

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»

Кафедра інженерії програмного забезпечення

КУРСОВА РОБОТА
(ПОЯСНЮВАЛЬНА ЗАПИСКА)

з дисципліни «Бази даних»
на тему:
«Бази даних музичного порталу»

студентки II курсу групи ІПЗ-21-2
спеціальності 121 «Інженерія
програмного забезпечення»
Ватиль Вікторії Валеріївни
(прізвище, ім'я та по-батькові)

Керівник к.тн, доц. І. І. Сугоняк

Дата захисту: " ____ " _____ 2022р.
Національна шкала _____
Кількість балів: _____
Оцінка: ECTS _____

Члени комісії

_____	<u>О.В. Коротун</u>
(підпис)	(прізвище та ініціали)
_____	<u>С.М. Кравченко</u>
(підпис)	(прізвище та ініціали)
_____	<u>О.В. Чижмотря</u>
(підпис)	(прізвище та ініціали)

Житомир – 2023

ДЕРЖАВНИЙ УНІВЕРСИТЕТ «ЖИТОМИРСЬКА ПОЛІТЕХНІКА»
Факультет інформаційно-комп'ютерних технологій
Кафедра інженерії програмного забезпечення
Освітній рівень: бакалавр
Спеціальність 121 «Інженерія програмного забезпечення»

«ЗАТВЕРДЖУЮ»
Зав. кафедри
_____ Т.М.Вакалюк
“ ____ ” _____ 2023 р.

ЗАВДАННЯ
НА КУРСОВУ РОБОТУ СТУДЕНТЦІ
Ватиль Вікторії Валеріївни

1. Тема роботи: бази даних електронної бібліотеки,
керівник курсового проекту: І.І. Сугоняк
2. Строк подання студентом: “18” січня 2022 р.
3. Вихідні дані до роботи: розробити базу даних електронної бібліотеки.
4. Зміст розрахунково-пояснювальної записки (перелік питань, які підлягають розробці)
 1. Постановка завдання
 2. Аналіз аналогічних розробок
 3. Алгоритми роботи програми
 4. Опис роботи програми
 5. Програмне дослідження
5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень)
 1. Презентація
 2. Посилання на репозиторій:
6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посади консультанта	Підпис, дата	
		завдання видав	завдання прийняв
1,2,3	Сугоняк І.І., к.т.н., доцент, доцент кафедри КН		

7. Дата видачі завдання “15” березня 2023 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів курсового проекту	Строк виконання етапів проекту	Примітки
1	Постановка завдання	25.03.2023	Виконано
2	Пошук, огляд та аналіз аналогічних розробок	25.03.2023	Виконано
3	Формулювання технічного завдання	28.03.2023	Виконано
4	Опрацювання літературних джерел	01.04.2023	Виконано
5	Проектування структури	07.04.2023	Виконано
6	Написання програмного коду	08.04.2023	Виконано
7	Відлагодження	16.04.2023	Виконано
8	Написання пояснювальної записки	17.04.2023	Виконано
9	Захист	18.05.2023	Виконано

Студент

(підпис)

Вікторія ВАТИЛЬ

(прізвище та ініціали)

Керівник роботи

(підпис)

Інна СУГОНЯК

(прізвище та ініціали)

РЕФЕРАТ

Курсова робота присвячена розробці бази даних музичного порталу.

Пояснювальна записка до курсової роботи на тему розробки бази даних музичного порталу складається з: вступу, основної частини (трьох розділів), висновків, списку використаної літератури.

Текстова частина викладена на 50 сторінках друкованого тексту.

Список використаних джерел містить 23 найменувань і займає 2 сторінки. В роботі наведено 22 рисунка. Загальний обсяг роботи – 53 сторінки.

Ключові слова: БД, SQL, зв'язок, дані, веб-додаток, авторизація, реляційні бази даних, інфологічна модель, даталогічна модель, sql-запити.

					ДУ «Житомирська політехніка».23.121.02.000 – ПЗ							
Змн.	Арк.	№ докум.	Підпис	Дата	Розробка бази даних музичного порталу			Літ.	Арк.	Аркушів		
Розроб.		Ватиль В.В.								4	55	
Перевір.		Сугоняк І.І.						ФІКТ Гр. ІПЗ-21-2[1]				
Керівник												
Н. контр.												
Зав. каф.												

ЗМІСТ

ЗМІСТ	5
ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ	7
ВСТУП	8
РОЗДІЛ 1. АНАЛІЗ ОСОБЛИВОСТЕЙ РОЗРОБКИ БАЗИ ДАНИХ МУЗИЧНОГО ПОРТАЛУ	
1.1 Аналіз інформаційних потреб та визначення предметної області дослідження	9
1.2 Обґрунтування вибору засобів реалізації	12
1.3 Технічне завдання на курсову роботу	18
Висновок до першого розділу.	24
РОЗДІЛ 2. ПРОЕКТУВАННЯ БАЗИ ДАНИХ ЗА НАПРЯМКОМ КУРСОВОЇ РОБОТИ	
2.1 Проектування загального алгоритму роботи програми	25
2.2 Розробка функціональних алгоритмів роботи програми	29
2.3 Розробка програмного забезпечення	33
Висновки до другого розділу	34
РОЗДІЛ 3. ОПИС РЕАЛІЗАЦІЯ ПІДСИСТЕМИ ОБРОБКИ ДАНИХ ЗА НАПРЯМКОМ «БАЗА ДАНИХ МУЗИЧНОГО ПОРТАЛУ»	
3.1 Проектування інтерфейсу обробки даних	35
3.2. Реалізація операцій обробки даних в БД за напрямком «база даних для музичного порталу»	36
3.3. Організація системи та тестування	40
Висновки до третього розділу	47
РОЗДІЛ 4. АДМІНІСТРУВАННЯ БАЗ ДАНИХ	
4.1. Розробка заходів захисту інформації в БД	48
4.2. Налаштування параметрів роботи з MySQL	49
Висновки до четвертого розділу	51

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				5
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ.....	52
СПИСОК ЛІТЕРАТУРИ.....	53

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				
Змн.	Арк.	№ докум.	Підпис	Дата		6

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

БД – База даних

СУБД – Система керування базами даних

КР – Курсова робота

ПЗ – Програмне забезпечення

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				
Змн.	Арк.	№ докум.	Підпис	Дата		7

ВСТУП

Актуальність теми дослідження. Музична індустрія є важливою складовою сфери культури та розваг. У сучасному світі музика доступна практично кожному завдяки численним музичним порталам, що дозволяють слухачам з різних куточків світу отримувати доступ до величезного масиву музичного контенту. Актуальність теми полягає в тому, що розвиток технологій та постійне зростання попиту на онлайн-музику створюють необхідність у розробці ефективних та масштабованих баз даних, які можуть забезпечити швидкий доступ, надійне зберігання та зручну організацію музичних творів. Розробка бази даних для музичного portalу є актуальним завданням, оскільки вона дозволяє забезпечити ефективне управління музичним контентом, надаючи користувачам можливість легко знаходити та прослуховувати улюблені композиції.

Метою дослідження є аналіз предметної області музичних порталів та розробка бази даних, яка буде основою для функціонування такого portalу.

Завданням є аналіз теоретичних засад проектування та реалізації баз даних, аналіз існуючих музичних порталів та їх структур баз даних, розробка логічної та фізичної моделі бази даних для музичного portalу, проектування та реалізація бази даних на основі обраної предметної області, розробка SQL-запитів для роботи з базою даних, створення програмного інтерфейсу для взаємодії з базою даних.

Об'єктом дослідження даної курсової роботи є процес розробки бази даних для музичного portalу.

Предметом дослідження даної курсової роботи є сукупність методів та інструментів, необхідних для розробки бази даних, яка забезпечить ефективне управління музичним контентом та інтеграцію з веб-інтерфейсом музичного portalу.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				
Змн.	Арк.	№ докум.	Підпис	Дата		8

РОЗДІЛ 1. АНАЛІЗ ОСОБЛИВОСТЕЙ РОЗРОБКИ БАЗИ ДАНИХ МУЗИЧНОГО ПОРТАЛУ

1.1 Аналіз інформаційних потреб та визначення предметної області дослідження

Сучасні музичні портали зберігають величезну кількість інформації: дані про музичних виконавців, новини, користувачів, музику та інші важливі дані. Щось можна зберегти в Google Sheets, файлах Excel або навіть у друкованому вигляді в папках.

Однак якщо йдеться про велику компанію з мільйонами операцій на день, то в онлайн-документі інформація вже не вміститься. Для цього потрібні бази даних. У них можна зберігати, змінювати та швидко шукати інформацію.

Сучасне цифрове суспільство керується даними. Кожен клік, пошуковий запит і покупка - це інформація, яка зберігається і обробляється. Бази даних відповідають за зберігання цієї величезної кількості інформації. Без баз даних веб-сайти, мобільні додатки та комп'ютерні програми не могли б функціонувати так, як ми звикли.

База даних — це структурована колекція даних, яка зберігається в електронному форматі та доступна для операцій обробки, пошуку та аналізу. За своєю суттю, база даних – це цифрове сховище, що містить інформацію про різні об'єкти, які можуть бути пов'язані один з одним (Рис. 1.1.1). До таких об'єктів належать люди, місця, події, транзакції, продукти та багато інших елементів реального і віртуального світу. Бази даних забезпечують швидкий доступ до інформації, безпеку, цілісність даних і дозволяють ефективно управляти інформацією.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				
Змн.	Арк.	№ докум.	Підпис	Дата		9

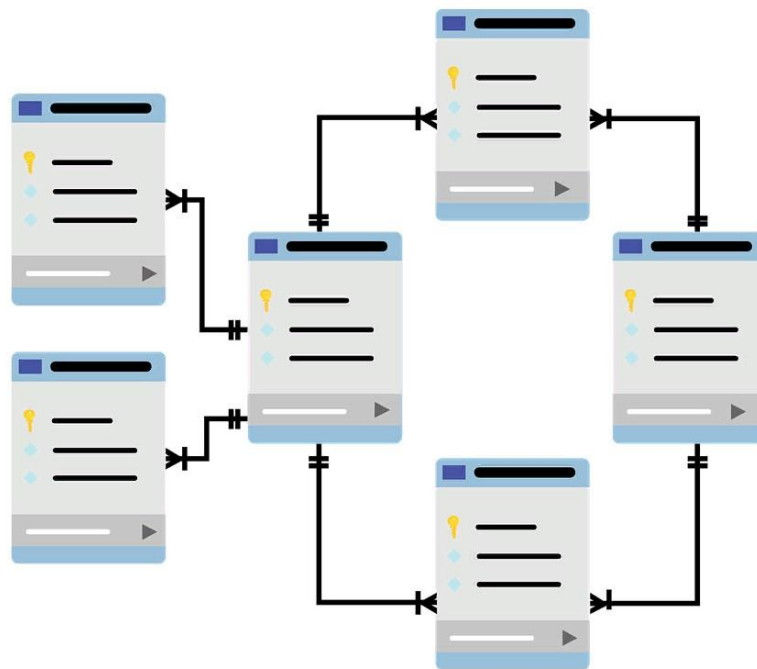


Рис. 1.1.1 Зв'язки в базі даних

Бази даних використовуються в різних сферах життя, де потрібно зберігати, організовувати та обробляти великі обсяги інформації. Найпоширенішими сферами застосування баз даних є:

- Бізнес і фінанси. Компанії використовують бази даних для зберігання даних про клієнтів, продукти, транзакції, фінансові операції та інше. Це допомагає їм вести облік, аналізувати тренди та приймати стратегічні рішення
- Освіта. У навчальних закладах бази даних використовуються для зберігання інформації про студентів, викладачів, навчальні програми, оцінки тощо
- Медицина. За допомогою БД, медичні працівники зберігають медичну інформацію про пацієнтів, історії хвороб, відомості про лікарські препарати та інші медичні дані.
- Наука і дослідження. У наукових дослідженнях бази даних використовуються для зберігання даних про експерименти, вимірювання, результати досліджень та іншої інформації, яка допомагає вченим робити висновки та розвивати науку.

- Інтернет-сервіси. Вебсайти та онлайн-платформи не просто використовують бази даних, а ґрунтуються на них для зберігання користувацьких профілів, контенту, коментарів, транзакцій та іншої інформації.

