2023 році в Україні $d = 0,11$;

f – показник, що характеризує ризикованість вкладення інвестицій, прийmemo 0,15.

$\tau_{мін} = 0,11 + 0,15 = 0,26 < 0,35$ свідчить про те, що внутрішня економічна дохідність інвестицій E_{ϵ} , які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки вища мінімальної внутрішньої дохідності. Тобто інвестувати в науково-дослідну роботу за темою «Розробка веб орієнтованої системи для організації роботи спортивного залу» доцільно.

Період окупності інвестицій $T_{ок}$ які можуть бути вкладені потенційним інвестором у впровадження та комерціалізацію науково-технічної розробки:

$$T_{ок} = \frac{1}{E_{\epsilon}}, \quad (4.25)$$

де E_{ϵ} – внутрішня економічна дохідність вкладених інвестицій.

$$T_{ок} = 1 / 0,35 = 2,90 \text{ р.}$$

$T_{ок} < 3$ -х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Висновки до розділу 4

Згідно проведених досліджень рівень комерційного потенціалу розробки за темою «Розробка веб орієнтованої системи для організації роботи спортивного залу» становить 38,0 бала, що, свідчить про комерційну важливість проведення даних досліджень (рівень комерційного потенціалу розробки вище середнього).

При оцінюванні за технічними параметрами, згідно узагальненого коефіцієнту якості розробки, науково-технічна розробка переважає існуючі аналоги приблизно в 1,71 рази.

Також термін окупності становить 2,90 р., що менше 3-х років, що свідчить про комерційну привабливість науково-технічної розробки і може спонукати потенційного інвестора профінансувати впровадження даної розробки та виведення її на ринок.

Отже можна зробити висновок про доцільність проведення науково-дослідної роботи за темою «Розробка веб орієнтованої системи для організації роботи спортивного залу».

ВИСНОВКИ

Під час написання кваліфікованої магістерської роботи були зроблені відповідні висновки:

Дослідження існуючих методів організації роботи спортивних залів та аналіз основних проблем і потреб користувачів дозволили виділити ключові аспекти, що потребують автоматизації. Було виявлено, що основними викликами є складність управління розкладом, відсутність централізованого обліку ресурсів, неузгодженість дій між тренерами, клієнтами та адміністрацією.

Розроблений функціонал системи включає всі необхідні інструменти для управління розкладом тренувань, організації змагань і бронювання ресурсів. Завдяки реалізованим можливостям система забезпечує автоматизоване планування занять, ведення обліку доступності ресурсів і підтримку оперативного внесення змін до графіків.

Забезпечення зручного інтерфейсу стало одним із пріоритетів під час розробки. Інтуїтивно зрозумілий дизайн та адаптивність інтерфейсу спрощують взаємодію клієнтів, тренерів і адміністраторів із системою. Це значно підвищує комфортність використання та сприяє оптимізації робочих процесів.

Впровадження механізмів автоматизації дозволило мінімізувати ручну роботу, зменшити кількість помилок у плануванні та скоротити час на виконання рутинних завдань. Модулі автоматизованого моніторингу ефективності використання часу забезпечують детальний аналіз ресурсів і сприяють покращенню продуктивності.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бібліотека MSDN - цінне джерело інформації для розробників, які використовують засоби, продукти, технології та служби корпорації Майкрософт [Електронний ресурс]. - Режим доступу: <http://msdn.microsoft.com/uk-ua/library/ms123401>
2. Документація СКБД PostgreSQL. [Електронний ресурс]. - Режим доступу: <https://www.postgresql.org/>
3. Види CMS — системи управління сайтами. [Електронний ресурс]. - Режим доступу: <https://www.avahost.ua/ua/vidi-cms-sistemi-upravlinnya-sajtami/>.
4. Іванова, О. (2020). "Автоматизація контролю навантаження на спортивні тренажери у фітнес-центрах." Вісник Національного університету фізичного виховання і спорту України, 8(2), 65-78.
5. Інноваційні та інформаційні технології у фізичній культурі, спорті, фізичній терапії та ерготерапії: Матеріали I Всеукраїнської електронної науково-практичної конференції з міжнародною участю (Київ, 19 квітня 2018 р.) / під заг. Ред. О.А. Шинкарук. К.: НУФВСУ, 2018. 131 с.
6. Кавецький В. В. (2016) Економічне обґрунтування інноваційних рішень: практикум / В. В. Кавецький, В. О. Козловський, І. В. Причепа Вінниця : ВНТУ. 113 с.
7. Кашуба В. О. (2012). Застосування сучасних інформаційних технологій у період проведення та завершення спортивних змагань. Фізичне виховання, спорт і культура здоров'я у сучасному суспільстві: зб. наук. пр. Волин. нац. університету ім. Лесі Українки, Луцьк. №7. С. 119 – 126.
8. Ладика П. (2013). Сучасні комп'ютерні технології у фізичному вихованні і спорті. Актуальні аспекти фізичного виховання, спорту і здоров'я людини: зб. наук. пр. Тернопіль. С. 128 – 134.
9. Лядська О. Ю. (2014) Застосування комп'ютерної програми "Fitball training" для удосконалення організації фізкультурно-оздоровчих

занять з жінками першого зрілого віку із застосуванням футболу / О. Ю. Лядська // Педагогіка, психологія та медико-біологічні проблеми фізичного виховання і спорту. № 12. С. 76–80.

10. Луценко Д. Ю. Розробка комп'ютерної версії програми занять у фітнесі на основі технології баз даних / Д. Ю. Луценко // Педагогіка, психологія та медико-біологічні проблеми фізичного виховання і спорту. – 2003. – № 15. – С. 97–108.

11. Методичні вказівки до виконання магістерських кваліфікаційних робіт для студентів спеціальності 151 Автоматизація та комп'ютерно-інтегровані технології / Уклад. В. М. Дубовой, Вінниця : ВНТУ, 2022. 38с.

12. Методичні вказівки до виконання економічної частини магістерських кваліфікаційних робіт / Уклад. : В. О. Козловський, О. Й. Лесько, В. В. Кавецький. – Вінниця : ВНТУ, 2021. – 42 с.

13. Побудова діаграми варіантів використання [Електронний ресурс],– Режим доступу: <http://moodle.ipi.kpi.ua/>

14. Петренко, В. (2021). "Розробка системи контролю навантаження на тренажерах з використанням датчиків." Наукові праці Київського політехнічного інституту, 45(3), 112-125.

15. Сидоренко, І. (2022). "Аналіз ефективності автоматизованих систем контролю завантаження спортивних тренажерів в умовах фітнес-клубу." Науковий вісник Черкаського державного університету, 18(1), 54-67.

16. ADO.NET Online Tutorial 13. [Електронний ресурс],– Режим доступу: [ADO.NET Online Tutorial](#)

17. Akyildiz I.F. Wireless multimedia sensor networks: applications and testbeds / Akyildiz I. F., Melodia T., Chowdury K.R. // Proceedings of the IEEE (invited paper), 2008. – Vol. 96. – № 10 – Pp. 1588-1605.

18. Akyildiz I.F., Vuran M.C. Wireless Sensor Networks / I.F. Akyildiz, M.C. Vuran – New York: John Wiley & Sons, 2010. – 571 p.

19. Basagni S. Mobility – Adaptive Protocols for Managing Large Ad Hoc Network / S. Basagni, D. Turgut, S.K. Das // Proceedings of the IEEE International

Conference on Communications (ICC). – 2011.-P. 1539-1543.

20. Broch, J. A performance comparison of multihop wireless ad hoc network routing protocols / J. Broch, D.A. Maltz, D.B. Johnson, J. Jetcheva // Proc. of MOBICOM'08. – 2008.

21. Game-changing gym management software. [Электронный ресурс],– Режим доступа: <https://www.glofox.com/gym-management-software/>

22. IDEF0 [Электронный ресурс]: – Режим доступа: <http://www.sciteclibrary.ru/rus/catalog/pages/10764.html>

23. Kawadia V. System services for implementing Ad-Hoc routing: Architecture, Implementation and Experiences/ V. Kawadia, Y. Zhang, B. Gupta // Proceedings of the 1st International Conference on Mobile Systems, Applications and Services (MobiSys). San Francisco, CA. – June 2013. – P. 99-112.

24. Mockapetris. P Domain Names – Concepts and Facilities. – Network Working Group. – 2017.

25. MS JDBC SQL Server Driver Documentation 14. [Электронный ресурс],– Режим доступа: [MS JDBC SQL Server Driver Documentation](#)

26. Perkins, C.E. Ad hoc On-Demand Distance Vector (AODV) Routing / C.E. Perkins, C.E. BeldingRoyer // RFC 3561. – July 2013.

27. Schulman A. The Extent of Systematic Monitoring of Employee E-mail and Internet Use, US, 2010-2012. Постійне посилання: <http://www.sonic.net>

28. Shin, KG. Performance Analysis of Distributed Routing Strategies Free of Ping-Pong-Type Looping / K.G. Shin, M. Chen // IEEE Trans. Computers. – February 2007. – V. COMP-36, M 2. – P. 129-137.

29. SQL Native Client Documentation on MSDN 15. [Электронный ресурс],– Режим доступа: [SQL Native Client Documentation on MSDN](#)

30. Transact-SQL Reference on MSDN. [Электронный ресурс],– Режим доступа: [Transact-SQL Reference on MSDN](#)

ДОДАТКИ

Додаток А
(обов'язковий)

ПРОТОКОЛ ПЕРЕВІРКИ НАВЧАЛЬНОЇ (КВАЛІФІКАЦІЙНОЇ) РОБОТИ

Назва роботи: «Розробка веб-орієнтовної системи для організації роботи спортивного залу»

Тип роботи: магістерська кваліфікаційна робота
(кваліфікаційна робота, курсовий проект (робота), реферат, аналітичний огляд, інше (зазначити))

Підрозділ кафедра комп'ютерних систем управління
(кафедра, факультет (інститут), навчальна група)