Існує декілька типів БД, що використовуються сьогодні, кожен з яких має власні переваги (Рис. 1.1.2):

- Реляційні бази даних

Цей тип вважається одним з найбільш відомих серед баз даних. Вони організовані у вигляді таблиць, де кожен рядок відповідає запису, а кожний стовпчик – полю. Реляційні бази даних використовують мову структурованого запиту (SQL) для операцій з даними.

- Документо-орієнтовані бази даних

Вони призначені для зберігання, отримання та оновлення документів у форматі JSON, XML тощо. Ці бази даних набули популярності в розробці веб-застосунків та розподілених системах.

- Бази даних Ключ-значення

Такі БД працюють на принципі, який використовує просту модель даних “ключ-значення”. У цій моделі кожен запис в базі даних містить ключ, що ідентифікує даний запис, і значення, яке зв’язане з цим ключем.

- Колонкові бази даних

Вони зберігають дані у вигляді стовпців, а не рядків, що робить їх ефективними для операцій з великими об’ємами даних та аналітичних запитів.

- Графові бази даних

Графові БД зберігають дані у вигляді графів, де вузлами є об’єкти, а ребрами — зв’язки між ними. Ці бази даних ідеально підходять для моделювання складних взаємозв’язків, таких як соціальні мережі або мережі транспорту.

- Самокеровані бази даних

Це нове покоління баз даних, які використовують штучний інтелект та машинне навчання для автоматизації адміністрування, оптимізації продуктивності та підвищення надійності. Вони можуть автоматично налаштовувати

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				11
Змн.	Арк.	№ докум.	Підпис	Дата		

себе, виявляти та усувати проблеми, реагувати на зміни в навантаженні та багато іншого.

- Хмарні бази даних

Ці бази даних розгортаються та працюють у хмарному середовищі, що дозволяє користувачам зберігати та обробляти свої дані віддалено, без необхідності наявності власної інфраструктури. Вони забезпечують високу доступність, масштабованість та безпеку, а також надають різноманітні сервіси для аналізу та обробки даних.

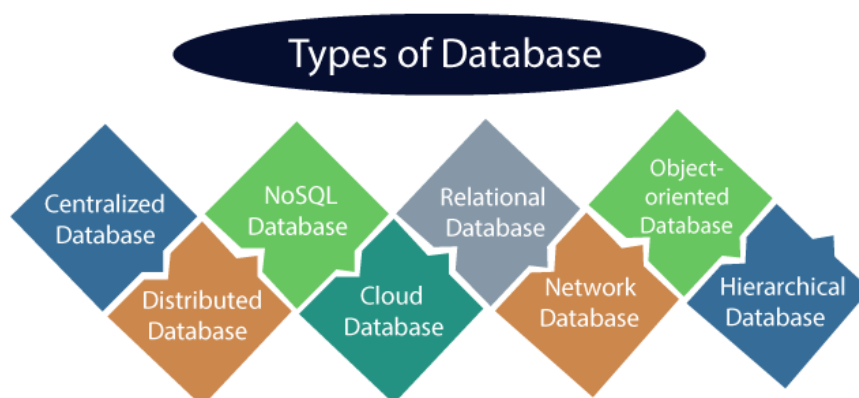


Рис. 1.1.2 Типи БД

Отже, предметною областю дослідження є розробка бази даних музичного порталу, який надасть користувачам можливість переглядати, шукати та керувати музичним контентом, таким як треки, альбоми, новини, а також інформацію про виконавців.

1.2 Обґрунтування вибору засобів реалізації

Є кілька можливостей реалізації БД для музичного порталу. Серед них ми маємо обрати СУБД для зручного формування бази даних.

Система управління базами даних (СУБД) — це спеціалізоване програмне забезпечення, яке дозволяє створювати, управляти та маніпулювати базами даних. Вона виступає в ролі посередника між користувачами або програмами та фізичним зберіганням даних. Популярні приклади СУБД це MySQL, PostgreSQL, Oracle Database, Microsoft SQL Server та інші.

Для реалізації музичного порталу було обрано розглянути три СУБД: MySQL, PostgreSQL та MongoDB. Далі проведемо їх порівняльний аналіз.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				12
Змн.	Арк.	№ докум.	Підпис	Дата		

Таблиця 1.2.1

Функціональне порівняння СУБД			
Характеристика	MySQL	PostgreSQL	MongoDB
Адміністративне керування	Добре	Відмінно	Добре
Графічні інструменти	Відмінно	Добре	Добре
Простота обслуговування	Відмінно	Відмінно	Добре
Механізм даних	Добре	Відмінно	Добре
Робота з декількома ЦП	Добре	Відмінно	Добре
Функції з'єднання і вибір індексів	Відмінно	Відмінно	Добре
Одночасний доступ декількох користувачів	Добре	Відмінно	Відмінно
Обробка даних мультимедіа	Погано	Добре	Відмінно
Підключення до Web	Добре	Відмінно	Відмінно
Повнотекстовий пошук	Добре	Відмінно	Добре
Функціональна сумісність	Добре	Відмінно	Добре
Інтеграція з іншими СУБД	Добре	Відмінно	Добре
Єдина реєстрація	Добре	Добре	Добре
Робота під керівництвом ОС	Задовільно	Добре	Відмінно
Можливості програмування	Добре	Відмінно	Добре
Процедури, що зберігаються, та тригери	Добре	Відмінно	Добре
Вбудована мова програмування	Добре	Відмінно	Добре
Побудова БД	Добре	Відмінно	Добре
Мова SQL	Відмінно	Відмінно	-
Підтримка об'єктно-орієнтованої парадигми	Добре	Відмінно	Відмінно
Робота в режимі віддаленого доступу	Відмінно	Відмінно	Відмінно
Тиражування	Відмінно	Відмінно	Відмінно
Розподілена обробка транзакцій	Відмінно	Відмінно	Відмінно
Дистанційне адміністрування	Добре	Відмінно	Добре
Організація сховищ даних і підготовка звітів	Відмінно	Відмінно	Добре
Засоби завантаження	Відмінно	Відмінно	Добре
Засоби аналізу	Відмінно	Відмінно	Добре

Таблиця 1.2.2

Порівняння за вимогами технічного завдання			
Необхідні вимоги	MySQL	PostgreSQL	MongoDB
Локалізація інтерфейсу користувача	+	+	+
Підтримка структури відносних даних	+	+	-
Підтримка технології клієнт/сервер	+	+	+
Підтримка багатопроцесорної архітектури	+	+	+
Підтримка кластерної архітектури	+	+	+
Засоби для створення індексів	+	+	+
Відновлення баз даних із використанням журналу транзакцій	+	+	+
Механізм блокування транзакцій	+	+	+
Підтримка ANSI SQL	+	+	-
Підтримка SQL-3	+	+	-
Підтримка ODBC	+	+	-
Контроль цілісності БД	+	+	+
Утиліти резервування БД	+	+	+
Імпорт/експорт таблиць БД	+	+	+
Сумісність з ОС	+	+	+
Підтримка мережевих протоколів	+	+	+
Графічний інтерфейс для адміністраторів БД	+	+	+
Контроль доступу до даних	+	+	+
Централізоване керування користувачами	+	+	+
Оптимізатор запитів	+	+	-
Підтримка BLOB	+	+	+
Підтримка OLAP технологій	+	+	-
Підтримка протоколів VI SAN	+	+	-
Механізм реплікації даних	+	+	+
Підтримка служби єдиного каталогу	+	+	-
Контроль цілісності БД	+	+	+
Утиліти резервування БД	+	+	+

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				14
Змн.	Арк.	№ докум.	Підпис	Дата		

Таблиця 1.2.3

Експертна оцінка багатокористувацьких СУБД					
СУБД	Продуктивність	Конкурент- ний доступ	Кількість кори- стувачів	Великі БД	Готовність
MySQL	6	7	5	5	7
PostgreSQL	7	8	6	7	8
MongoDB	8	9	8	8	9

На основі сформованих таблиць (Табл. 1.2.1 – 1.2.3) можемо назвати певні переваги та недоліки кожної з СУБД (Рис. 1.2.1).



Рис. 1.2.1 Варіанти СУБД

MySQL

В даній СУБД є певні переваги над іншими. Така система управління як MySQL показує високу продуктивність, особливо в умовах обробки великої кількості одночасних запитів. Це робить її ефективною для застосування у високонавантажених веб-додатках, таких як музичні портали.

Також MySQL відома своєю простотою в установці, налаштуванні та обслуговуванні. Це значно зменшує витрати на підтримку бази даних, особливо в умовах обмежених ресурсів.

MySQL легко інтегрується з іншими СУБД та технологіями, що робить її універсальним вибором для різних проектів.

Але, в порівнянні з іншими системами управління, є декілька недоліків: MySQL менш ефективна в обробці складних транзакцій порівняно з іншими реляційними СУБД, такими як PostgreSQL.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				15
Змн.	Арк.	№ докум.	Підпис	Дата		

Також MySQL має обмежені можливості для обробки великих двійкових об'єктів (BLOB), що може бути недоліком для музичного порталу з великим обсягом мультимедійного контенту

PostgreSQL

Система управління PostgreSQL пропонує розширений набір можливостей, таких як підтримка складних транзакцій, тригерів, збережених процедур і об'єктно-реляційної моделі, що робить її більш гнучкою для складних завдань.

PostgreSQL також добре підходить для роботи зі складними структурами даних, такими як JSON, XML та інші. Це корисно для додатків, які потребують зберігання різнорідних даних.

Але є також певна кількість недоліків: встановлення та налаштування PostgreSQL може бути складнішим і вимагати більше часу та ресурсів, ніж у випадку з MySQL, що може ускладнити розгортання і підтримку для невеликих команд.

Разом з цим, хоча PostgreSQL дуже потужна, вона може бути менш ефективною у виконанні простих запитів у порівнянні з MySQL, що може бути важливим для музичного порталу з великою кількістю простих операцій.

MongoDB

Система управління MongoDB є документоорієнтованою і добре підходить для зберігання неструктурованих даних, таких як мультимедійні файли та метадані. Це корисно для музичних порталів, де дані можуть мати різні формати.

Також така СУБД як MongoDB забезпечує високу швидкість читання та запису даних, що особливо важливо для додатків, де швидкість доступу до даних є критичною.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				16
Змн.	Арк.	№ докум.	Підпис	Дата		

І все ще є певні мінуси – відсутність стандартної мови запитів SQL. MongoDB використовує власний синтаксис для запитів, що може вимагати додаткового навчання і складнішого переходу для команд, які звикли працювати з SQL.

Отже, MySQL, PostgreSQL і MongoDB — це потужні та гнучкі системи управління базами даних, кожна з яких має свої унікальні особливості та переваги. Вибір між ними часто залежить від специфіки проєкту. MySQL та PostgreSQL використовуються в технологіях, що потребують структурованого зберігання даних, а MongoDB краще підходить для проєктів, яким потрібна гнучкість структури даних, великі об’єми даних або даних, що часто змінюються.

MySQL — чудовий вибір для веб-додатків, які потребують високої швидкості й надійності. PostgreSQL краще підходить для великих проєктів, які потребують складних запитів та гнучкості в обробці даних, а MongoDB відмінно підходить для проєктів, яким потрібна висока швидкість роботи з великим обсягом даних і високої масштабованості.

На основі проведеного аналізу було визначено, що MySQL є найкращим вибором для розробки бази даних музичного порталу, адже володіє більшістю необхідних критеріїв для розробки БД музичного порталу (Рис.1.2.2).

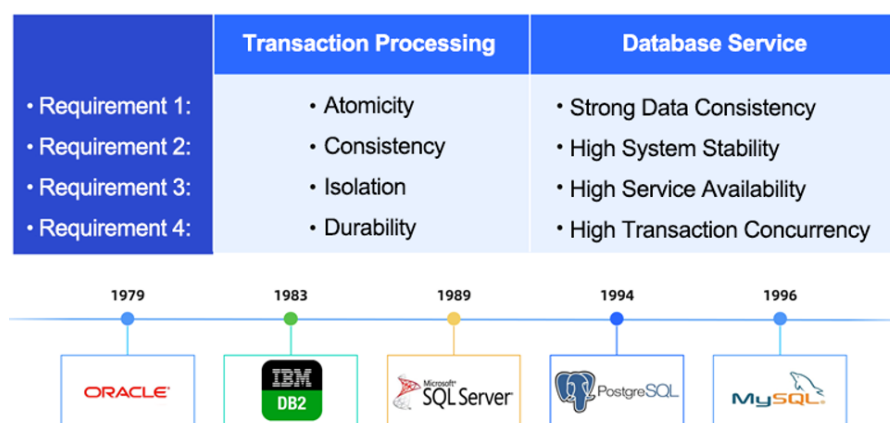


Рис. 1.2.2 Вимоги до СУБД

Незважаючи на деякі обмеження, MySQL пропонує оптимальне поєднання продуктивності, простоти адміністрування, масштабованості та сумісності, що робить її ідеальним вибором для проекту, який вимагає надійного та ефективного управління великими обсягами даних і одночасних запитів.

PostgreSQL хоч і пропонує більш розширений функціонал та високу надійність, але вимагає більш складного адміністрування і налаштування, що може бути надмірним для даного проекту.

Таким чином, MySQL забезпечить стабільну основу для розвитку музичного порталу, поєднуючи простоту використання з необхідною функціональністю та ефективністю.

1.3 Технічне завдання на курсову роботу

1. Загальне положення

1.1. Найменування програмного засобу

Повне найменування програмної системи: "База даних музичного порталу" (надалі "програма"). Коротка назва програмної системи - "Новини"

1.2. Призначення розробки та область застосування

Програмна система "База даних музичного порталу" призначена для зберігання, обробки та відображення даних про музичні новини та виконавців.

Система може бути впроваджена в роботу музичного порталу з кількістю співробітників до 25 осіб та базою вакансій та кандидатів до 100000 записів.

1.3. Мета

Програмна система "База даних музичного порталу" дозволить підвищити ефективність, повноту та швидкість пошуку та відбору новин музичного світу, та забезпечить користувачів своєчасною та повною інформацією про виконавців у повному обсязі.

1.4. Найменування розробника та замовника.

Розробник даного продукту – студентка групи ІПЗ-21-2 Ватиль Вікторія Валеріївна (надалі "розробник").

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				18
Змн.	Арк.	№ докум.	Підпис	Дата		

Замовник програмного продукту - кафедра програмного забезпечення обчислюваної техніки Житомирського державного технологічного університету в межах виконання курсової з дисципліни «Бази даних» (надалі замовник).

2. Підстава для розробки

2.1. Документ на підставі якого ведеться розробка

Робота ведеться на підставі навчального плану за напрямом 6.050201 «Програмна інженерія» наказу про закріплення тем курсових робіт за студентами.

3. Вимоги до програми

3.1. Вимоги до функціональних характеристик.

3.1.1. Загальні вимоги веб-додаток має забезпечувати:

Програмна система має забезпечувати:

- можливість дистанційної роботи з робочих станцій локальної та глобальної мережі підприємства;
- інтерфейс користувача, що не залежить від операційної системи;
- постійний доступ користувачів до БД;
- оптимальне збереження даних (за обсягом та структурою);
- аутентифікацію користувачів та захист інформації від несанкціонованого доступу;
- надійне збереження даних та можливість відновлення даних у випадку непередбачуваних збоїв системи;
- створення архівів даних, що не використовуються протягом визначеного терміну;
- можливість модернізації системи через зміну функціональних потреб користувача або модернізацію обладнання;

3.1.2. Склад виконуваних функцій

Розробити базу даних музичного порталу, що підтримує виконання наступних транзакцій:

1. Реєстрацію користувачів;

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				19
Змн.	Арк.	№ докум.	Підпис	Дата		

5. Пошук та відбір новин по мультикритеріями (жанр, виконавець, рік тощо).

6. Відбір та впорядкування виконавців за мультикритеріями (рік, жанр тощо).

7. Підтримку розкладу виходів альбому;

8. Формування звітності щодо виконавців та новин.

3.1.3. Організація вхідних і вихідних даних

Вхідними даними є інформація про виконавців (Назва, опис, жанр, картинка)

Організація вхідних і вихідних даних повинна відповідати інформаційній структурі виконуваних з операцій, вхідним та вихідним паперовим документами.

Введення оперативних даних повинно виконуватися з використанням діалогових екранних форм, побудованих на основі візуальних компонентів. Введення даних виконується на основі затверджених форм документів: анкета, заява, інформаційна довідка та в режимі online оператором зі слів користувача.

3.1.4. Часові характеристики і розмір пам'яті, необхідної для роботи програми.

Час реакції програми на дії користувача (маніпуляції з пристроями введення даних) не повинен перевищувати 0,25 с.

Час виконання команд меню не більше 1 с. Відображення масивів даних за запитом не більше 3 хвилин.

Доступність БД – 90% цілодобово.

Операції з'єднання з БД не більше 1 хвилини.

Обсяг оперативної пам'яті, необхідний для роботи програми не менше 1Гб.

Дисковий простір, необхідний для збереження програми і файлів даних не більше 300 Мб для робочої станції та 20 ГБайт.

Інсталяційний пакет програми, що містить у складі БД не повинні перевищувати 100 Мбайт.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				20
Змн.	Арк.	№ докум.	Підпис	Дата		

3.2. Вимоги до методів рішення і мов програмування

3.2.1. Вимоги до надійного функціонування

Програма повинна нормально функціонувати при безперебійній роботі ПК. Доступність БД 90% при одночасному доступі 30 користувачів.

При апаратних збоях, відновлення нормальної роботи програми повинне виконуватися після:

а) апаратні збої сервера - перезавантаження ОС сервера, запуск сервера БД (запуску резервного сервера, використання технологій RAID для збереження даних);

б) апаратні збої робочої станції – перезавантаження ОС ПК, запуск виконуваного файлу програми.

в) БД повинна повертатись в найближчий несуперечний стан – передбачити точки відновлення.

При збоях програмного забезпечення:

а) система повинна забезпечувати можливість відновлення даних та фіксацію і «відкат» транзакцій.

б) в системі має бути реалізована коректна обробка виняткових ситуацій.

3.2.2. Контроль вхідної і вихідної інформації

Для контролю коректності вхідної інформації та захисту від помилок оператора:

- Перевірка відповідності даних доменам інформаційних атрибутів;
- Використанням механізму авто заповнення та вибору за переліком для зв'язаних даних;
- Захист від помилок оператора (залипань, випадкових символів тощо).

Визначені некоректні дії повинні супроводжуватись повідомленнями про помилку і блокуванням операцій оновлення даних. В системі має бути передбачений захист від загального блокування.

Для вихідної інформації передбачити:

- відсутність блокування даних через багато користувальницький доступ;

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				21
Змн.	Арк.	№ докум.	Підпис	Дата		

- постійне оновлення даних у відображених на екрані звітних формах.
- повідомлення про використання режиму монопольного доступу супервайзерами БД.

3.2.3. Час відновлення після відмови.

Час відновлення після відмови, не пов'язаною з роботою програми, повинен складатися із: часу перезавантаження операційної системи; часу запуску сервера БД (підключення до сервера) запуску виконуваного файлу, часу повторного введення або зчитування даних.

3.3. Умови експлуатації і збереження

Програма використовується у багато користувальницькому середовищі. Регламенті операції проводити за графіком:

- створення резервних копій даних: 1 раз на тиждень;
- збереження резервної копії журналу транзакцій – 1 раз на тиждень;
- архівування даних – 1 раз на місяць;
- обслуговуючі функції (прибирання “сміття”, де фрагментація тощо) - щоденне;
- оновлення системи ідентифікації та аутентифікації користувачів – 1 раз на місяць та за потребою.

Збереження архівних копій - на визначених носіях. Збереження резервних копій – на визначеному диску без перезапису. Знищення копій за регламентом.

3.4. Вимоги до інформаційної і програмної сумісності.

3.4.1. Вимоги до інформаційних структур на вході і виході

Формат відображення даних має дозволяти імпорт даних в додатки MS Office для редагування та перегляду.

3.4.2. Вимоги до методів рішення і мов програмування

Вибір методів рішення здійснюється розробникам без узгодження з замовником. СУБД обирається у відповідності до характеристик визначених в п.3. З замовником погоджується вибір варіанту за вартісною ознакою.

3.4.3. Вимоги до системи програмних засобів.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				22
Змн.	Арк.	№ докум.	Підпис	Дата		

Вимоги до програмного забезпечення сервера:

ОС – серверна версія ОС Windows не нижче Windows 2000 Server, ІІS (або інший сервер), СУБД визначається встановлюється та налаштовується розробником (ліцензування СУБД виконується замовником).

Вимоги до програмного забезпечення робочої станції:

ОС - родина Windows не нижче Windows XP.

Драйвери периферійних пристроїв - введення/виводу визначаються та встановлюються при встановленні ОС в залежності від конфігурації робочої станції.

Пакет додатків MS Office.

Робоча станція є клієнтом мереж Microsoft.

3.5. Вимоги до складу і параметрів технічних засобів.

Вимоги до складу технічних засобів:

1. Сервер:

- сервер у базовій конфігурації із підтримкою RAID;
- з'ємний запам'ятовуючий пристрій для архівування даних;
- принтер для друку;
- засоби для під'єднання до локальної мережі.

2. Робоча станція:

- ПК на базі процесорів Intel, AMD у стандартній комплектації;
- периферійні пристрої друку;
- засоби для під'єднання до локальної мережі. засоби для під'єднання до локальної мережі.
- Pentium III з тактовою частотою 750 МГц;
- оперативна пам'ять - 128Мбайт;
- об'єм дискової пам'яті - 20Гбайт

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				23
Змн.	Арк.	№ докум.	Підпис	Дата		

Висновок до першого розділу.

У цьому розділі було проведено детальний аналіз трьох популярних систем управління базами даних (СУБД) - MySQL, PostgreSQL та MongoDB. Були розглянуті переваги та недоліки кожної з них, а також оцінені можливості кожної СУБД відповідно до вимог, поставлених до бази даних музичного порталу.

MySQL була обрана як остаточне рішення для розробки бази даних музичного порталу завдяки своїй високій продуктивності, простоті адміністрування та масштабованості. MySQL ефективно обробляє великі обсяги даних та одночасні запити, що є важливим для високонавантажених веб-додатків, таких як музичний портал.

PostgreSQL, пропонуючи більш потужні функції і високу надійність, вимагає більш складного налаштування і адміністрування і може не підходити для цього проекту; MongoDB має репутацію гнучкої обробки неструктурованих даних і швидкості обробки, але має транзакційні можливості і стандартну Обмежена підтримка мови SQL робить її непридатною для використання в середовищах, де цілісність та узгодженість даних є критично важливими.

Загалом, вибір MySQL для розробки бази даних музичного порталу забезпечить високу продуктивність системи, надійність і ремонтпридатність, які є необхідними для успіху проекту.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				24
Змн.	Арк.	№ докум.	Підпис	Дата		

РОЗДІЛ 2. ПРОЕКТУВАННЯ БАЗИ ДАНИХ ЗА НАПРЯМКОМ КУРСОВОЇ РОБОТИ

2.1 Проектування загального алгоритму роботи програми

Для ефективного функціонування музичного порталу важливо створити структурну схему, що відображає взаємодію між основними компонентами системи. Такі діаграми допомагають зрозуміти, як різні елементи системи працюють разом для досягнення бізнес-цілей і задоволення потреб користувачів. У цьому пункті представлені блок-схеми, що показують, як працює система, створені в інтегрованому середовищі візуального моделювання з використанням методів діаграм (Рис. 2.1.1).