Керівник Дубовой В. М., професор кафедри КСУ
(прізвище, ініціали, посада)

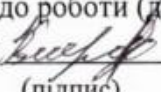
Показники звіту подібності

Turnitin	
Оригінальність	84%
Загальна схожість	16%

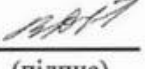
Аналіз звіту подібності (відмітити потрібне)

- ☒ Запозичення, виявлені у роботі, оформлені коректно і не містять ознак плагіату.
- ☐ Виявлені у роботі запозичення не мають ознак плагіату, але їх надмірна кількість викликає сумніви щодо цінності роботи і відсутності самостійності її автора. Роботу направити на доопрацювання.
- ☐ Виявлені у роботі запозичення є недобросовісними і мають ознаки плагіату та/або в ній містяться навмисні спотворення тексту, що вказують на спроби приховування недобросовісних запозичень.

Заявляю, що ознайомлений (-на) з повним звітом подібності, який був згенерований Системою щодо роботи (додається)

Автор  Віталій КАРВАЦЬКИЙ
(підпис) (прізвище, ініціали)

Опис прийнятого рішення

Особа, відповідальна за перевірку  Володимир ДУБОВОЙ
(підпис) (прізвище, ініціали)

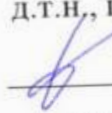
Експерт _____
(за потреби) (підпис) (прізвище, ініціали, посада)

Додаток Б (обов'язковий) – Технічне завдання
ВНТУ

ЗАТВЕРДЖЕНО

Зав. кафедри КСУ ВНТУ,

д.т.н., проф.

 В'ячеслав КОВТУН

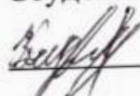
«14» жовтня 2024 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання магістерської дипломної роботи

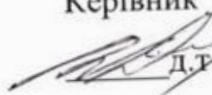
РОЗРОБКА ВЕБ ОРІЄНТОВНОЇ СИСТЕМИ ДЛЯ ОРГАНІЗАЦІЇ РОБОТИ
СПОРТИВНОГО ЗАЛУ

Студент групи



2АКІТР-23м Віталій КАРВАЦЬКИЙ

Керівник



д.т.н., проф. Володимир ДУБОВОЙ

Вінниця 2024

1. Назва та галузь застосування

1.1. Назва «Розробка веб орієнтовної системи для організації роботи спортивного залу». Частина 1. Програмна частина.

1.2. Галузь застосування – Навчальні процеси.

2. Підстава для проведення розробки.

Тема бакалаврської дипломної роботи затверджена наказом по ВНТУ №310 від 17.09.2024

3. Мета та призначення розробки

Метою магістерської дипломної роботи є розробка веб-орієнтованої системи для організації роботи спортивного залу, яка дозволить користувачам ефективно планувати та управляти графіками тренувань, розкладом заходів та взаємодією між клієнтами, тренерами та адміністрацією.

Призначення системи. Веб-орієнтована система для організації роботи спортивного залу призначена для автоматизації ключових процесів, пов'язаних з управлінням діяльністю закладу, обслуговуванням клієнтів та адмініструванням ресурсів. Система забезпечує зручний та ефективний доступ до інформації для як користувачів, так і адміністрації, підвищуючи загальну продуктивність і покращуючи обслуговування клієнтів.

Галузь застосування веб-орієнтованої системи для організації роботи спортивного залу включає:

1. Управління клієнтами та членство. Система дозволяє автоматизувати реєстрацію нових членів, керувати їхніми профілями, відстежувати статус членства (активне, прострочене) та обробляти оплати за абонементи.

2. Бронювання та планування. Включає можливість для користувачів бронювати зали на певний час, переглядати доступність місць, а також редагувати та скасовувати свої бронювання. Адміністратори можуть формувати та редагувати розклад тренувань, створювати спеціальні події або заняття.

3. Взаємодія з клієнтами. Система забезпечує комунікацію між адміністраторами та користувачами через сповіщення про новини, акції або зміни в розкладі. Вона також може містити функції для збору зворотного зв'язку, організації опитувань і надання підтримки через чат або інші канали.

4. Облік і звітність. Адміністратори отримують доступ до аналітичних даних, зокрема до інформації про завантаженість залів, фінансові показники, а також активність клієнтів. Це дозволяє краще оцінювати роботу залу та коригувати стратегії обслуговування.

5. Інтеграція платіжних систем. Обробка онлайн-платежів через різні платіжні системи дозволяє клієнтам зручно здійснювати оплати за послуги, тоді як система зберігає фінансові дані в захищеному вигляді.

4. Джерела розробки

4.1. Методичні вказівки до виконання магістерських кваліфікаційних робіт для студентів спеціальності 151 Автоматизація та комп'ютерно-

інтегровані технології / Уклад. В. М. Дубовой, Вінниця : ВНТУ, 2022. 38с.

4.2. Інформаційні технології та моделювання бізнес-процесів: навч. посіб. [Текст] / О. М. Томашевський, Г. Г. Цигелик, М. Б. Вітер, В. І. Дудук. К. : Центр учбової літератури, 2012. 296 с.

4.3. Клушин Ю. С. Підвищення швидкості роботи веб – додатків. Ю.С. Клушин, Ю.Б. Захарчин. Комп'ютерні системи і мережі. URL : <https://science.lpnu.ua/sites/default/files/journalpaper/2021/may/23469/csnv2n12020-37-47.pdf>

4.4. Мацанюк Б.М. Використання ІТ – технологій для оптимізації управління бізнес – процесами організацій та підприємств /Б.М. Мацанюк //Математичні методи, моделі та інформаційні технології в управлінні підприємством. Матеріали IV студентської вузівської наукової конференції. Вінниця: ВТЕІ КНТЕУ, 2019. С. 180 – 182

4.5. James E. Hunton. Blending Information and Communication Technology with Accounting Research. James E. Hunton. Accounting Horizons. Vol. 16. 2015.

4.6. Програмування на C#/ URL: http://lifeprog.ru/ukr/view_cat.php?cat=10

5. Вимоги до розробки.

5.1. Перелік головних функцій:

Управління членством. Реєстрація нових користувачів. Ведення профілю клієнта (редагування особистої інформації). Відстеження статусу членства (активне, прострочене). Обробка та автоматизація платежів за абонементи.

Бронювання залів. Перегляд доступності спортивних залів. Бронювання часу для індивідуальних або групових занять. Підтвердження, редагування та скасування бронювань.

Планування тренувань. Створення розкладу тренувань адміністраторами. Запис клієнтів на групові або індивідуальні тренування. Відстеження відвідуваності та результативності занять.

Комунікація з клієнтами. Надсилання повідомлень про новини, зміни в розкладі або акції. Організація зворотного зв'язку через опитування або чат. Сповіщення про майбутні події та нагадування про тренування.

Аналітика та звітність. Генерація звітів про завантаженість залів та тренувальні сесії. Фінансові звіти та аналіз доходів. Оцінка популярності тренувальних програм та послуг.

Керування адміністраторами та користувачами. Призначення ролей та прав доступу для адміністраторів. Моніторинг змін на сайті та персональна відповідальність за них. Ведення журналу дій адміністраторів та користувачів.

Інтеграція платіжних систем. Підтримка онлайн-платежів через різні платіжні платформи. Забезпечення безпеки фінансових транзакцій.

5.2. Основні технічні вимоги до розробки.

5.2.1. Вимоги до програмної платформи

- підтримка різних ОС (Windows, Linux, macOS)
- підтримка SQL Server, MySQL, PostgreSQL
- використання ASP.NET Core

5.2.2. Умови експлуатації системи

Кросплатформність. Система повинна бути сумісною з різними операційними системами, включаючи Windows, Linux та macOS, що дозволяє користувачам і адміністраторам працювати з нею на будь-яких пристроях.

Продуктивність. Система повинна забезпечувати високу швидкість обробки запитів та масштабованість для обробки великої кількості одночасних підключень та операцій (бронювань, реєстрацій, оплат).

Модульність та гнучкість. Використання сучасних фреймворків, таких як ASP.NET Core, які дозволяють підключати модулі та розширювати функціонал системи без повної зміни базової структури.

Інтеграція з базами даних. Система повинна підтримувати інтеграцію з різними типами баз даних, наприклад, SQL Server, MySQL, PostgreSQL, що забезпечує зберігання та обробку даних про користувачів, бронювання, розклади та фінансові операції.

Інтеграція платіжних систем. Платформа повинна мати можливість інтеграції з популярними платіжними шлюзами для прийому онлайн-платежів, забезпечуючи безпечне проведення фінансових операцій.

Безпека. Платформа повинна підтримувати шифрування даних (SSL) для захисту персональних і фінансових даних користувачів. Реалізація захисту від атак (наприклад, CSRF, XSS, SQL Injection) для забезпечення безпеки веб-додатку.

Адаптивний дизайн. Система повинна мати інтерфейс, адаптований для мобільних пристроїв та різних розмірів екранів, що полегшує доступ користувачам зі смартфонів та планшетів.

Легкість розгортання та підтримки. Система повинна мати зрозумілу інструкцію для розгортання та налаштування, а також підтримувати автоматизовані інструменти для оновлення та моніторингу.

Підтримка масштабованості. Платформа повинна дозволяти масштабування для роботи з більшою кількістю користувачів та адміністративних ролей при зростанні кількості клієнтів спортзалу.

Впровадження API. Для розширення можливостей та інтеграції з іншими системами платформа повинна підтримувати API для взаємодії з мобільними додатками, сторонніми сервісами або додатковими модулями.

6. Стадії та етапи розробки.

6.1 Пояснювальна записка:

1. Аналіз методів, принципів, підходів і засобів реалізації задачі автоматизації процесами в об'єкті управління відповідно до теми дипломної роботи. Постановка задач дослідження «20» жовтня 2024 р.

2. Удосконалення технології прийняття рішень при автоматизації об'єкту управління «25» жовтня 2024 р.

3. Визначення технічних характеристик системи «25» жовтня 2024 р.