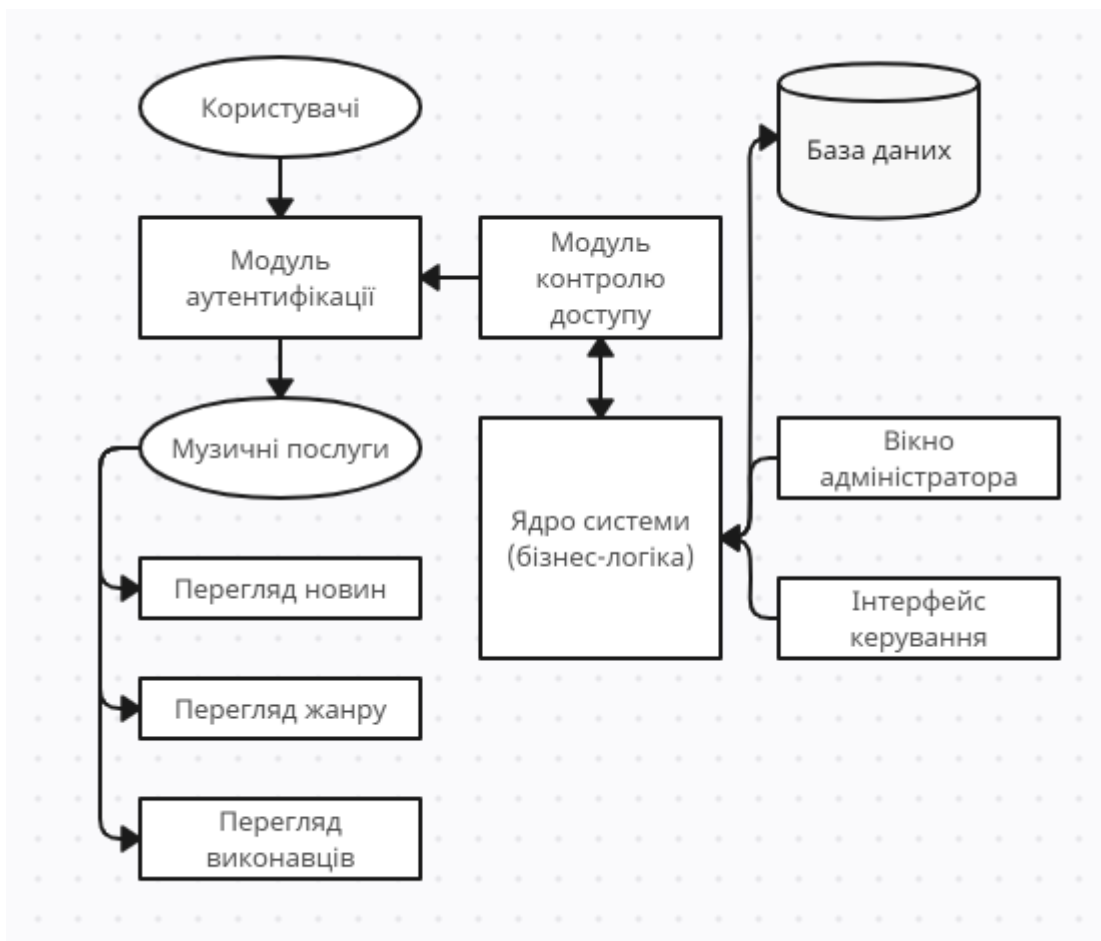


Рис. 2.1.1 Структурна схема системи музичного порталу

На основі попереднього аналізу було розроблено структурну схему системи музичного порталу, яка ілюструє основні компоненти системи та їх взаємодію.

Основні структурні блоки системи:

1. Користувачі

Користувачі є центральним елементом системи, оскільки саме для них розробляється музичний портал. Вони можуть виконувати різноманітні дії, які забезпечуються іншими компонентами системи. Користувачі поділяються на декілька категорій, зокрема:

- *Зареєстровані користувачі.*

Мають доступ до всього функціоналу порталу, включаючи можливість створювати персональні плейлисти, додавати треки до бібліотеки, залишати відгуки та оцінки, підписуватись на оновлення артистів тощо.

- *Незареєстровані користувачі*

Можуть слухати музику в режимі обмеженого доступу, переглядати інформацію про треки та артистів, але не можуть взаємодіяти з порталом на рівні збереження особистих даних або налаштувань.

- **Адміністратори**

Мають доступ до адміністративних функцій порталу, включаючи управління контентом, модерацію користувачів, аналіз статистики та роботу з аналітичними звітами.

2. Модуль аутентифікації

Модуль аутентифікації є важливою складовою системи безпеки музичного порталу. Він відповідає за перевірку облікових даних користувачів під час входу в систему. Основні функції модуля включають:

- **Реєстрація нових користувачів**

Ця функція забезпечує створення нового облікового запису на основі введених користувачем даних (ім'я, електронна адреса, пароль тощо). Важливо також передбачити підтвердження реєстрації через електронну пошту.

- **Вхід у систему**

Перевіряє відповідність введених даних користувача до даних у базі даних. У випадку успішної аутентифікації користувач отримує доступ до свого облікового запису та персональних налаштувань.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				26
Змн.	Арк.	№ докум.	Підпис	Дата		

- **Аутентифікація при кожній сесії**

Забезпечує автоматичне входження користувача у систему при повторному відвідуванні порталу, використовуючи збережені куки або токени.

3. **Модуль контролю доступу**

Модуль контролю доступу відповідає за визначення рівнів доступу для різних категорій користувачів. Це необхідно для захисту конфіденційної інформації та забезпечення безпеки даних.

Основні функції модуля включають:

- **Рольовий доступ**

Визначення ролей для користувачів (наприклад, звичайний користувач, модератор, адміністратор) та налаштування відповідних рівнів доступу.

- **Доступ до адміністративних функцій**

Забезпечення доступу лише для адміністраторів до панелі керування, де вони можуть виконувати операції з контентом, переглядати звіти, змінювати налаштування системи.

- **Контроль доступу до платного контенту**

Визначає правила доступу до преміум-функцій та платного контенту для користувачів, які мають відповідну підписку.

4. **База даних**

База даних є центральним сховищем інформації для музичного порталу. Вона забезпечує збереження та швидкий доступ до великої кількості даних, необхідних для функціонування системи. Основні компоненти бази даних включають:

- **Дані про користувачів**

Містять в собі інформацію про облікові записи користувачів, їх налаштування, оцінки та відгуки.

- **Музичний контент**

Включає метадані про альбоми, виконавців, жанри, обкладинки альбомів та інші пов'язані дані.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				
Змн.	Арк.	№ докум.	Підпис	Дата		27

В якості СУБД було обрано **MySQL** завдяки її надійності, високій продуктивності, а також підтримці складних запитів та масштабованості. MySQL дозволяє ефективно працювати з великими обсягами даних і забезпечує необхідний функціонал для підтримки всіх вимог проекту.

5. Музичні послуги

Модуль музичних послуг забезпечує доступ користувачів до музичного контенту та пов'язаних функцій, такі як доступ до музичних новин, плейлистів, рекомендацій, нових релізів та інших музичних сервісів.

6. Ядро системи (бізнес-логіка)

Є центральним компонентом, що виконує всі ключові бізнес-процеси системи. Він обробляє запити користувачів, взаємодіє з базою даних, керує музичним контентом та надає інтерфейс для адміністрування.

7. Інтерфейс користувача

забезпечує зручний та інтуїтивний доступ до функціоналу системи. Включає веб-інтерфейс та мобільні додатки, що дозволяють користувачам взаємодіяти з системою з будь-якого пристрою.

8. Вікно адміністратора

Вікно адміністратора надає розширений доступ до керування порталом для адміністраторів та модераторів. Цей інтерфейс дозволяє виконувати широкий спектр адміністративних завдань:

- **Управління контентом:** Можливість додавання, редагування, видалення музичних новин, альбомів, інформації про виконавців.
- **Керування користувачами:** Дозволяє переглядати профілі користувачів, змінювати їх ролі, а також за необхідності блокувати доступ.
- **Моніторинг системи:** Включає можливість відстеження продуктивності системи, стану серверів, потоку даних, та інших технічних показників.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				28
Змн.	Арк.	№ докум.	Підпис	Дата		

- **Адміністрування звітів:** Забезпечує доступ до генерації та перегляду статистичних та аналітичних звітів. Адміністратори можуть налаштовувати періодичність та формат звітів для отримання максимально корисної інформації.

Розроблена структурна схема музичного порталу, включаючи детальний опис основних структурних блоків, дозволяє чітко розуміти, як система функціонує та які завдання виконують окремі її компоненти. Такий підхід до моделювання системи сприяє підвищенню ефективності її розробки та експлуатації, забезпечуючи інтеграцію всіх необхідних функцій для досягнення цілей проекту.

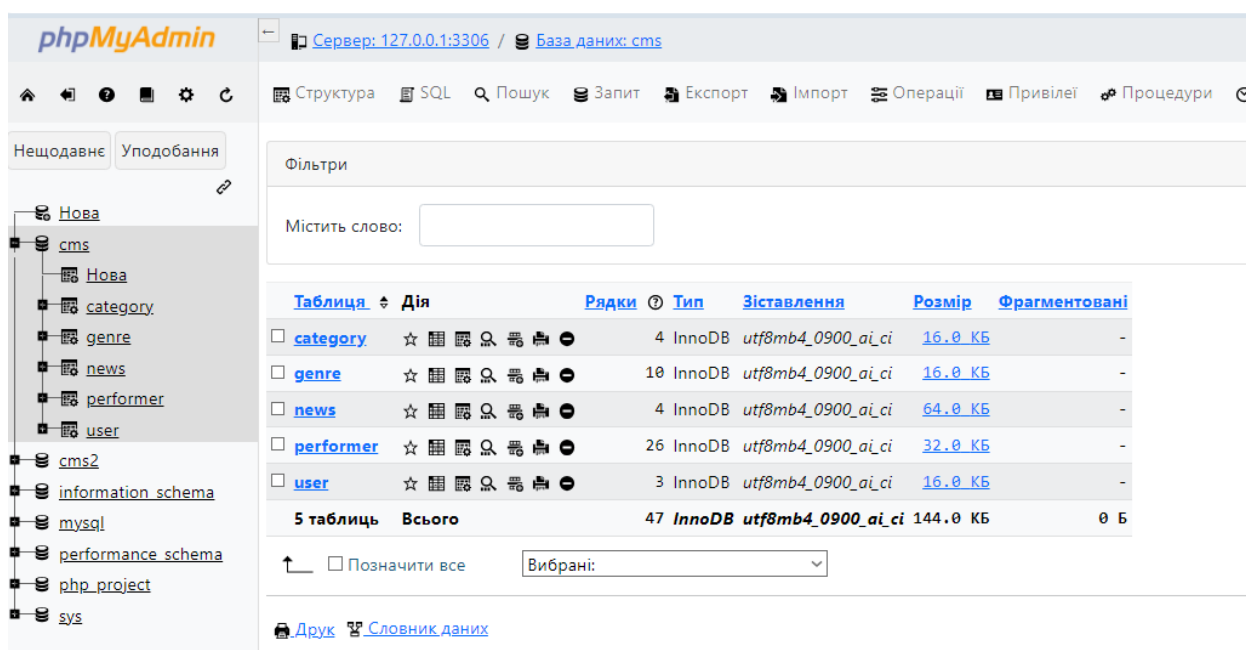
2.2 Розробка функціональних алгоритмів роботи програми

Основна мета нашої роботи – реалізувати базу даних для інтернет-магазину оптики. Як вже було зазначено раніше, для цього ми використаємо базу даних MySQL. Щоб виконати всі цілі, які були перед нами поставлені в минулих пунктах, нам знадобилось створити 5 колекцій, а саме:

- news
- performer
- genre
- category
- user

База була створена локально у середовищі PhpMyAdmin та мала наступний вигляд:

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				29
Змн.	Арк.	№ докум.	Підпис	Дата		



2.2.1. Загальний вигляд створення БД в phpMyAdmin

Розглянемо кожну таблицю детальніше, щоб дізнатись, за що вона безпосередньо відповідає:

1. genre

Таблиця містить в собі три рядки – id, name та photo (табл. 2.2.1).

Рядки «id» та «name» передаються у таблицю «performer» і «news», що дає змогу їх розрізняти за категоріями і зробити окремі унікальні поля.

Таблиця 2.2.1

Назва	Тип поля	Призначення
id	int	Ідентифікаційний код
name	varchar (255)	Назва жанру
photo	varchar (255)	Фото-опис жанру

Ці поля відповідають за жанри, які будуть наявні в музичному порталі та буде використовуватись при відображенні як виконавців, так і детальної інформації про новини, при редагуванні, додаванні та видаленні товару адміністратором, при фільтрації та пошуку даних.

2. performer

Таблиця містить в собі такі рядки як id, name та id_genre (табл. 2.2.2).

Рядок «id_genre» отримано з таблиці genre для зручного пошуку користувачами виконавців. Також поле «id» даної таблиці використовується в таблиці «news», що допомагає користувачам отримати повну інформацію новин та пов'язаних з ними виконавцями.

Таблиця 2.2.2

Назва	Тип поля	Призначення
id	int	Ідентифікаційний код
name	varchar (255)	Назва жанру
id_genre	int	Код жанру

Ці поля відповідають за виконавців, які будуть наявні в музичному порталі та буде використовуватись при відображенні як виконавців та пов'язаної з ними інформації.

3. category

Таблиця містить в собі два рядки – id та category_name (табл. 2.2.3).

Рядки «id» та «category_name» передаються у таблицю «news», що дає змогу їх розрізнити за категоріями і зробити окремі унікальні поля.

Таблиця 2.2.3

Назва	Тип поля	Призначення
id	int	Ідентифікаційний код
category_name	varchar (255)	Назва категорії

Ці поля несуть в собі інформацію про категорії новин, що відображаються на сайті. Така можливість дає користувачам завчасно інформацію про зміст новини.

4. news

Ця таблиця містить в собі такі рядки: id, title, id_genre, id_category, id_performer і text (табл. 2.2.4) .

Рядки «id_genre», «id_category» та «id_performer» отримано з таблиць «genre», «performer» і «category» що дає змогу користувачам отримати усю необхідну інформацію, переглядаючи новину.

Таблиця 2.2.4

Назва	Тип поля	Призначення
id	int	Ідентифікаційний код
title	varchar (255)	Заголовок новини
id_genre	int	Код жанру
id_category	int	Код категорії
id_performer	int	Код виконавця
text	text	Зміст новини

Ці поля несуть в собі інформацію про новини. Таблиця «news» відповідає за новини, які будуть наявні в музичному порталі та буде використовуватись при відображенні як новин, так і детальної інформації про кожний з них, при редагуванні, додаванні та видаленні товару адміністратором, при фільтрації та пошуку даних

5. user

Таблиця «user» містить в собі 5 рядків – id, login, password, firstname, lastname та access_level (табл. 2.2.5).

Таблиця 2.2.5

Назва	Тип поля	Призначення
id	int	Ідентифікаційний код
login	varchar (255)	Логін
password	varchar (255)	Пароль користувача
firstname	varchar (255)	Ім'я
lastname	varchar (255)	Прізвище
access_level	int	Рівень доступу

Таблиця “users” відповідає за усіх користувачів на сайті. Вона також буде використовуватись при усіх операціях з ними: зміна усіх даних користувача та надання різним користувачам різних прав доступу.

Діаграма бази даних:

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				
Змн.	Арк.	№ докум.	Підпис	Дата		32

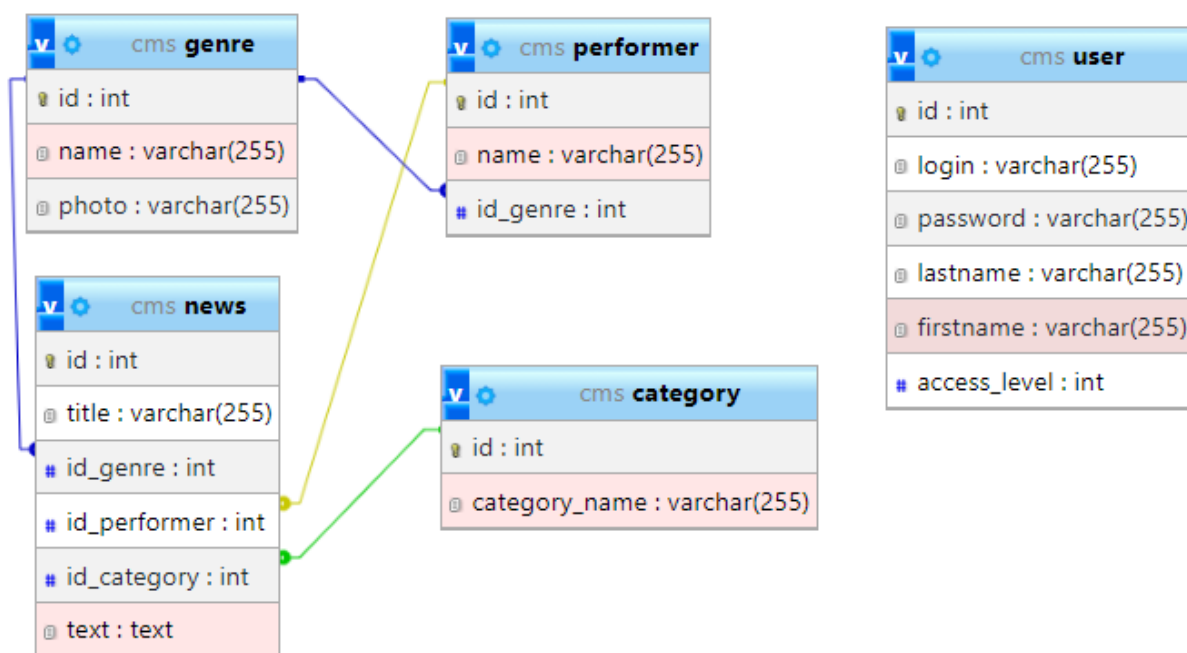


Рис. 2.2.1. ERD-діаграма бази даних

2.3 Розробка програмного забезпечення

Для реалізації проектної системи, було використано технології створення веб-сайтів: PHP з використанням технології HTML, CSS та JavaScript.

Також були залучено використання бази даних MySQL. Дані технології дозволяють виконати всі вимоги технічного завдання. Завдяки основним відомостям про викори-стані технології будуть дані відповіді як та для якої мети вони були залучені.

1. Встановлення і запуск OpenServer
2. Перегляд схожих веб-сайтів для визначення дизайну
3. Визначення функціоналу
7. Верстка сайта з використанням Bootstrap.
8. Основні частини верстки: about, abs, bloodrequest, contact, head, header, index, main, login, logout, userpage
9. Створення БД з таблицями SQL
10. Запустити проект
11. Таблиці про виконавців, користувачів, жанри, новини, категорії новин.

12. Реєстрація користувача
13. Вхід користувача
14. Вибір вкладок з запитам на кров чи донорство
15. Фільтр по жанрам
16. Вихід з акаунту користувача
17. Пошук
18. Панель редагування категорій і жанрів
19. Додавання музичних новин.
20. Редагування новин.
21. Вихід з акаунту лікарні

Висновки до другого розділу.

У даному розділі було сформовано загальний алгоритм роботи програми, спроектовано архітектуру та розглянуто загальні принципи роботи.

Спроектовано базу даних, описано її таблиці та поля та визначено взаємодію між її ними. Опрацьовано математичні операції, які є необхідними для музичного порталу.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				
Змн.	Арк.	№ докум.	Підпис	Дата		34

РОЗДІЛ 3. ОПИС РЕАЛІЗАЦІЯ ПІДСИСТЕМИ ОБРОБКИ ДАНИХ ЗА НАПРЯМКОМ «БАЗА ДАНИХ МУЗИЧНОГО ПОРТАЛУ»

3.1 Проектування інтерфейсу обробки даних

В даному пункті було створено структуру інтерфейсу музичного пор-
талу для спрощення розробки веб-інтерфейсу (Рис.3.3.1).

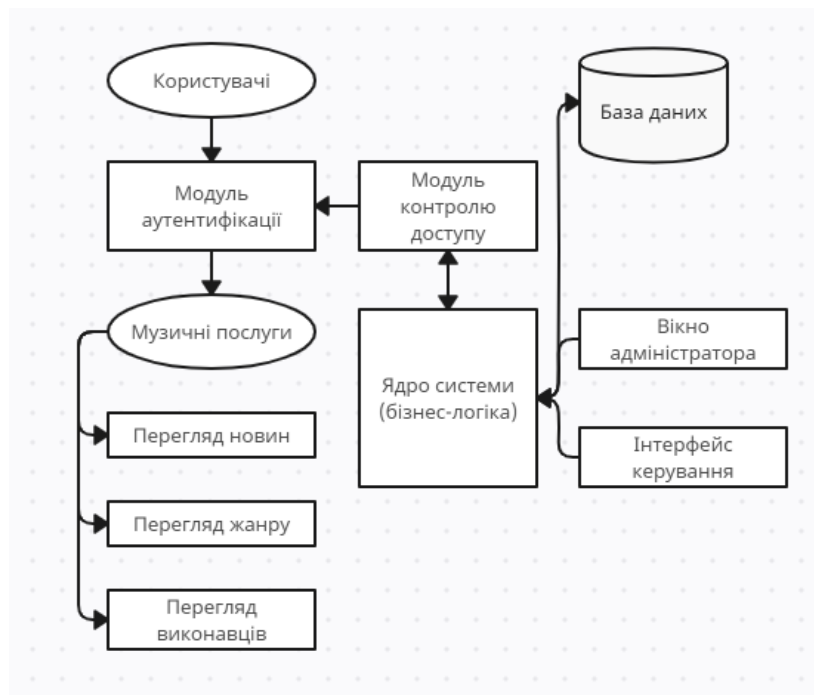


Рис. 3.3.1 Загальна структура інтерфейсу інтернет-магазину

Також було розглянуто і створено алгоритм побудови веб-сторінки, не-
обхідний для подальшої розробки веб-додатку.

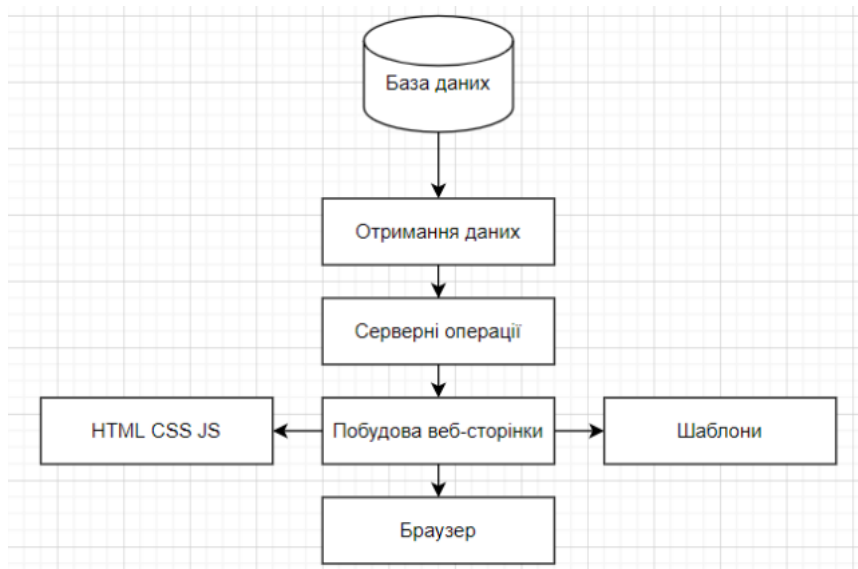


Рис. 3.3.2 Алгоритм побудови веб-сторінки

		Ватиль В.В.		
		Сугоняк І.І.		
Змн.	Арк.	№ докум.	Підпис	Дата

У проєкті також використовуємо патерн MVC (Model View Controller) для двосторонньої взаємодії динамічних веб-сторінок. Його алгоритм роботи виглядає наступним чином:

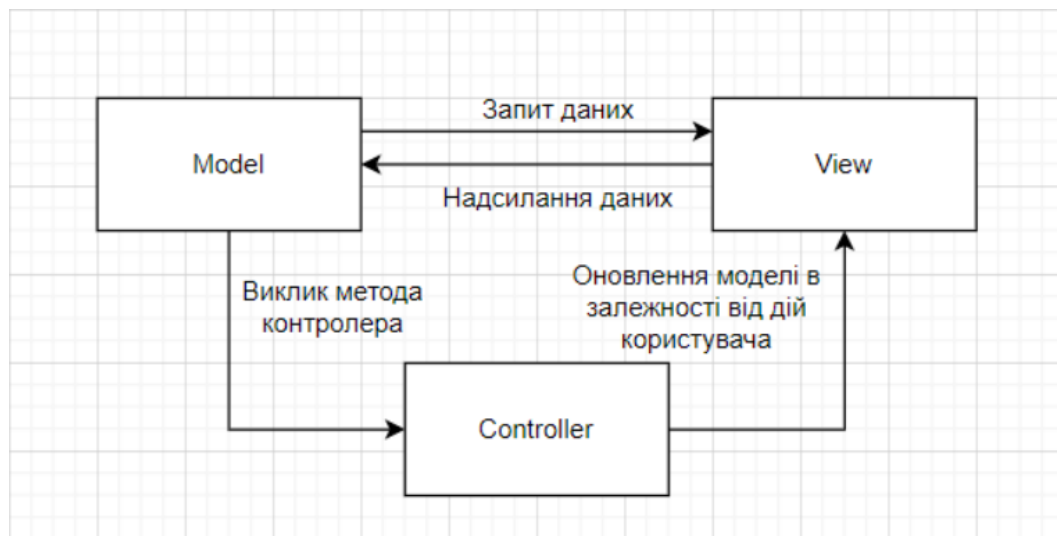


Рис. 3.3.3 Алгоритм роботи патерну MVC у додатку

3.2. Реалізація операцій обробки даних в БД за напрямком «база даних для музичного порталу»

Для початку реалізуємо підключення до бази даних за допомогою метода connect():

```

public function Initialize() { //встановлення з'єднання з базою даних
    session_start();

    $this->db = new DB(DATABASE_HOST, DATABASE_LOGIN,
    DATABASE_PASSWORD, DATABASE_BASENAME);

    $this->requestMethod = $_SERVER['REQUEST_METHOD'];
}
  
```

Найважливішим моментом є виведення новин. Воно виконується одразу з урахуванням фільтрів (його категорією) та має наступний вигляд:

```

$stmt1 = $conn->prepare("SELECT COUNT(*) As total_records
FROM products WHERE product_category=? ");
  
```

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				36
Змн.	Арк.	№ докум.	Підпис	Дата		

```

$stmt1->bind_param('s', $category);

$stmt1->execute();

$stmt1->bind_result($total_records);

$stmt1->store_result();

$stmt1->fetch();

$total_records_per_page = 8;

$offset = ($page_no - 1) * $total_records_per_page;

$previous_page = $page_no - 1;

$next_page = $page_no + 1;

$adjacents = "2";

$total_no_of_pages = ceil($total_records / $total_records_per_page);

$stmt2 = $conn->prepare("SELECT * FROM products WHERE
product_category=? LIMIT $offset,$total_records_per_page");

$stmt2->bind_param("s", $category);

$stmt2->execute();

$products = $stmt2->get_result(); //[]

```

Було прописано код для реєстрації користувача та шифрування його паролю в базі даних:

```

public static function addUser($login, $password, $lastname,
$firstname)//додання у БД нового користувача
{
    \core\Core::getInstance()->db->insert(//з'єднання з базою даних

    self::$tableName, [

```

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				37
Змн.	Арк.	№ докум.	Підпис	Дата		

```

        'login' => $login,

        'password' => self::hashPassword($password),

        'lastname' => $lastname,

        'firstname' => $firstname

    ]

);

}

public static function hashPassword($password) //одностороннє шифру-
вання пароллю

{

    return md5($password);

}

```

Для авторизованого користувача, щоб його вхід в систему був успішний, необхідно знайти його унікальний адрес поштової скриньки та перевірити пароль (фрагмент коду).

```

$user = \core\Core::getInstance()->db->select(self::$tableName, '*', [

    'login' => $login

]);

return !empty($user);

```

Також у нас є роль адміністрування, де можливо редагувати, додавати та видаляти категорії.

Створення категорії:

```

public static function addCategory($name, $photoPath)

{

```

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				38
Змн.	Арк.	№ докум.	Підпис	Дата		

```

do {

    $fileName = uniqid() . '.jpg';//створення унікального імені для
фото

    $newPath = "files/category/{ $fileName}";

} while (file_exists($newPath));

move_uploaded_file($photoPath, $newPath);

Core::getInstance()->db->insert(self::$tableName, [

    'name' => $name,//назва категорії

    'photo' => $fileName

]);

}

```

Змінення категорії:

```

public static function updateCategory($id, $newName)

{

    Core::getInstance()->db->update(self::$tableName, [

        'name' => $newName

    ], [

        'id' => $id

    ]);

}

```

```

public static function getCategories()//отримання списку всіх кате-
горій

```

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				39
Змн.	Арк.	№ докум.	Підпис	Дата		

```

{
    $rows = Core::getInstance()->db->select(self::$tableName);

    return $rows;
}

```

Видалення категорії:

```

public static function deleteCategory($id)
{
    Core::getInstance()->db->delete(self::$tableName,
    [
        'id' => $id
    ]);
}

```

3.3. Організація системи та тестування

При відкритті веб-додатка пешою показується головна сторінка, на якій зверху та знизу знаходиться навігація по сайту, реалізована карусель на якій відображено різні новини. На сторінці Контакти знаходиться інформація за якою можна зв'язатися з адміністратором.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				40
Змн.	Арк.	№ докум.	Підпис	Дата		

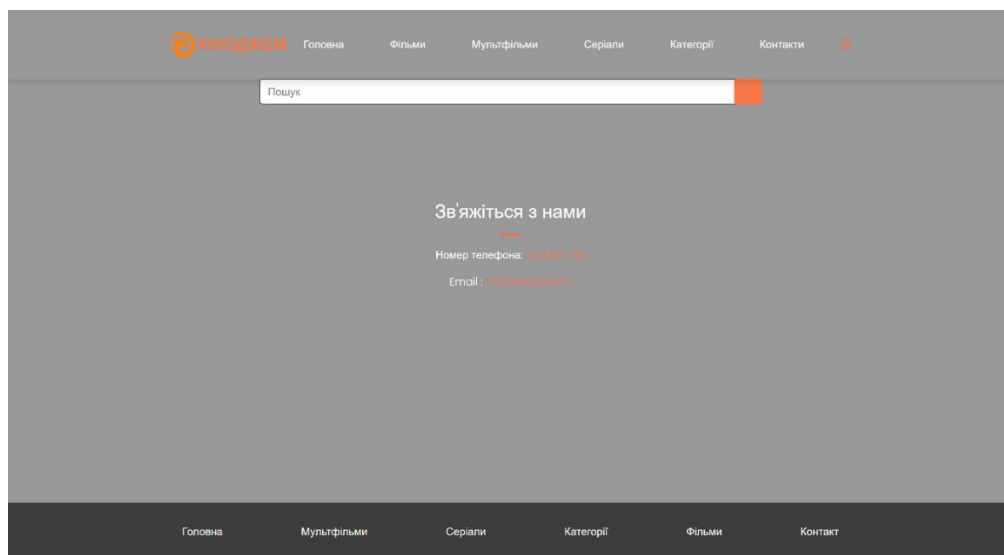


Рисунок 3.2.1. Сторінка контактів

На сторінці Авторизації, на яку можна перейти нажавши на знак профілю справа зверху, можна авторизуватися вже зареєстрованому користувачу.

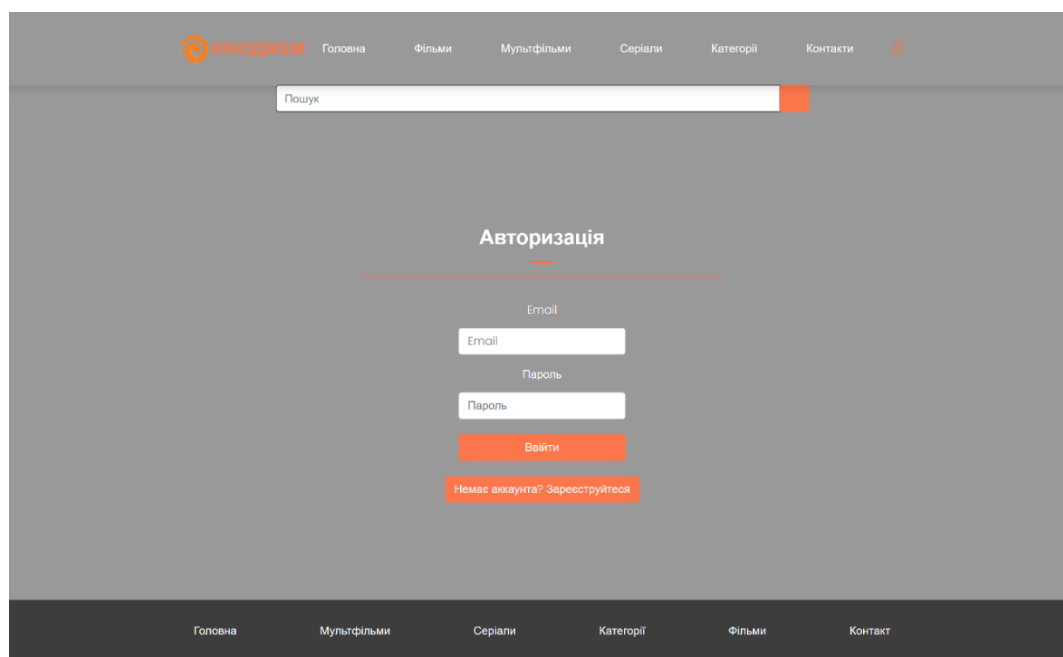


Рисунок 3.2.2. Сторінка авторизації

На сторінці Реєстрація, на яку можна перейти нажавши на знак профілю справа зверху, та потім нажавши “Немає аккаунта? Зареєструйтеся”, можна зареєструватися новому користувачу.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				41
Змн.	Арк.	№ докум.	Підпис	Дата		

Рисунок 3.2.3. Сторінка реєстрації

Після успішної авторизації користувач потрапляє до себе в профіль де є його ім'я та email, а також він може змінити пароль.

Рисунок 3.2.4. Авторизація користувача

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				42
Змн.	Арк.	№ докум.	Підпис	Дата		

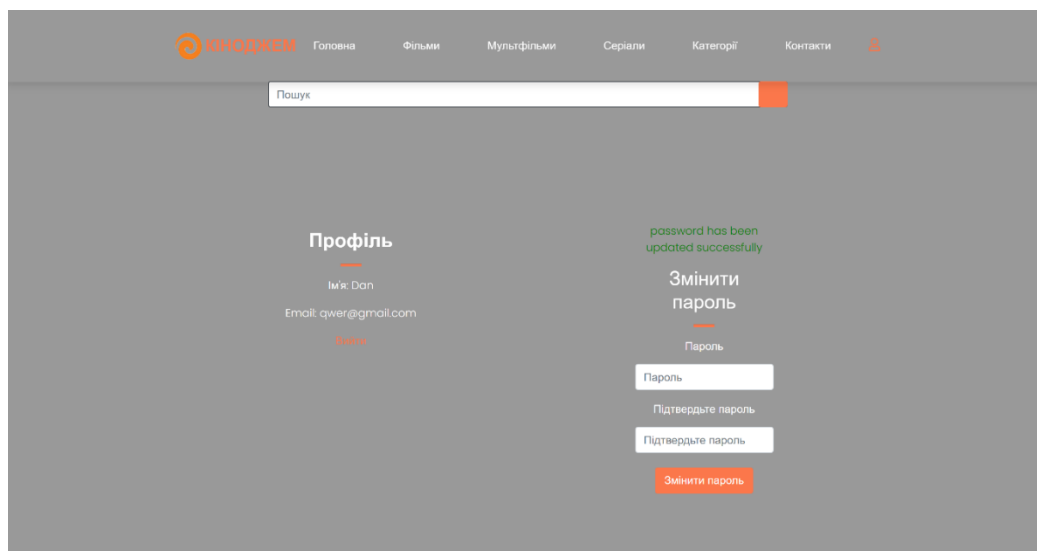


Рисунок 3.2.5. Зміна паролю користувача

Для того щоб увійти в систему керування спочатку в адресній строці потрібно ввести /admin, після чого можна потрапити в систему авторизації (Рисунок 3.1), де вводиться правильний email та пароль(в даному випадку це: email: admin@gmail.com, пароль: 12345).