4. Розробка апаратної частини системи «30» жовтня 2024 р.

5. Розробка програмної частини системи «10» листопада 2024 р.

6.2 Графічні матеріали:

1. Розробка моделі системи «15» листопада 2024 р.

2. Розробка діаграм системи «15» листопада 2024 р.

3. Тестування програмного забезпечення «25» листопада 2024 р.

7. Порядок контролю і приймання.

7.1. Хід виконання роботи контролюється керівником роботи. Рубіжний контроль провести до 01 грудня 2024 р.

7.2. Атестація проєкту здійснюється на попередньому захисті. Попередній захист магістерської кваліфікаційної роботи провести до 05 грудня 2024 р.

7.3. Підсумкове рішення щодо оцінки якості виконання роботи приймається на засіданні ЕК. Захист магістерської кваліфікаційної роботи провести до 09 грудня 2024 р.

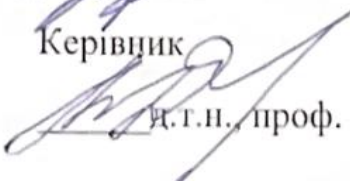
Додаток В – Ілюстративна частина
(обов'язковий)

ІЛЮСТРАТИВНА ЧАСТИНА
на виконання магістерської дипломної роботи
РОЗРОБКА ВЕБ ОРІЄНТОВНОЇ СИСТЕМИ ДЛЯ ОРГАНІЗАЦІЇ РОБОТИ
СПОРТИВНОГО ЗАЛУ

Студент групи

 2АКІТР-23м Віталій КАРВАЦЬКИЙ

Керівник

 д.т.н., проф. Володимир ДУБОВОЙ

Вінниця 2024

ВІННИЦЬКИЙ НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ
ФАКУЛЬТЕТ ІНТЕЛЕКТУАЛЬНИХ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ ТА АВТОМАТИЗАЦІЇ
КАФЕДРА КОМП'ЮТЕРНИХ СИСТЕМ УПРАВЛІННЯ

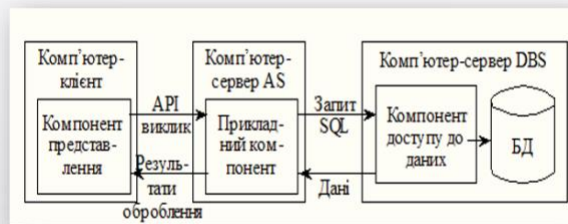
Розробка веб-орієнтовної системи для організації роботи спортивного залу

Виконав: студент 2-го курсу, групи 2AKITP-23м спеціальності
174 – Автоматизація, комп'ютерно-інтегровані технології та роботехніка
Віталій КАРВАЦЬКИЙ
Керівник: д.т.н., проф каф. КСУ
Володимир ДУБОВОЙ

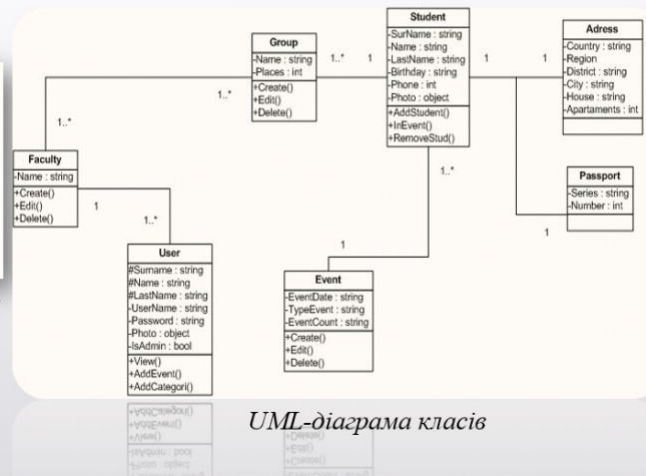
ПОСТАНОВКА ЗАДАЧІ

- **Актуальність теми.** Веб-орієнтовані системи є актуальними інструментами для автоматизації та оптимізації управління роботою спортивних закладів. Вони дозволяють ефективніше розпоряджатися ресурсами, зручніше організовувати графіки тренувань та підвищувати рівень обслуговування клієнтів. Розробка та впровадження такої системи здатні вирішити проблеми, які виникають під час взаємодії між клієнтами, тренерами та адміністрацією, сприяючи підвищенню продуктивності роботи спортивних залів.
- **Мета** - розробити веб-орієнтовану систему для організації роботи спортивного залу, яка дозволить користувачам ефективно планувати та управляти графіками тренувань, розкладом заходів та взаємодією між клієнтами, тренерами та адміністрацією. Для досягнення поставленої мети необхідно виконати наступні **завдання**:
 1. Дослідити існуючі методи організації роботи спортивних залів і виявити основні проблеми та потреби користувачів.
 2. Розробити функціонал системи, який включатиме можливість керування розкладом тренувань, змагань та бронюванням ресурсів.
 3. Забезпечити зручний інтерфейс для клієнтів, тренерів та адміністрації.
 4. Впровадити механізми автоматизації процесів планування та моніторингу ефективності використання часу.
 5. Забезпечити безпеку даних та конфіденційність користувачів.
- **Об'єкт дослідження** – веб-орієнтована система для організації роботи спортивного залу, яка спрямована на покращення ефективності планування тренувань та управління ресурсами.
- **Предмет дослідження** – процеси організації та управління тренувальним процесом в спортивних залах.
- **Практична цінність.** Розроблена веб-система дозволить спростити управління спортивним залом, автоматизувати процеси планування тренувань, розподілу ресурсів та взаємодії між тренерами та клієнтами. Це призведе до підвищення ефективності використання ресурсів, покращення якості обслуговування клієнтів та збільшення лояльності відвідувачів. Система також допоможе уникнути конфліктів у розкладі та оптимізувати роботу персоналу.
- **Інноваційність розробки.** Розробка веб-орієнтованої системи є інноваційною через інтеграцію різних функцій керування графіками тренувань, автоматизацію бронювання ресурсів та забезпечення зручної взаємодії між користувачами. Система дозволить значно зменшити ручні процеси, підвищити ефективність роботи спортивних закладів та створити кращі умови для їх відвідувачів.
- **Апробація.** Результати дипломної роботи були представлені на LIV Всеукраїнській науково-технічній конференції підрозділів Вінницького національного технічного університету у 2025 році на тему: «Розробка веб-орієнтовної системи для організації роботи спортивного залу», науковий керівник: д.т.н., професор, В.М. Дубовой.
- **Публікація.** Дубовой В.М., Карвацький В.Ю. Розробка веб-орієнтовної системи для організації роботи спортивного залу. *Матеріали LIV Всеукраїнської науково-технічної конференції підрозділів Вінницького національного технічного університету. Вінниця. ВНТУ, 2025*

РОЗРОБЛЕННЯ АРХІТЕКТУРИ ПРОГРАМНОЇ СИСТЕМИ



Модель тривірневої архітектури веб-орієнтованої системи



UML-діаграма класів

Архітектура веб-орієнтованої інформаційної системи та технології розв'язання поставлених задач