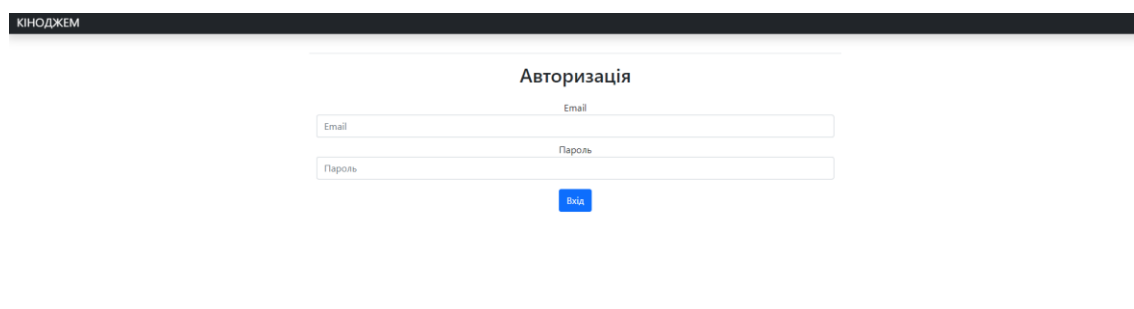


Рисунок 3.2.6. Авторизація в нашу cms

Після успішного входу до системи керуванням вмістом сайту ми переходимо на сторінку адміністратора де розміщується id адміністратора, його ім'я, та email (ці данні залежать від користувача який зайшов в систему керування, які зберігаються в базі даних).

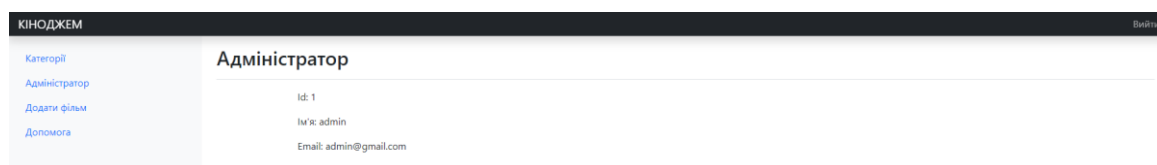


Рисунок 3.2.7. Сторінка адміністратора

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				43
Змн.	Арк.	№ докум.	Підпис	Дата		

При переході на сторінку “Новини” можна побачити все кіно, що було добавлено. Воно розміщено в таблиці, в якій є id, картинка, назва, категорія, відео, зміна картинки, редагування та видалення кіно.

Id	Картинка	Назва	Категорія	Відео	Змінити картинку	Редагувати	Видалити
1		Меню	film	https://www.youtube.com/embed/Kx5SRkyrh0k	Змінити картинку	Редагувати	Видалити
2		Джонні	film	https://www.youtube.com/embed/EYXUC3sd7Q	Змінити картинку	Редагувати	Видалити
3		Аватар: шлях води	film	https://www.youtube.com/embed/d9MyW72ELq0	Змінити картинку	Редагувати	Видалити
4		Лета ніка	film	https://www.youtube.com/embed/s53e4Hm_s	Змінити картинку	Редагувати	Видалити
5		Про долю	film	https://www.youtube.com/embed/6VNN3w0Gpw	Змінити картинку	Редагувати	Видалити

Рисунок 3.2.8. Категорії

Після натискання на “Змінити картинку” можна змінити картинку.

Рисунок 3.2.9. Редагування картинки

Після натискання на “Редагувати” можна змінити назву, опис категорію та відео картинки. (Рисунок 3.1.19)

Рисунок 3.2.10. Редагування вмісту

Після натискання на “Видалити” можна видалити кнію.

При переході на сторінку “Додати” можна додати новину, а точніше

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк. 44
		Сугоняк І.І.				
Змн.	Арк.	№ докум.	Підпис	Дата		

його назву, опис, категорію, відео та картинку.

The screenshot shows the 'Додати кіно' (Add movie) form in the 'КІНОДЖЕМ' application. The form is located on the right side of the page, with a sidebar on the left containing links to 'Кіно', 'Адміністратор', 'Додати кіно', and 'Допомога'. The form fields are: 'Назва' (Name), 'Опис' (Description), 'Категорії' (Categories) with a dropdown menu showing 'Новинки', 'Відео' (Video), and 'Картинка' (Image) with a file upload button 'Виберіть файл' and a status 'Файл не вибран'. A blue 'Зберегти' (Save) button is at the bottom of the form.

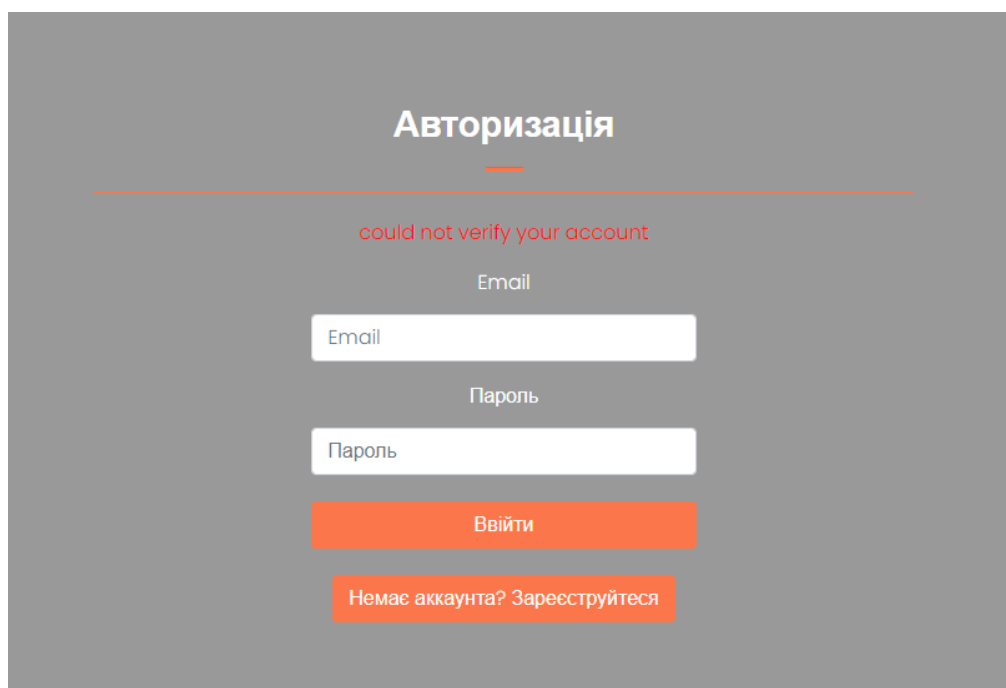
Рисунок 3.2.11. Додавання

На сторінці “Допомога” можна вказати email та номер людини яка відповідальна за веб-додаток.

The screenshot shows the 'Допомога' (Help) page in the 'КІНОДЖЕМ' application. The page is located on the right side of the page, with a sidebar on the left containing links to 'Кіно', 'Адміністратор', 'Додати кіно', and 'Допомога'. The page displays contact information: 'Email: admin@email.com' and 'Номер: 12345678'.

Рисунок 3.2.12. Редагування вмісту

Зроблена перевірка при авторизації та реєстрації, якщо при авторизації неправильний email чи пароль, висвітиться текст про помилку (Рис. 3.2.13). Якщо при реєстрації якесь з полів не заповнене; паролі не співпадають; пароль менше 6 символів; email вже використовується – виникне відповідна помилка (Рисунок 3.2.14).



Авторизація

could not verify your account

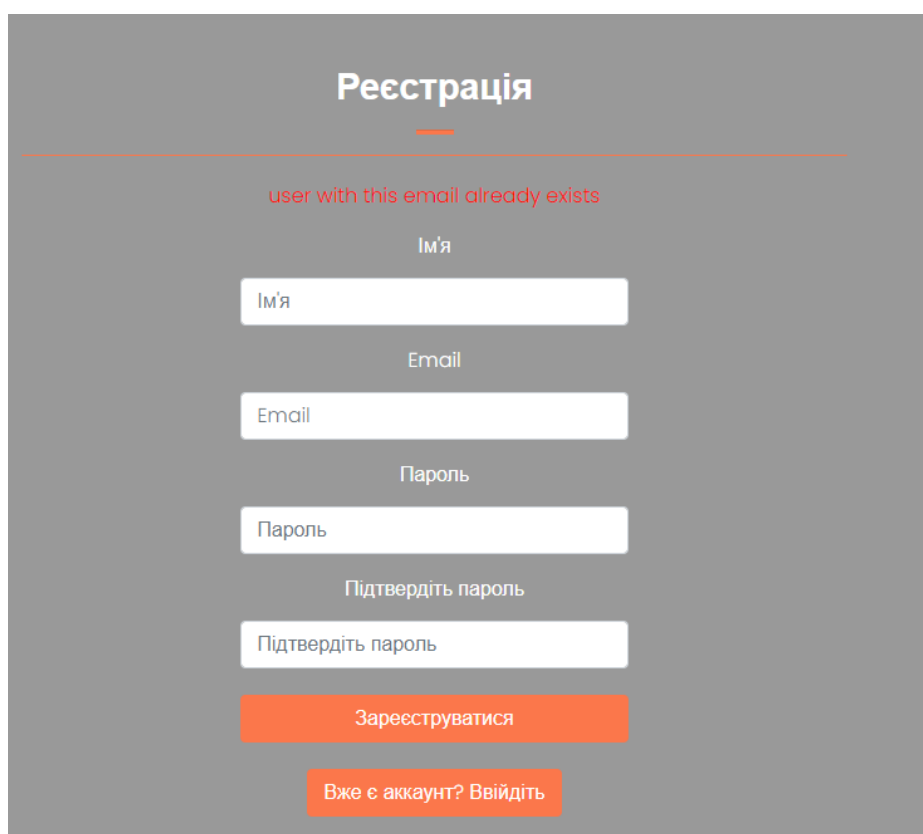
Email

Пароль

Ввійти

Немає аккаунта? Зареєструйтеся

Рисунок 3.2.13. Помилки при авторизації



Реєстрація

user with this email already exists

Ім'я

Email

Пароль

Підтвердіть пароль

Зареєструватися

Вже є аккаунт? Ввійдіть

Рисунок 3.2.14. Помилки при реєстрації

Аби в систему керування даними не змогли зайти прості користувачі, була створена авторизація, при введенні неправильних даних буде

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				46
Змн.	Арк.	№ докум.	Підпис	Дата		

виводиться помилка (Рисунок 3.2.15).

КІНОДЖЕМ

Авторизація

could not verify your account

Email

Email

Пароль

Пароль

Вхід

Рисунок 3.2.15. Помилка при спробі реєстрації

Висновки до третього розділу

Було спроектовано інтерфейс обробки даних, реалізовано операції обробки даних в базі даних, детально опрацьовано вибірку даних. Також організовано та продемонстровано звітність системи.

РОЗДІЛ 4. АДМІНІСТРУВАННЯ БАЗ ДАНИХ

4.1. Розробка заходів захисту інформації в БД

Для доступу до бази даних було створено 4 види користувачів з різними типами доступу, а саме: неавторизований користувач, авторизований користувач та адміністратор.

1) Неавторизований користувач: можливість перегляду новин. Неавторизований користувач не має особистого кабінету, тому не може змінити дані про себе.

2) Авторизований користувач: можливість перегляду товарів, залишити коментар та замовити необхідну їх кількість. У створеному особистому кабінеті авторизований користувач може змінити пароль.

3) Адміністратор: можливість перегляду новин, їх додавання по категоріям, редагування та видалення без можливості їх відновлення.

Паролі в базі даних зберігаються в хешованій формі. Хешуються паролі за допомогою вбудованої функції md5 – алгоритмом хешування, що описаний в RFC 1321:

```
public static function hashPassword($password) //одностороннє шифрування паролю
```

```
{  
  
    return md5($password);  
  
}
```

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				
Змн.	Арк.	№ докум.	Підпис	Дата		48

	<u>id</u>	<u>login</u>	<u>password</u>
	ID	Логін	Пароль
1	ipz212_vvv@student.ztu.edu.ua	4527c9b5fc3955d911b2df21c81f4563	
2	ipz212@student.ztu.edu.ua	e10adc3949ba59abbe56e057f20f883e	
3	ipz21_vvv@student.ztu.edu.ua	b59c67bf196a4758191e42f76670ceba	

Рис. 4.1.1 Відображення паролів у базі даних

4.2. Налаштування параметрів роботи з MySQL

Для реалізації даного проекту нам потрібна база даних і місце де вона буде зберігатися. Для цього ми використовували офіційний графічний клієнт для адміністрування та перегляду даних Open Server Panel, де присутній phpMyAdmin для використання реляційних БД. Для його встановлення необхідно зайти на офіційний сайт та перейти до встановлення:



Open Server Panel

Версия 6.0.0 Сейчас в сети 10 717 Всего загрузок 3 885 058

Главная Новости Скачать Документация Форум Поддержка

Программная среда для веб-разработки

Open Server Panel — это портативная программная среда, созданная специально для веб-разработчиков с учётом их рекомендаций и пожеланий.