Діаграма активності процесу авторизації користувача

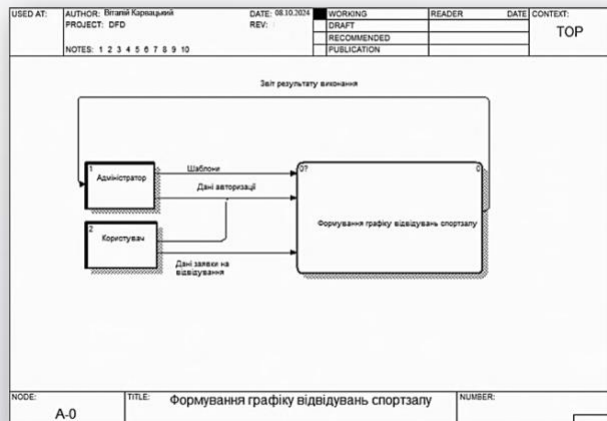


Діаграма активності процесу реєстрації користувача

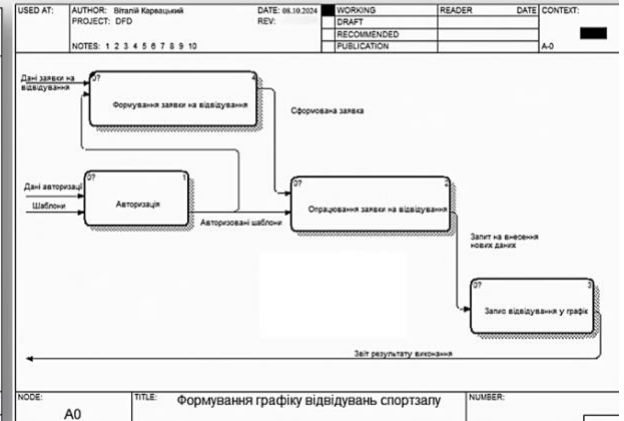


Діаграма активності процесу додавання відвідування

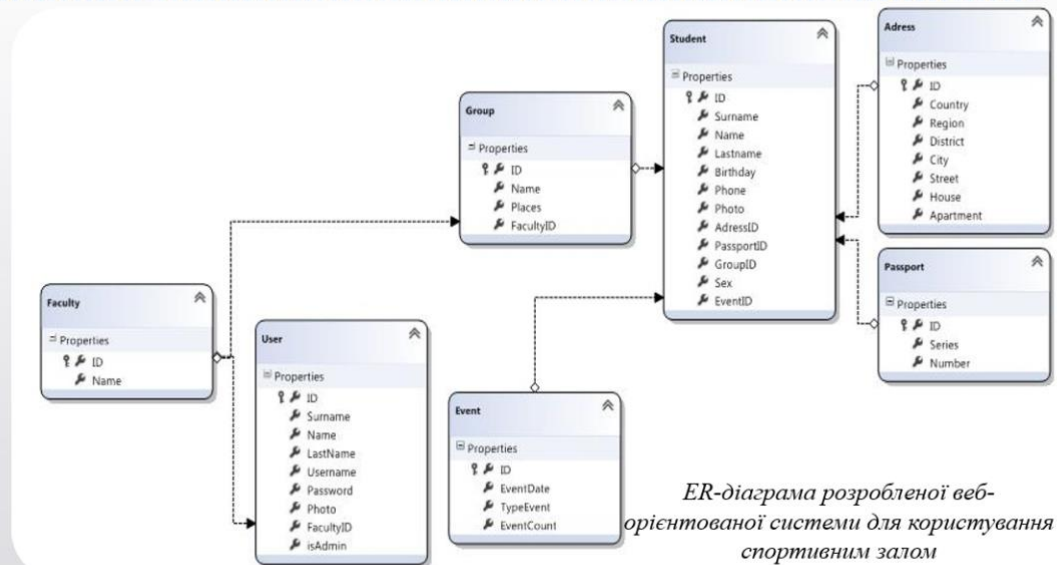
Проектування структури бази даних



Діаграма потоків даних



Діаграма декомпозиції



Програмна реалізація проекту

```
public bool AuthorizeAdmin(string username, string password)
{
    // Перевірка логіна та пароля
    if (username == "admin" && password == "password123")
    {
        return true; // Авторизація пройшла успішно
    }
    return false; // Авторизація не вдалася
}
```

Фрагмент коду «авторизація адміністратора на основі логіна та пароля»

```
private readonly MyDbContext _context;
public UserService(MyDbContext context)
{
    _context = context;
}

// Метод для отримання адміністратора за його ID
public Administrator GetAdministratorById(int adminId)
{
    return _context.Administrators
        .FirstOrDefault(admin => admin.Id == adminId);
}

// Метод для отримання користувача за його ID
public User GetUserById(int userId)
{
    return _context.Users
        .FirstOrDefault(user => user.Id == userId);
}

// Метод для внесення змін до даних
public void UpdateUserData(User updatedUser)
{
    var existingUser = _context.Users.Find(updatedUser.Id);
}
```

Фрагмент коду сервісу користувачів для взаємодії з базою даних

Організація інтерфейсу з користувачем

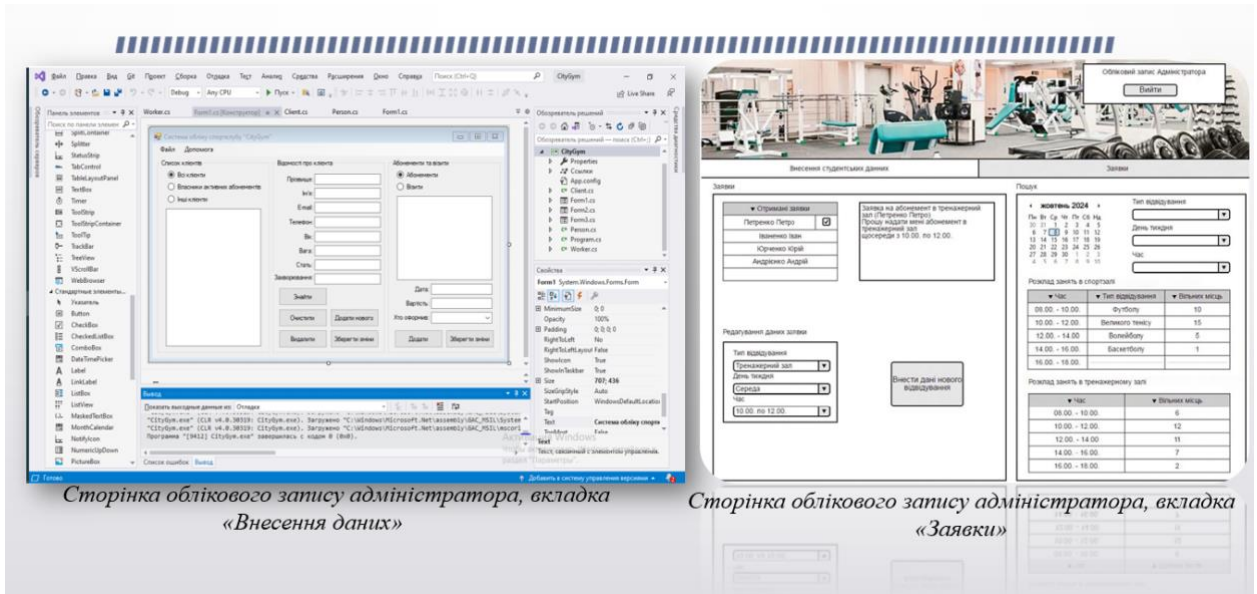


Головна сторінка сайту спортивного залу

Головна				Графік відвідувань			
Час	Пн	Вт	Ср	Чт	Пт	Сб	Нд
09:00	Program time	Stretching	Program time	Stretching	Plates		
10:00	Body sculpt	Ступені ніжки	Tabata	Upper body + прес	Ступені ніжки	Plates	Stretching
11:00	Stretching	Plates	Повільний швидкості АБТ	Plates	Body sculpt	Body sculpt	Functional training
12:00						Boxe	Body sculpt
13:10	Групова сила	(17:00) Прес+Ступені	Stretching	(17:00) Upper body	Калістетика		
14:10	Body sculpt	(18:00) СТОП Цикли	Кругові тренування	(18:00) Body sculpt	Прес+Ступені		
15:10	Кругові тренування	(19:00) Boxe	Ступені ніжки	(19:00) Boxe	Кругові тренування		
20:10	Step Dance	(20:30) Boxe	Functional+Step	(20:30) Boxe	Step Dance		
21:10	Stretching RELAX		Stretching RELAX		Stretching RELAX		
22:10	Відпочинок		Відпочинок		Відпочинок		

Сторінка перегляду розкладу занять

Сторінка аутентифікації



Тестування													
#	Description	Steps to Reproduce	Expected Result	Actual Result	Passed/Failed	Environment	Nr	Description	Steps to Reproduce	Expected Result	Actual Result	Passed/Failed	Environment
1	Перевірка створення нової бази даних	1. Натиснути праву кнопку на "Databases" 2. Обрати "New Database" 3. Ввести назву бази даних, натиснути "ОК"	База даних створена без помилок, відображається в списку	База даних успішно створена і з'явилася в списку	Passed	SQL Server Management Studio	1	Перевірка кнопки "Зареєструватися"	Натискання на кнопку "Зареєструватися" на головному екрані	Користувач перенаправлений на сторінку реєстрації	Користувач успішно перенаправлений на сторінку реєстрації	Passed	Windows 10, Chrome
2	Перевірка створення таблиці Faculty	1. Вибрати базу даних 2. Створити SQL-скрипт для таблиці Faculty і виконати	Таблиця Faculty створена без помилок, з усіма полями	Таблиця Faculty успішно створена з відповідними полями	Passed	SQL Server Management Studio	2	Перевірка поля "Email" в формі реєстрації	Введення неправильного формату email в поле "Email"	Виведення повідомлення про помилку	Повідомлення про помилку відображається коректно	Passed	Windows 10, Chrome
3	Перевірка зв'язку між таблицями	1. Створити таблицю User з полем Faculty_id 2. З'єднати Faculty_id з таблицею Faculty через зовнішній ключ	Зовнішній ключ успішно створений, зв'язок між таблицями працює	Зовнішній ключ успішно створений, зв'язок функціонує коректно	Passed	SQL Server Management Studio	3	Перевірка поля "Пароль"	Введення пароля менш ніж 6 символів	Виведення повідомлення про помилку	Повідомлення про помилку відображається коректно	Passed	Windows 10, Chrome
4	Перевірка вставки запису у таблицю	1. Ввести SQL-скрипт для додавання запису у таблицю Faculty 2. Виконати скрипт	Запис додається до таблиці Faculty без помилок	Запис успішно додано до таблиці Faculty	Passed	SQL Server Management Studio	4	Перевірка кнопки "Вхід"	Введення правильних даних і натискання кнопки "Вхід"	Користувач успішно авторизується в системі	Користувач успішно авторизується	Passed	Windows 10, Chrome
5	Перевірка виконання запитів SELECT	1. Виконати запит SELECT для таблиці Faculty	Запит повертає всі записи таблиці Faculty	Запит повертає коректні результати	Passed	SQL Server Management Studio	5	Перевірка кнопки "Вийти"	Натискання на кнопку "Вийти" після авторизації	Користувач успішно виводиться з системи	Користувач успішно виведений з системи	Passed	Windows 10, Chrome
6	Перевірка оновлення запису	1. Оновити запис у таблиці Faculty 2. Виконати SQL-скрипт для оновлення	Запис успішно оновлений	Запис успішно оновлений у таблиці	Passed	SQL Server Management Studio	6	Перевірка відображення форми пошуку	Перехід на сторінку "Таблиця даних"	Всі дані відображаються коректно	Всі дані відображаються коректно	Passed	Windows 10, Chrome
7	Перевірка видалення запису	1. Видалити запис з таблиці Faculty 2. Виконати SQL-скрипт для видалення	Запис успішно видалений з таблиці Faculty	Запис успішно видалений	Passed	SQL Server Management Studio	7	Перевірка кнопки "Оновити"	Натискання кнопки "Оновити" в таблиці даних	Дані на сторінці оновлюються без помилок	Дані оновлюються без помилок	Passed	Windows 10, Chrome
							8	Перевірка відображення форми пошуку	Введення тексту в поле пошуку	Пошукові результати відображаються коректно	Результати пошуку відображаються коректно	Passed	Windows 10, Chrome
							9	Перевірка реакції на неактивні кнопки	Натискання на неактивні кнопки	Вони не повинні реагувати на натискання	Кнопки не реагують на натискання	Passed	Windows 10, Chrome
							10	Перевірка адаптивності інтерфейсу	Зміна розміру вікна браузера	Інтерфейс повинен адаптуватися до різних розмірів екрана	Інтерфейс коректно адаптується	Passed	Windows 10, Chrome



ДЯКУЮ ЗА УВАГУ