Наш программный пакет включает в себя тщательно подобранный набор серверного программного обеспечения, а также удобную в использовании утилиту с широкими возможностями для администрирования и настройки всех доступных компонентов.

На сегодняшний день **OSPanel** широко используется веб-разработчиками по всему миру с целью разработки, отладки и тестирования веб-проектов, а также для предоставления веб-сервисов в локальных сетях. Наибольшую популярность проект завоевал у начинающих веб-мастеров, поскольку позволяет быстро создать рабочее окружение и сразу начать изучение веб-технологий без сложных манипуляций по установке и настройке большого количества незнакомых программных компонентов.

Устройство среды

Рис. 4.2.1 Офіційний сайт Open Server Panel

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				49
Змн.	Арк.	№ докум.	Підпис	Дата		

Для цього обираємо версію, операційну систему, на яку буде встановлено та завантажуюємо файл типу exe. Знаходимо його у встановленому нами місці та запускаємо:

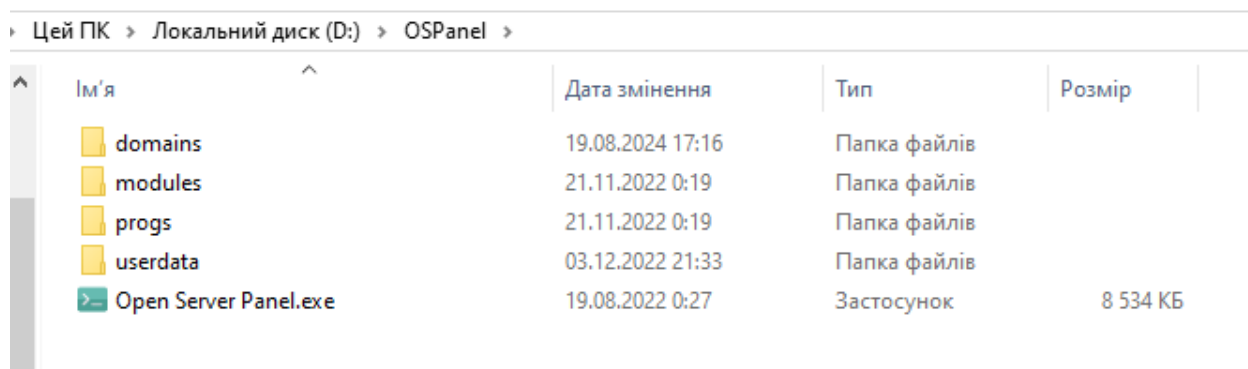


Рис. 4.2.2 Файл для запуску Open Server Panel.exe

Після запуску програми ми можемо запустити phpMyAdmin та створити необхідну базу даних:

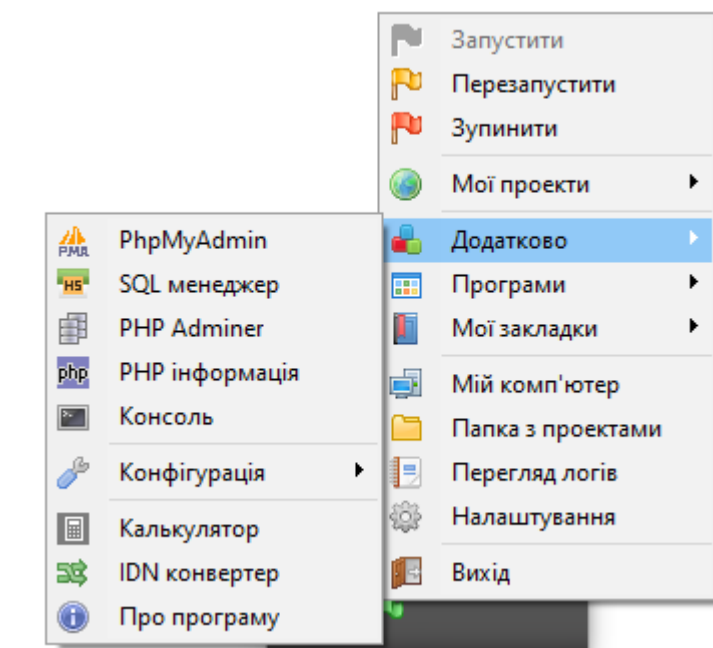


Рис. 4.2.3 Процес відкриття phpMyAdmin

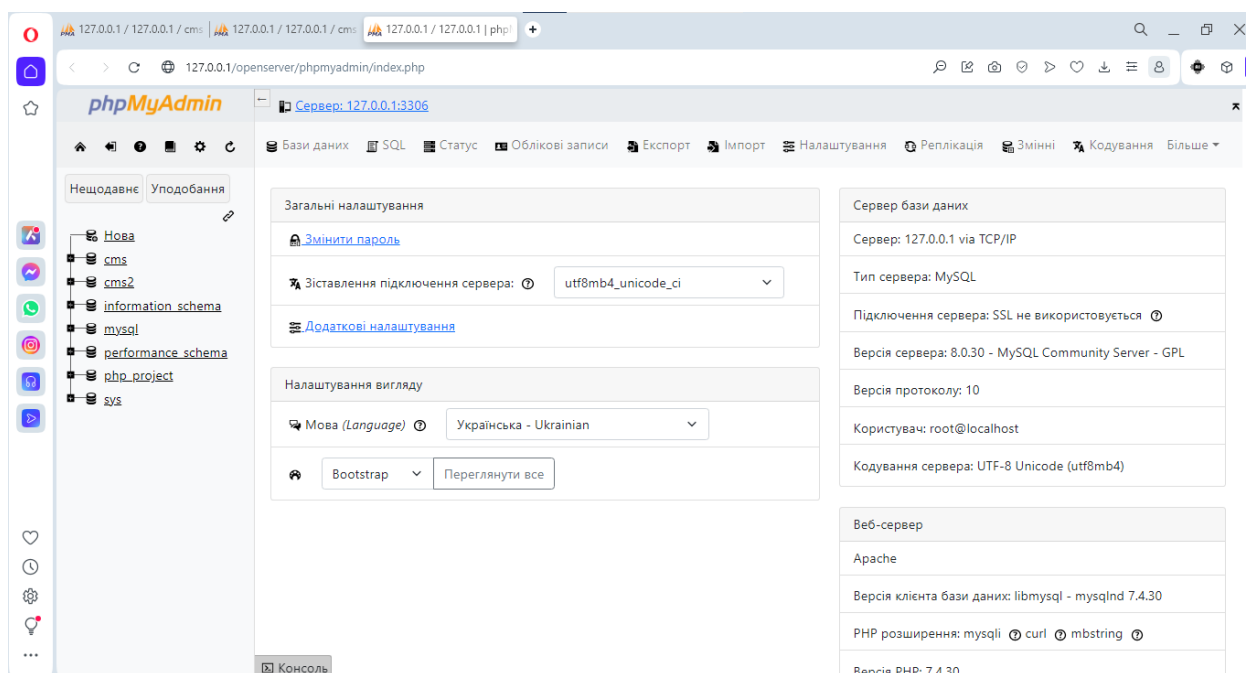


Рис. 4.2.4. Інтерфейс phpMyAdmin

Висновки до четвертого розділу

Для захисту інформації в базі даних було створено три рівня доступу до інформації, а саме: неавторизований користувач, авторизований користувач та адміністратор. Було детально розписано відмінності кожної з ролей та була створена матриця доступу для кращого розуміння. Було детально описано роботу зі зберіганням нашої бази даних.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				51
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

У процесі написання курсової роботи було здобуто важливий досвід роботи з базою даних MySQL, використовуючи інструмент phpMyAdmin для управління та адміністрування даних. Робота над проектом дозволила повністю розробити базу даних для музичного порталу, що забезпечує зручне управління великою кількістю користувачів та музичних ресурсів.

У першому розділі було сформульовано основні та допоміжні завдання системи на основі аналізу існуючих музичних порталів. Після аналізу було вибрано MySQL як основну базу даних завдяки її високій продуктивності, масштабованості та підтримці SQL-запитів, що забезпечує ефективну обробку великої кількості даних.

У другому розділі було розроблено загальний алгоритм роботи системи, що включає всі необхідні операції для управління музичним контентом та користувачами порталу.

У третьому розділі реалізовано інтерфейс для обробки даних, що включає операції вставки, видалення, редагування та вибірки даних, необхідних для роботи музичного порталу. Завдяки використанню phpMyAdmin, було досягнуто високого рівня зручності в управлінні БД.

У четвертому розділі було приділено увагу безпеці даних. Було реалізовано різні рівні доступу для адміністраторів та звичайних користувачів, що забезпечує захист конфіденційної інформації. Також розглянуто заходи щодо захисту від SQL-ін'єкцій та інших можливих загроз. Описано процес роботи з phpMyAdmin, який дозволяє ефективно управляти доступом до бази даних і контролювати дії користувачів.

У результаті виконаної роботи було створено функціональний музичний портал з інтегрованою базою даних MySQL. Розроблена система відповідає вимогам сучасних музичних порталів і забезпечує високий рівень продуктивності та безпеки даних.

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				52
Змн.	Арк.	№ докум.	Підпис	Дата		

СПИСОК ЛІТЕРАТУРИ

1. Backend-розробка, Освітній портал ДУ «Житомирська політехніка».
<https://learn.ztu.edu.ua/course/view.php?id=4750>
2. Frontend-розробка, Освітній портал ДУ «Житомирська політехніка».
<https://learn.ztu.edu.ua/course/view.php?id=4749>
3. Електронний HTML і CSS довідник: <https://html-css.co.ua/>
4. MVC Design Pattern: <https://www.geeksforgeeks.org/mvc-design-pattern/>.
5. Сучасний підручник Javascript: <https://uk.javascript.info/>
6. Документація з PHP: <https://www.php.net/manual/uk/index.php>
7. Javascript Tutorial: <https://www.w3schools.com/js/>
8. Посібник з PHP: <https://www.php.net/>
9. Документація з використання SQL: <https://uk.wikibooks.org/wiki/SQL>
10. Документація з використання MySQL: <https://dev.mysql.com/doc/>
11. Документація з MySQL: <https://dev.mysql.com/doc/refman/8.0/en/>
12. Тьюторіал по використанню MySQL: <https://www.mysqltutorial.org/>
13. Інтерактивний посібник по SQL: <https://www.w3schools.com/sql/>
14. Стаття по схемам БД: <https://foxminded.ua/skhemy-bazy-danyh/>
15. Документація по phpMyAdmin: <https://docs.phpmyadmin.net/uk/latest/>
16. Документ по MySQL <https://downloads.mysql.com/docs/administrator-uk.pdf>
17. Створення БД:
<https://burstnet.com.au/index.php/knowledgebase/70/How-to-create-a-MySQL-database-in-cPanel.html?language=ukrainian>
18. MVC-паттерн https://visnikkrnu.kdu.edu.ua/statti/2022_1_16.pdf
19. CMS https://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0_%D0%BA%D0%B5%D1%80%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F_%D0%B2%D0%BC%D1%96%D1%81%D1%82%D0%BE%D0%BC

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				53
Змн.	Арк.	№ докум.	Підпис	Дата		

20. CMS використання: <https://developers.hubspot.com/docs/cms/guides>
21. ТUTORІАЛ створення CMS <https://docs.cmsmadesimple.org/tutorials/a-very-basic-website/tutorial-a-very-basic-website-introduction>
22. Стаття по CMS: <https://kinsta.com/knowledgebase/content-management-system/>
23. Stack Overflow: <https://stackoverflow.com/>

		Ватиль В.В.			ДУ «Житомирська політехніка».23.121.02.000 – ПЗ	Арк.
		Сугоняк І.І.				
Змн.	Арк.	№ докум.	Підпис	Дата		54